Inaugural-Dissertation

zur Erlangung der Doktorwürde der

Gesamtfakultät für Mathematik,
Ingenieur- und Naturwissenschaften

der

Ruprecht-Karls-Universität
Heidelberg

vorgelegt von

Dennis Aumiller, M.Sc.

aus Günzburg

Tag der mündlichen Prüfung:

Towards a Unified Framework for Aspect-based Multi-document Text Summarization

by

DENNIS AUMILLER

Supervisor: Prof. Dr. Michael Gertz

Abstract

For a growing number of knowledge workers, the rapid ingestion of textual information is crucial for their daily tasks. Confronted with expansive bodies of text, the fastest way to glean central pieces of information is usually a *summary*, condensing the most relevant points into a shorter piece of text. However, the manual curation of high-quality text summaries is a laborious and time-intensive task, requiring intense focus and attention. This motivates the central topic of this thesis: *the automatic generation of textual summaries*. Instead of relying on humans, we intend to summarize texts with the help of algorithms, designed to capture the central importance. Yet, despite decades of research into automatic text summarization systems, we are still not at a point where the resulting algorithms could provide the basis for a product that sees large-scale adoption by the general public.

This thesis focuses on this obvious gap and provides a fundamental framework to address some of the remaining shortcomings in automatic text summarization systems. We investigate the direction of current research, and detail key challenges, which we divide into three central problems. 1) Modern neural network-based approaches to text summarization are extremely data-hungry, yet *high-quality, task-specific data* remains a scarce resource, particularly for languages besides English. 2) From a modeling perspective, we also point out that *existing works over-index on narrow domains*, such as news summarization, with an additional lack of inclusion of user-centric perspectives for summary generation. 3) We reiterate the *lack of comprehensive and meaningful evaluations* of text summarization systems. Where systemic comparisons nowadays rely on a singular ground truth and metric scores, subjective and nuanced differences in a summary should be included in more evaluations again.

For all three of these focus areas–data, evaluation, and models–we work towards the elimination of remaining issues under a shared theoretical framework. We introduce two new datasets suitable for research purposes, enabling multilingual and domain-specific summarization applications, ensuring their quality standards with semi-automatic filtering techniques. To improve the utility of evaluations, we further provide an overview of failure cases in existing evaluation setups, and reiterate the necessity of focusing on truthful summary generation, by providing a metric for factuality-focused evaluation of generated summaries.

Aggregating these insights from our investigation of existing limitations, we introduce a two-staged hybrid summarization model, combining a multi-aspect-oriented retrieval system with a similarly aspect-compatible re-writing module as a second stage. We hypothesize that this framework allows for a more user-centric experience for text summarization systems by enabling a customizable generation depending on user needs. The final two chapters focus on the practical consequences of such a two-staged model at the example of specific generation and retrieval aspects, and how these can be improved.

Zusammenfassung

Für eine wachsende Zahl von Wissensarbeitern ist die schnelle Aufnahme von textueller Information von entscheidender Bedeutung. Angesichts extrem langer Texte bieten *Zusammenfassungen*, die die wichtigsten Punkte in einem kürzeren Textstück wiedergeben, oft den schnellste Weg, solches Wissen zu erfassen. Die manuelle Erstellung von qualitativ hochwertigen Textzusammenfassungen ist jedoch eine mühsame und zeitaufwändige Aufgabe, die ihrerseits einen hohen Grad an Konzentration und Aufmerksamkeit erfordert. Dies bringt uns zum zentralen Thema dieser Dissertation: *die automatische Erstellung von Textzusammenfassungen*. Anstatt sich auf Menschen zu verlassen, wollen wir Texte mit Hilfe von Algorithmen zusammenfassen, die darauf ausgelegt sind, die gleichen zentralen Aspekte eines Textes wiederzugeben. Trotz jahrzehntelanger Forschung auf dem Gebiet der automatischen Textzusammenfassung sind wir jedoch noch immer nicht an einem Punkt, an dem Produkte, die auf dieser Forschung basieren, in der Allgemeinheit angekommen sind.

Die vorliegende Arbeit konzentriert sich nun auf diese offensichtliche Diskrepanz und bietet ein grundlegendes Rahmenwerk, um die verbleibende Probleme automatischer Textzusammenfassungssystemen anzugehen. Wir analysieren den Stand der aktuellen Forschung und zeigen einige der wichtigsten Herausforderungen auf, die wir in drei wesentliche Problembereiche unterteilen. 1) Moderne, auf neuronalen Netzen basierende Ansätze zur Textzusammenfassung sind extrem datenhungrig. Dennoch sind *qualitativ hochwertige und domänenspezifische Datensätze schwer zu finden*. 2) Wir merken weiterhin an, dass sich *bestehende Arbeiten überwiegend auf hochspezifische Aufgabenbereiche fokusieren* und dabei nutzerorientierte Aspekte für die Generierung außen vor lassen. 3) Schließlich verweisen wir auch auf den *Mangel an aussagekräftigen Evaluierungen* von Textzusammenfassungssystemen. Hier sollten besonders subjektive und nuancierte Unterschiede in einer Zusammenfassung wieder stärker in die Evaluation einbezogen werden.

Für alle drei dieser Schwerpunkte–Daten, Auswertung und Modellierung–leisten wir konstruktive Beiträge zur Beseitigung der verbleibenden Probleme. Wir stellen zwei neue Datensätze vor, die mehrsprachige und domänenspezifische Anwendungen im Kontext von Zusammenfassungen ermöglichen. Um Evaluationsmethoden zu verbessern, geben wir darüber hinaus einen Überblick über systematische Fehlstellungen in bestehenden Analysen. Wir bekräftigen zudem den Fokus auf die Evaluation faktengetreuer Zusammenfassung mittels einer neuartigen Metrik.

Wir stellen schließlich ein zweistufiges Modell für Textzusammenfassungen vor, welches ein aspektorientiertes Suchsystem mit einem kompatiblen Modul zur Umschreibung als zweiter Stufe kombiniert. Wir argumentieren, dass dieses Mdoell einen stärkeren Fokus auf subjektiv anpassbare Zusammenfassungen ermöglicht, da die Generierung in Abhängigkeit der Nutzerbedürfnisse besser abbildbar ist. Wir diskutieren zudem praktischen Konsequenzen eines solchen zweistufigen Modells am Beispiel spezifischer Generierungs- und Suchaspekte.

Acknowledgments

This thesis is only possible with the continued support and encouragement from many people. First and foremost, I want to thank my advisor, Michael Gertz, for his continuous support and supervision throughout all the years. Over the time I realized the amount of sheer energy it takes to provide students with a productive and welcoming environmment, and fostering the continuous encouragement academia requires "behind the scenes", and I still don't know how you make it look so effortless. I additionally want to extend my thanks to Stefan Riezler, for coming on board as a secondary advisor, and providing valuable guidance for the direction of this thesis.

My awesome colleagues past and present deserve special thanks, too: Satya Almasian, whom I had the pleasure of sharing much of my PhD with. I am grateful not only for the productive collaborations, but also sharing all the ups and downs of a PhD (with the definite highlight being the conference travels), and for helping me stay on track particularly during the final writing phase. Andreas Spitz, who has been encouraging particularly my early research efforts, and shaped much of my critical reviewer eye over time. Ashish Chouhan, Philip Hausner, Sebastian Lackner, Ghadeer Mobasher, Nicolas Reuter, Jayson Salazar and John Ziegler, for sharing the office, lunch breaks, and ideas with me, and enduring my endless feedback and suggestions.

A lot of thanks also goes to the many talented students that shaped this thesis in many different ways and who have taught me how to be an aspiring advisor myself. Jing, who managed to squeeze co-authorship of two papers into an undergraduate degree. David, Jiahui, Lisa, Lucy and Sarah, who among several others endured my nit-picky code reviews and have been extremely enjoyable student collaborators throughout their time at the group. Victoria, Andreas, Diego and Achim for entrusting me with all the facets of PhD life before I even started, and teaching me that the length of a board games may scale linearly in the number of players.

Beyond academia, I am extremely grateful for the research opportunities that further helped me grow as a researcher. Alex Wang and my colleagues at Cohere, who have shaped the final result of this work by revealing the sheer complexity of modern model training, and providing me with enough mental capacity to finish the thesis. The team at Codefy, for teaching me a thing or two about working with lawyers and search, and Omar Zaidan for taking a chance on me as an intern at Amazon. Special thanks go to Silvia Severini, for being a constant sounding board during the internship and conference buddy after. Speaking of, I am grateful to Satya, Silvia, Leonie, Maik,

Benjamin, Regina, Sian, Mike, Elisa, Max, Kris, and many others that I cannot possibly name here. You made me realize just why people were longing for the return of in-person conferences after Covid. I hope to still see some of you from time to time at future conferences!

To get to the point of starting a PhD, I have to specifically thank Philip Hausner and Philipp Jung, who made the experience of obtaining a Bachelor (and Master's) degree more entertaining than it had any right to be. Dominik Walter, for sharing the experience of being a Computer Science PhD student, despite neither one of us having a clear idea of what the other *actually* does. Matthias, Tamara and Markus for making time whenever I swing by at home, no matter how short-noticed my announcements. And of course all my friends in Heidelberg, for making the city a home over the past ten years!

Of course my gratitude also goes to my family: for firmly supporting my academic endeavors from the very beginning and always welcoming me back home, no matter how far away I have moved. And finally to Maya, who supports me well beyond the PhD and is now finally getting her own shot at it. You give me with all the love and support I could ever hope for.

Contents

1 Introduction

"I am very proud that I am smart enough to get to the point."

Harry S. Truman

Imagine for a second that you are one of the examiners of this thesis: you have about 200 pages of dense writing ahead of you, and have to judge the relevance and contents of this work. To provide a fair and objective evaluation, you will have to continuously build up a mental model of the thesis contents, extend your knowledge by newly introduced concepts, and revisit some of the assumptions and evaluations discussed throughout. This presents an incredibly challenging mental task, which requires multiple re-reads, jumping between different chapters to revisit prior definitions or concepts, and oftentimes jotting down subjective notes of the most central points and your evaluation of those arguments.

As a nod to the less inclined reader and a core motivation of our work,[1] we can provide one immediately useful application of text summarization that assist humans with complex tasks.Figure 1.1 presents a *summary of this particular work*, which has been machine-generated by the system which will be introduced throughout this thesis.[2] Instead of having to manually sift through the 200+ pages ourselves, we may now absorb the central points in the fraction of a time and use the summary as a reference point when reading specific sections to quickly jump to central points of other chapters! In our opinion, this is only one of many areas in which the automatic generation of summaries can greatly improve human productivity and would offer immediate benefits if done correctly.

Reading the summary, however, we can immediately acknowledge several observations about the conceptual complexity of text summarization, which highlights some of the problems that we will address in this thesis:

1. A summary is orders of magnitudes shorter than the original input text, in this case compressing the contents by a factor of roughly 200 times.
2. In creating a summary, we must omit a large portion of the depth and details within this work, but without failing to mention the *central concepts and contributions*.

[1] And a helpful guidance for the examination committee.

[2] The summary was generated given the full text excluding the introduction chapter, which itself can be seen as a partial summary of the thesis contents.

The thesis addresses limitations in current text summarization systems and introduces innovative approaches to improve their practicality and effectiveness. It emphasizes the need for high-quality data, robust models, and reliable evaluation methods as the foundations for successful summarization systems. The author contributes to this field by creating the EUR-Lex-Sum dataset, a high-quality, human-written, and multilingual legal summarization resource available in all 24 official EU languages.

The thesis also introduces a novel metric called SRLScore, which improves the evaluation of factuality in system generations. SRLScore utilizes publicly available software and does not rely on labeled datasets, making it reference-free and widely applicable.

Furthermore, the thesis proposes a hybrid and aspect-based architecture for text summarization, incorporating user preferences or unsupervised extrapolations. It discusses the ex-ante and ex-post aspect categorization, highlighting the wide range of customization parameters that can be incorporated to generate more user-tailored summaries.

In addition, the thesis makes contributions to document-level text simplification by creating the Klexikon dataset, a German resource built from alignments between Wikipedia and a children's encyclopedia. It also discusses open challenges and synthetic approaches to generating segment-level alignments between texts and their simplified versions.

Finally, the thesis explores the use of temporal information in summarization, drawing inspiration from Almasian et al. to define temporal query operators for ex-ante filtering. It also discusses the works of Hausner et al. and Markert, who view timelines as a form of summarization, influencing the generation process by imposing a particular temporal order.

Figure 1.1: A single-page summary of the contents of this thesis (excluding the introduction), generated by the system proposed in subsequent chapters. A full specifications of parameters can be found in Section 4.5.

3. The attentive reader may even notice disparities between the portrayed facts in the summary versus their long-form explanation in the later chapters. This points to a problematic issue of *consistency*, which both human- and system-generated summaries struggle with.
4. Throughout this work, a “summary” purely focuses on textual inputs, ignoring any other modality (e.g., figures, tables, or other forms of media).
5. Depending on personal expertise and prior knowledge, some of the chosen sentences in the example summary may be redundant (or, in other cases, insufficient) to provide appropriate background given the level of familiarity with the topic.

These observations allow us to illustrate the deeper meaning of the opening quote by Mr. Truman: summarization, i.e., the task of “getting to the point”, presents an incredibly complex problem not just for humans, but also machines. With this complexity in mind, we originally set out to investigate the landscape of what is otherwise known as “automatic text summarization” in the field of Natural Language Processing (NLP).

While we have already seen first attempts during the early days of computer science research to tackle the task of letting machines abstract away unnecessary information in scientific texts (Luhn, 1958), we are still not at a point where automated summarization systems have found their way into mainstream applications, despite ever-increasing technological utilization within society.

Simultaneously, apps providing human-generated, digestible snippets of entire books, e.g., “Blinkist”,[3] are constantly gaining in popularity. The company, for example, employs an estimated 200 person staff to *manually generate* these snippets, showcasing a definite economical utility aside from the subjective benefit of a lessened burden in terms of information input.

It is with this juxtaposition of a theoretically useful tool for generating automated summaries and its practical absence that we pose our first central research question in this thesis:

Why do we not regularly interact with automatic text summarization systems?

1.1 Motivation

In an attempt to answer the previously posed question both from an analytical and practical standpoint, the contents of this work will be roughly split into two parts and are primarily driven by the quest for a better summarization system. In the first half, we begin to understand the nuances of current text summarization systems, what practical use cases may be considered, and an empirical discussion of some of the central limitations within the field.

With recent advancements in Machine Learning (ML) and NLP becoming more palpable in everyday life, the natural expectation of many is that such advancements should also naturally trans-

[3] `https://blinkist.com`, last accessed: 2023-08-24.

late to sub-fields such as the summarization community, as argued before. However, the field itself is still struggling to overcome some of the most basic obstacles, such as providing solid ground truth data, and effectively dealing with the automated evaluation of a summary's truthfulness to the original input text.

This issue is only exacerbated when extending our scope beyond English-language texts: in an increasingly globalized world, inclusion must take a leading priority by respecting not only different backgrounds, but also a diverse set of languages. Yet, both in academic research and recently popularized chatbot systems, we still find an overly strong focus on monolingual systems, which are hard (if not impossible) to reproduce in other languages, and oftentimes lack the necessary data to do so. Specifically for summarization, we further lack diversity of resources in their domain origin. Within the last decade, a majority of works have focused exclusively on marginally improving results on one particular domain: news texts. Primarily due to its abundance of available data on the web, but also with its relatively concise (and similar) content structure, this domain has established itself as the go-to for researchers. To allow a broader view of practically useful systems, it is therefore paramount to extend our resources with not just better *systems*, but also *better data* and *a user-centric approach* to system design.

In the second half of this work, we subsequently switch our focus to providing a remedy for some of the aforementioned system-centric limitations, and discuss a generalized framework that can accommodate a broader range of user preferences with no additional changes required to the incorporated systems. This means firstly incorporating some of the user-centric preferences into summarization systems: instead of providing static and generic summaries, our system is able to adjust outputs towards specific backgrounds, and re-write them as necessary. Furthermore, we refrain from training our system on domain-specific summarization tasks (partially in absence of appropriately user-centric datasets), and instead argue for a much more modular architectural design, which allows for improved flexibility in different settings.

1.2 Contributions

We have already teased a number of issues and challenges within the field of text summarization that will be relevant to this thesis. We would like to additionally highlight a more structured list of contributions within the work, and set the expectation for readers at this point:

1. We investigate a series of limitations in existing summarization research, with the usability of such systems in mind. Our findings can be broadly categorized into three categories of errors, namely data-centric issues (limited length of sample texts, quality of samples), model-centric problems (training/evaluating on narrow domains, trivial summaries), and shallow evaluation (e.g., the lack of appropriate metrics and limited analysis of outputs).

2. Regarding model-centric problems, existing research overly focuses on the English language and a singular application domain with news text. We actively work towards a broader applicability of systems to other languages and domains, by introducing two new resources for training and evaluation. With one exclusively German dataset, and another multilingual resource, we provide a test bed for domain-specific summarization systems that exceed traditional document lengths.
3. We further address some quick fixes to improve the quality of existin datasets, by proposing a set of simple heuristic filters to increase the quality of data-related issues. In our empirical analysis on German summarization systems, the filtered datasets indicate a more difficult evaluation setting, and allow a retrospective investigation of model errors.
4. We develop a new metric for the finer-grained evaluation of factual consistency in generated summaries, which poses as a central limitation in the current usability of systems. Our metric has the added benefit of utilizing human-interpretable intermediate representations while performance remains competitive with other state-of-the-art metrics in the field.
5. We further introduce a theoretical framework that differs from previous definitions by accepting a broader range of *subjective* target summaries as ground truth. While the original (psychological) definition of summarization acknowledges the importance of differentiating between user-specific requirements, most of the recent advances rely on simple single-truth gold standards for experimentation and empirical evaluation. Our extended model definition introduces the concept of *aspects*, generally thought of as a subset of singular larger solution, which can be defined as both user-centered points, but also closely related to the structure and content of original texts.
6. We propose a first prototype implementation based on the aforementioned framework, which consists of a two-stage architecture, combining a number of cheap aspect-focused document filters and a generic re-writing module. This modular architecture can handle much longer input documents than previous architectures, while simultaneously allowing for a wider variety in generated responses.
7. We illustrate the challenges of extending the re-writing stage at the example of text simplification and discuss the data- and model-centric requirements for extending generic text-to-text models to particular aspects.
8. Finally, we also present a temporal hierarchy to exemplify the extensibility of individual aspects into more complex document representations.

The remainder of this work is structured as follows. We begin with a formal introduction to the task of automatic text summarization in Chapter 2 and further illustrate some of the prominent algorithms developed over time. Particularly, we outline how practical usability concerns are currently absent in many research works. In Chapter 3, we analyze some more pressing limitations of existing summarization system, particularly surrounding the nature of data and evaluation processes, as well as non-English summarization settings. Chapter 4 then introduces a light-weight and customizable two-stage framework that we believe can alleviate some of the existing issues regarding the incorporation of a more user-centric generation of summaries. We further present a first prototype implementation of our approach, which we have showcased earlier in this introduction. To complement our high-level architectural overview, we continue with extensions of the generative layer in Chapter 5. This includes the introduction of a multi-aspect resource for German text summarization, as well as potential alignment strategies for document-level paraphrasing. On the retrieval side of our model, Chapter 6 exemplifies the extensibility of our model, by introducing a more complex hierarchical temporal representation of time stamps, and the subsequent application to retrieval scenarios. We ultimately conclude with a short overview in Chapter 7, where we also discuss some of the remaining open questions and avenues for future work.

2 Background and Related Work

"If you can't explain something to a first year student, then you haven't really understood."

Richard P. Feynman

The task of "summarization" can be highly ambiguous without further formal specification, as already becomes apparent in the previous chapter. To have a solid foundational framework on which reasonable assumption can be mathematically expressed, we formalize the task of text summarization in Section 2.1: starting from a generic modeling point, we introduce the notion of a flexible *multi-document* summarization system, which is primarily concerned with the selection of "relevant" input sentences to generate a summary from multiple input sources. Subsequently, Section 2.2 will extend this generic framework to distinct variants thereof, such as single-document systems (probably the most prominent architecture in current research), and, more importantly, the distinction along a more functional axis. Systems can be generally separated into *extractive* summarizers, which simply extract a particular subset of text elements from the input text, therefore being more robust to systemic failures (more on this in Chapter 3). Aside from this, *abstractive* systems have recently emerged as a second type of approach, which re-interpret the text and give a formulation as output that not necessarily repeats the reference input. Finally, a combination of the two approaches is generally referred to as a *hybrid* summarization system, which will be used as the basis for our own contributions towards more flexible summaries in Chapter 4.

We continue in Section 2.3 by addressing some of the key needs a summary has to address; in particular, there is a clear psychological motivation for information needs when it comes to human evaluators. Given that the main objective of this work is to re-calibrate summarization research with a more specific focus on the usability from a user perspective, we further include an analysis of the different evaluation dimensions used in prior works on the topic. These will serve as a backdrop for the analysis of limitations within summarization systems in further chapters.

In addition to the formal view, Section 2.4 gives a concise introduction to relevant methods in the area of text summarization. This involves some of the most popular algorithms for extractive text summarization at the time of writing, but also a series of baselines that are intended to put model performance in a more comprehensive light; finally, we also briefly introduce the currently employed tools for automatically evaluating summarization quality – which gives a first glimpse at the discord between idealized evaluation settings and the practically used tools.

We conclude this introductory chapter with Section 2.5, briefly discussing the implications of particular modeling choices for domain-specific use cases of text summarization systems. Ultimately, a large number of current research focuses on a single domain, namely the summarization of singular news articles. This trend further complicates the holistic evaluation of summaries and the underlying systems; in this work, we aim to incorporate modeling choices that are particularly relevant for less exposed domains, such as legal or medical use-cases.

2.1 A Formal Model of Summarization

In previous work, summarization systems are designed as a singular "black box", which is fed with input documents and somehow generates output texts (summaries). In order to analyze this closed system in more detail, we need to formulate a set of axioms that generally hold true for any summarization system. This requires a formal setting that is often vastly simplified (or ignored altogether) in prior work.

To begin with, we define a **document collection** as a set $\mathcal{D} := \{d_1, ..., d_m\}, m \in \mathbb{N}$, consisting of one or more **documents** d_i. Generally, documents in the collection are referred to as "source documents" or "reference texts" throughout the remainder of this work, with $\mathcal{D}$ being occasionally called the "reference set". $|\mathcal{D}| = m$ expresses the cardinality of $\mathcal{D}$, i.e., the number of source documents available.

As for a single document $d_j \in \mathcal{D}$, it simply consists of one or more ordered **segments**

$$D_j := [t_1^j, ..., t_n^j], n \in \mathbb{N} \tag{2.1}$$

. Without further specification, we can think of segments as any naturally delimited (and usually disjoint) text element. This includes, but is not limited to, paragraphs, sentences, or individual words (tokens). More granular text blocks, e.g. paragraphs or sentences, can further be split into smaller units themselves, e.g., individual words. We generally try to avoid such a recursive representation of a document and, if not explicitly stated otherwise, assume the granularity of either *sentences* or *tokens* for the remainder of this work. We similarly define the cardinality of a single document as its length, $|D| = n$. Importantly, the length may be defined at different granularity levels as well. Throughout the later chapters, we consider the following segment-level distinctions for calculating length: $|\cdot|_{char}$, referring to the total number of characters in a sequence (including whitespaces, punctuation and special characters, etc.). $|\cdot|_{token}$ instead utilizes the level of individual words (also referred to as *tokens*) for measuring the length. And finally, we can also have segment-level length measures, such as $|\cdot|_{sent}$ for the number of sentences in a document, etc. If not specified further, we assume that the particular choice of granularity does not matter for the respective implementation, and can be chosen at the user's discretion.

Notably, there are often multiple ways in which segments can be "tokenized", resulting in a separation of individual tokens with slightly different results. For the sake of simplicity, we assume in this case that a perfect tokenization can be obtained for any input text, although this is not necessarily the case in practical scenarios. One exemplary problem during the segmentation into words is the non-trivial task of deciding whether a particular symbol (e.g., the period symbol ".") belongs to a token or not. Take, for example, the distinction between a sentence-concluding period such as the one at the end of this sentence, versus a symbol belonging to a word, such as it is the case for abbreviations like "Dr.". These and similar edge cases cause the tokenization to be highly ambiguous and tools still suffer from the occasional problem, even in highly researched languages such as English or German. We refer to the set of distinctly occurring tokens in an input document collection $\mathcal{D}$ as the **vocabulary** V, formally defined as

$$V_{\mathcal{D}} = \{t \mid t \in d_i \land \, d_i \in \mathcal{D}\}. \tag{2.2}$$

Coming back to the level of a document collection, we must note one particular distinction: the fact that its contained documents are inherently considered as an unordered collection, and are therefore permutation-invariant. On the other hand, this implies that document collections consisting of the concatenation of documents are distinctly different from the collection of individually retained documents, i.e.,

$$\mathcal{D}_{seq} := [d_1 \oplus ... \oplus d_m] \neq \mathcal{D} = \{d_1, ..., d_m\}. \tag{2.3}$$

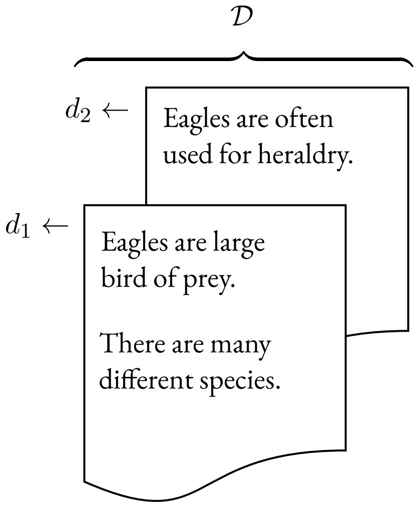
The concatenation $\oplus$ is defined here as a sequential merging operation, expressed as

$$d_i \oplus d_j = [t_1^i, ..., t_m^i, t_1^j, ..., t_n^j]. \tag{2.4}$$

With this, a **summary** can now be defined as a finite subset of the original document collection. In order to consider all potentially generated summaries from $\mathcal{D}$, we introduce the superset of all segments across documents, also known as $\mathcal{S}$.

$$\mathcal{S} := \{t_i^j \,|\, t_i^j \in d_j, d_j \in \mathcal{D}\} \tag{2.5}$$

It is important to note that – quite like the document collection itself – the superset $\mathcal{S}$ is a *multiset* and therefore retains duplicate segments or documents (i.e., segments appearing in distinctly different documents D_i and D_j will be represented accordingly). However, within $\mathcal{S}$, there is


$\mathcal{D}$
$d_2 \leftarrow$
Eagles are often used for heraldry.
$d_1 \leftarrow$
Eagles are large bird of prey.
There are many different species.

$$t_1^1 = \text{[Eagles, are, large, bird, of, prey, .]}$$
$$t_2^1 = \text{[There, are, many, different, species, .]}$$
$$t_1^2 = \text{[Eagles, are, often, used, for, heraldry, .]}$$
$$\mathcal{S} = \{t_1^1,\, t_2^1,\, t_1^2\} \qquad |\mathcal{S}|_{sent} = 3$$
$$|\mathcal{D}| = 2$$
$$|d_1|_{sent} = 2$$
$$|d_1|_{token} = 13$$

Figure 2.1: Example illustrating the notation of our document model. For this collection, we assume a segment level of individual sentences, and two input documents. We further assume naturally segmented tokens, i.e., a human-like interpretation of a single "word".

similarly *no fixed order between documents* (or even between segments) retained; we will address this particular issue in more detail later on. Similar to a single document D, we can define the cardinality of a document superset $\mathcal{S}$ as

$$|\mathcal{S}| = \sum_{d \in \mathcal{D}} |d|. \tag{2.6}$$

The size of $\mathcal{S}$ also differs from the cardinality notation of the document collection, $|\mathcal{D}|$, which expresses the *number* of documents, rather than their cumulative length, as it is the case for the cardinality of $\mathcal{S}$. Figure 2.1 illustrates the notation at the example of two (very short) documents in a collection. Note specifically that we fixate two distinct parameters in this example: first, the segment level is set to sentences (it could have likewise been set to paragraphs, or sub-sentence units, etc.), and the tokenization specifically uses a naturally delimited word as a unit (instead of a simpler whitespace split, or more complex subword unit splitting (Sennrich et al., 2016)).

Building on this relatively generic definition of a document collection, we can now formulate a first naive definition of an (extractive) summarization system, referred to as "*summm*". Given a document collection $\mathcal{D}$ and its associated superset $\mathcal{S}$, we formalize $summm$ as a function

$$summm : \mathcal{D} \rightarrow \mathcal{S}, \tag{2.7}$$
$$summm(\mathcal{D}) = s := [t_i^j, t_k^m, \ldots],\ t_i, t_k \in \mathcal{S}. \tag{2.8}$$

In other words, a summarization system will generate a **summary** s, represented as an (ordered) subset of segments from the document collection $\mathcal{D}$. In more practical terms, it can be assumed that $|s| \ll |\mathcal{S}|$, i.e., the number of segments present in a generated summary should be significantly smaller than the combined length of the input documents.

Systems generate, however, not just a random sub-selection of segments in $\mathcal{D}$ – rather, they should optimize the perceived text quality of a summary s with respect to $\mathcal{D}$. For the sake of simplified notation, we refer to the collective optimization across all desirable evaluation dimensions as *information density*.[1]

One problematic assumption present in this naive definition is the implication that summaries are equivalent for *every user*, irrespective of their individual prior knowledge or information needs. This knowledge prior may shift the information density of a particular generated summary, which in turn violates the previously stated assumption about a summary independent of a user. However, we have already established that this may be one of the primary reasons that summarization systems see little practical use right now. The implicit need to adjust outputs for individuals is in fact one of the key contributions that will be presented in Chapter 4, focusing on the proposal of an aspect-focused summarization model.

Additionally, it can be noted that $summ(\mathcal{D})$ does not directly express the disjoint nature of segments in a summary: specifically, it may be asserted that no two segments $t_i, t_j \in s$ should have the same *semantic meaning*. As an example, imagine a syntactically different sentence, differing significantly in its wording from t_1^1 in Figure 2.1:

Eagles are a predatory species of bird.

However, while the two sentences use different vocabulary, one would still want to assign these two sentences as "highly (semantically) similar", as their meaning is preserved even with syntactic changes.

Regarding a summary, we instead want the selected sentences to be *semantically diverse*, i.e., covering a broad range of information content while keeping the number of selected sentences minimal. This requirement of diversity is equivalent to the previously mentioned notion of information density, which is why one can construct a more precise requirement for generated summaries by defining a function of **semantic similarity** between two segments $t_i, t_j \in \mathcal{S}$ as

$$\text{sim} : d \times d \to \mathbb{R}. \tag{2.9}$$

[1] Arguably, some of the earlier summarization systems do a better job of maximizing this objective than more recently developed summarization systems, see, e.g., Goldstein and Carbonell (1998). The lack thereof is related to the transition towards entirely gradient-based training routines.

The resulting similarity score can be interpreted that a higher score implies a closer semantic similarity between two segments. In reality, scores are also limited to a fixed range, e.g., the closed interval of $[-1, 1]$ for cosine similarity, a commonly used function when comparing semantic vector representations of two segments. Another reason for the range limitation is the satisfaction of the "self-similarity guarantee" between segments. Here, the similarity function sim has to guarantee that

$$\text{sim}(t_i, t_i) \geq \text{sim}(t_i, t_j), \text{where } t_i, t_j \in \mathcal{S}, t_i \neq t_j. \tag{2.10}$$

Aside from the self-similarity, we generally *assume* any similarity function to be symmetric, i.e., $\text{sim}(t_i, t_j) = \text{sim}(t_j, t_i)$. In fact, one can observe practical implementations violating the symmetry assumption quite frequently. E.g., the computation of "similarity" through neural networks is not per se symmetric. This is primarily based on the fact that "similarity" may be computed from a segmented input text `<start>` t_i `<separator>` t_j `<end>`, where positional biases and latent interactions cause a breakdown of symmetric representations.

Importantly, a desirable similarity function should be extensible to express relations between multiple segments as well, e.g., $\text{sim}([t_i..., t_j], [t_k, ..., t_l])$. This allows then for the measurement of similarity between a summary s (with fixed order of segments in s) and the segment superset $\mathcal{S}$. In practice this would likely be a sequential representation of the document collection $\mathcal{D}_{seq}$ instead, as the non-ordered representation of $\mathcal{S}$ could be ambiguous.

Our naive summarization system *summ* can then be expressed as an optimization problem, where the summary is chosen such that the maximum similarity between the original document collection and the summary is achieved:

$$\textit{summ}(\mathcal{D}) := \operatorname*{argmax}_{s \in \mathcal{D}} \text{sim}(s, \mathcal{S}). \tag{2.11}$$

Problematically, Equation (2.11) states that the argmax in the naive case would simply yield $\mathcal{S}$ as the "optimal" result – which obviously constitutes a poor "summary". Instead, we may reformulate this statement as a constrained optimization problem, where the maximization of (content) similarity is subject to a minimization constraint on the length of the summary instead:

$$\begin{aligned} \textit{summ}(\mathcal{D}) := &\operatorname*{argmax}_{s \in \mathcal{D}} \text{sim}(s, \mathcal{S}), \\ \text{subject to} \quad &\min |s|. \end{aligned} \tag{2.12}$$

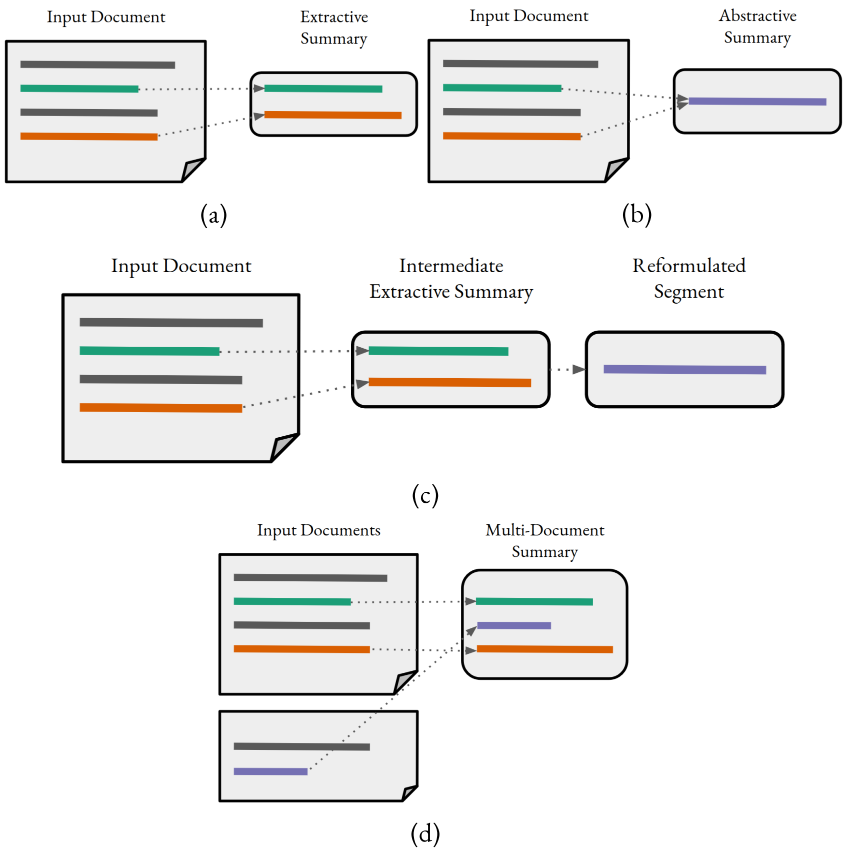


Figure 2.2: Illustration of various automatic summarization system variants. (a) Extractive summarization provides summaries in the form of copied text content, usually at the segment level. (b) Abstractive summarization systems may re-write or combine information in the generated summary and are thus not restricted by input document content. (c) Hybrid summarization systems couple an extractive and abstractive stage, to reduce computational load on the re-writing module. (d) Multi-document summarization systems extend the previous concepts to accommodate multiple input documents at the same time. Combinations with other variants exist.

Importantly, we reiterate the distinction from human preferences of an "ideal" summary; particularly, the optimization for constrained length may conflict with other preferences, such as a need of a cohesive of factually correct summary.

2.2 Variants of Summarization Systems

As expected, only a minority of the current summarization systems actually fall neatly into the previously proposed axioms. In order to accurately represent a wider range of models that are encountered in practice, we adjust summarization frameworks with individual extensions of our formal representation to account for the various model classes. Broadly, we distinguish between the type of extraction performing the actual summarization step, and secondly the size of our input document collection $\mathcal{D}$. We further illustrate the particular differences in Figure 2.2.

2.2.1 Extractive Summarization

The system that closest aligns with the previously given axioms is an **extractive summarization system**. Hereby, the summary s is indeed chosen in such a way that the output text will consist of verbatim copies from the input document(s). Notably, these are particularly cheap in terms of computational complexity, as the copying of text is fairly efficient. However, arriving at the conclusion *which* segments to copy may still take up significant computational resources. The vast majority of works in text summarization pre-dating the neural era (i.e., predating the 2010s) are based on some variant of extractive summarization (Erkan and Radev, 2004b; Mihalcea and Tarau, 2004).

As eluded to in the previous definitions, extractive systems may not return the most coherent generated results. As the inter-segment coherence is not guaranteed, particularly if segments are chosen from different positions within a document (or across multiple documents), it may be hard for a reader to follow what particular concepts are referenced at one point in time. Regardless, given that the extractive models operate on the segment level, it at least guarantees a high *intra*-segment coherence (i.e., grammatically correct text as it appears in reference segments). As an indicator for an extractive summarizer, we write "$summ_{ext}$".

2.2.2 Abstractive Summarization

At the other end of the generation spectrum, **abstractive summarization systems** no longer exclusively extract existing text fragments. Instead, the abstractive part of the system refers to the fact that it may generate text sequences that do *not* appear in the original text. To adjust our formalization to allow for relaxed requirements on the summary s, one can instead consider an underlying vocabulary of a target language T, where $V_T \neq V_{\mathcal{D}}$.[2] Then, the summary s is no longer necessarily consisting of a sub-selection of segments from $\mathcal{S}$, but rather a sequence of m tokens drawn from the vocabulary, or

$$s = [t_1, ..., t_m], t_i \in V_T \forall i \in [1, m]. \tag{2.13}$$

Consequently, an abstractive summarization system can be expressed by $summ_{abs} : \mathcal{D} \rightarrow V_T$. In contrast to an extractive summarization system, an abstractive model has an inherent advantage in terms of outputting (syntactically) fluent summary texts. However, generating – to some ex-

[2] In practice, the vocabulary *may* still be based on the input document collection in some ways; more recently, however, the vocabulary is more likely to be defined by a much larger pre-training dataset.

tent arbitrary – text comes at a high additional cost of modeling the underlying language. Purely because of this fact alone, requirements in (annotated) training data increases by several orders of magnitude compared to an extractive system. Among other reasons, this is a primary cause for English language setups to dominate the current landscape of summarization research: Few languages offer the instant availability of large-scale datasets and related literature that allow for a comprehensive study of summarization (or other language) phenomena. We also investigate the lack of *semantically* coherent summaries, particularly with respect to factually verifiable statements Section 3.5. The fallacy of "trusting" model output can thus lead to detrimental results, including the misrepresentation of original claims due to reliance on summary texts (Fabbri et al., 2021).

2.2.3 Hybrid Approaches

Another limitation of current abstractive systems, which we will analyze in more detail later on, is the limit on overall input text length. Generally, for any sufficiently large $\mathcal{D}$, abstractive models alone will be unable to handle the text load in a single iteration. While splitting input segments and iteratively building summaries is certainly an option to circumvent this problem (Beltagy et al., 2020), the additional burden of high inference cost makes this a less attractive option. Instead, **hybrid summarization systems** have been proposed as an intermediate solution (Liu et al., 2018a; Liu and Lapata, 2019), combining the best of both worlds. Formally, a hybrid system can be represented as the functional concatenation of two individual summarization systems, also referred to as different *stages*, formally

$$s = summ_{abs}(summ_{ext}(\mathcal{D})). \tag{2.14}$$

Notably, the intermediate representation of the extractive summary is then generally considered as a "single input document" (i.e., $|\mathcal{D}_{intermediate}| = 1$). This is no strict requirement, and different hybrid architectures may have varying assumptions about the intermediate representation, depending on the exact method used in the first extractive stage.

The extractive summarizer is generally developed to be efficient for retrieval across potentially large document collections, akin to a "retriever-reader" Question Answering (QA) architecture (Chen et al., 2017), two-stage ranking approaches in Information Retrieval (IR) (Nogueira and Cho, 2019), or Retrieval-Augmented Generation (RAG) (Lewis et al., 2020b). The key differentiation for summarization settings is the abstractive nature of the second "stage" in the process, i.e., the abstractive re-writing of selected text, with the simultaneous shortening of texts. In comparison, the second stages of both the QA and IR setting differ in that they do not perform any text generation. For QA settings, the later stage generally performs a span-selection on "relevant" passages, whereas re-ranking architectures in IR focus on improving ranking order over the naive first-stage approach. Regardless, we can consider the research hybrid architecture settings deeply connected across several sub-fields. Particularly the extractive algorithms for first-stage ranking are oftentimes shared between architectures; popular choices are, e.g., term-frequency/inverse-document frequency (TF-IDF) (Jones, 2004) or BM25 (Robertson and Jones, 1976).[3] In contrast, for RAG systems, the task setting is relatively constrained, and follows a more directional expected output format (explicit input sources, as well as a summary-style output text). As an example, RAG systems may be used to solve a range of tasks, such as QA or generic text generation, without specifically focusing on a summarization objective, while injecting knowledge from a retrieval stage into the generated outputs.

2.2.4 Multi-Document Summarization

While the previously introduced variants have been concerned with differing definitions of the summary generation process, we continue by introducing variance on the data side as well. Starting with **multi-document summarization**, or *MDS* for short, we simply assert that the number of input documents is strictly larger than one, or formally $|\mathcal{D}| > 1$.

Generating a summary from multiple distinct input documents may be particularly challenging for a variety of reasons. First and foremost, information across documents may be duplicated, a prominent issue in the areas of trend summarization from social networks (Ziegler et al., 2021; Campos et al., 2022). Formally, content duplication can be defined as segments $t_m \in d_i$ and $t_n \in d_j$, such that

$$sim(t_m, t_n) > \delta, \tag{2.15}$$

where δ is a chosen similarity threshold.[4] Readers should note that the requirements on input size is in fact complementary to the modeling dimensions; we can therefore talk about, e.g., a “hybrid multi-document system”. Only in some instances are the two modeling dimensions correlated. Particularly for the multi-document setting, abstractive approaches are less popular, as the input length can quickly spiral out of the possible content limit for abstractive models.

This implies a set of particular challenges for the summary generation in multi-document settings:

1. It might have to be ensured that the generated summary does not contain duplicate content, or conflicting opinions are accurately represented even in the summary (which specifically *necessitates* partial repetition).

[3] Formal definitions are given in Section 4.5.2.

[4] Incidentally, this is also an issue for summaries from single documents, where t_m and t_n are from the same document d_i. However, we generally do not assume that such “repetitions” appear in short text documents, as it is the case for most research settings on summarization utilizing news-based datasets.

2. Since individual documents are generally assumed to be semantically self-contained, it is necessary for a multi-document summarizers to ensure *content consistency*, as well as *grammatical fluency* of a generated summary. Especially for extractive systems, this is non-trivial to enforce, given the nature of generated summaries for such a model.
3. Aside from content-based selection criteria, the documents in $\mathcal{D}$ may be arranged on meta-information available, a topic we describe in greater detail in Chapter 6.

2.2.5 Single-Document Summarization

Despite the broadly applicable setting of multi-document summarization, most evaluation settings prefer a simplified setting with $|\mathcal{D}| = 1$, also known as **single-document summarization** (or *SDS*). For reasons explained just before, SDS settings simplify the task setting noticeably, as no cross-document coherence has to be explicitly modeled (thus also easing the process of generating a coherent summary).

A more pragmatic reason for specifically focusing on SDS is the abundance of readily available training and evaluation corpora: while the collection of several relevant documents may require further human interaction, collections for single-document summarization can be solicited in an automated fashion. In many instances, suitable alignments between a single input document and ground truth summaries already exist, e.g., on the internet, and may be (semi-)automatically extracted from these sources without too much expensive annotation necessary. This trend towards the exploitation of available semi-structured annotations in summarization can be attributed largely to the creation of the CNN/DailyMail corpus (Hermann et al., 2015).

While the investigation of how to summarize singular documents is certainly a relevant research question, it has recently dominated the sub-field of text summarization to the point that researchers have forgotten that prominent evaluation metrics where originally designed with multi-document corpora in mind (see Section 2.4.5). When returning to a more user-centric focus in summarization settings, we find that the opposite is the case: users frequently do not even known the specific document from which to summarize in the first place (Giorgi et al., 2022). For this reason, we continue by paying close attention to the way in which human preferences are modeled in text summarization evaluation.

2.3 Assessing the Quality of a Summary

With a formal summarization system now defined, we may establish what exactly would be considered a high-quality summary, as it becomes heavily intertwined with the final usability of a text summarization system – users will strongly prefer summaries that naturally align with their individual needs. Whether a particular length of an output summary, the information contained within it, or the way in which it is written, many factors can be considered as a measure of (subjective) quality for a piece of generated text. With the added difficulty of considering the original piece of reference text, the process of summarization is especially hard to perform even for human users, making it particularly challenging to even properly evaluate the quality of machine-generated snippets. Surprisingly, already some of the earlier literature in the area of automatic text summarization has been addressing the question about how to consistently rate summarization quality, with varying degrees of specificity in their definition for evaluation settings (Jones, 1999; Mani, 2001; Guo and Stylios, 2005). Particularly Guo and Stylios (2005) heavily lean on the roots of text summarization within the area of psychology. In their interpretation, summaries need to be supported by human mental models of information and knowledge, in order to be considered useful. The general takeaway for the purpose of our work can be the explicit need to focus on a *user perspective*, i.e., include explicit evaluation dimensions that factor in different qualitative factors of summarization quality.

Importantly, however, researchers generally agree that human feedback is comparatively difficult to obtain: For reasons that involve both temporal and monetary reasons, explicit annotations are often disregarded in experimental setups, particularly for heavily subjective tasks such as text summarization. Disregarding human feedback in experiments ultimately may lead to a deviation from the original objectives of building *usable* tools, and instead relying entirely on automatic (and somewhat misaligned) evaluation metrics. In addition to the general disregard, we observe additional complexity stemming from differing usage scenarios of text summarization systems, which lead to potentially widely differing requirements in the desired system outputs. As an example, researchers consider both news headlines and radiology report conclusions as a form of "summary", and treat them as somewhat equal targets when designing experiments. In practical terms, the differences between a headline and report summary could not be any larger, with different expectations on the length, topical depth, and formatting of individual segments being some notable aspects.

To clarify what an ideal setting for (productive) text summarization research may look like, we start by giving a brief overview of different quality dimensions introduced in prior works. Furthermore, the most important factors for aspect-based text summarization settings are elaborated in more detail.

2.3.1 Qualitative Dimensions of a Summary

The selected evaluation dimensions are an aggregation between multiple different works, which we deem especially relevant for the focus on a user-centric summarization model (and its evaluation). We particularly encourage readers to additionally read the works by Grusky et al. (2018) and Fabbri et al. (2021), which align closest with our selection of highlighted rating dimensions.

These works are also the basis of several follow-up studies using the data sources introduced in the respective works. Such evaluation dimensions are also used to evaluate Retrieval-Augmentend Generation systems (Es et al., 2024). We further acknowledge influences of these dimensions by several other works in the respective sections, however, they often only look at a chosen subset of the following four dimensions. From a practical standpoint, it should be said that the evaluation dimensions are chosen without being entirely orthogonal. Meaning, a summary that scores well in one area (e.g., the coherence) may also have a high relevance score, but it is not necessarily guaranteed. To our knowledge, no study is available that looks at the correlation between various rating dimensions.

Regarding the setup that is to be evaluated, we may assume that a summary may have been written either by humans or generated by an algorithm; the distinction does not matter with respect to the evaluation. It is, of course, possible to summarize from a single or multiple original input documents. We make no further assumption about the setting, but note that in multi-document settings it is more likely to encounter duplicate content.

For some of the following criteria, a comparison with an existing text is required to judge the quality, whereas other criteria (such as the grammaticality or fluency of a piece of text) can be rated independently. In most research literature, a reference-based setting is assumed, where one or multiple so-called *reference summaries* (also often called *gold summaries*) exist, which is used to ease the mental burden on evaluators, as they have a pre-existing (shorter) text segment to check against. On the other hand, in reference-free evaluation settings, the judgment will be made with respect to the reference text alone. We will briefly discuss the relation between evaluation dimensions and available automated metrics, but refer the reader to later chapters for a more detailed introduction (see Section 2.4.5).

Relevance

We refer to the judgment of the quality of presented content within a summary in relation to its original source text as **relevance**. Initially, we may perceive relevance as a metric that strictly increases with the length of a presented survey, as more (relevant) content can be included. However, relevance judgments may be lower in the presence of duplicate (or overlapping) content being included (Fabbri et al., 2021). As an example, we can imagine writing a literature survey, incorporating the related work sections of multiple academic papers: Summarizing content across different related papers within the same sub-field is likely to yield a large overlap in the referenced works, which would generally be avoided in an ideal summarization setting. Earlier works often incorporate relevance indirectly through token-based precision/recall metrics (Mani et al., 2002), which in turn is a direct predecessor to relevance-based automated evaluation metrics (Lin, 2004). The latter approach also exhibits strong correlation with human relevance ratings (Fabbri et al., 2021).

Grusky et al. (2018), who refer to relevance as "informativeness", further clarify that a summary should focus on the *key points* of the input article, which is similar to the earlier interpretation by the Document Understanding Conference (DUC) organizers, which asks evaluators about content clarity (Dang, 2005).

Factuality

Aside from the relevance of individual text fragments, an arguably more important aspect is whether factual information in the generated text can be supported by the original input. This is generally defined as the **factuality** of a generated summary.[5] Especially in the day and age of purposefully spread misinformation or "fake news", it becomes increasingly important to assess how well systems are able to cope with the faithful representation of input documents. For the summarization of multiple input documents, we encounter the additional problem of representing *potentially conflicting information*. While this poses a significant challenge even for human evaluators, we argue that this is beyond the scope of the current work. However, active efforts exist in related fields towards fair representation of conflicting views (Jin et al., 2016; Jang and Allan, 2018). We generally see strong evidence for the dominance of extractive systems over abstractive summarizers when it comes to the evaluation of factuality (Grusky et al., 2018; Fabbri et al., 2021). Particularly the problem of content not appearing in the original input documents – so-called hallucinations – are a frequent cause for low perceived factuality (Ji et al., 2023). We note that extractive models are no guarantee for perfect factual summaries, either, mostly due to poor content selection or intra-sentence phenomena, such as unresolved anaphora or co-reference mentions (Zhang et al., 2023).

Coherence

Where the previous metrics are measuring the semantic accuracy of a summary, coherence is more concerned with the syntactic accuracy. This dimension is heavily focused on the system-level, evaluating the overall structure of a (generated) text at the level of an entire (output) document. Evaluators judging coherence are usually asked whether systems are able to generate cohesive text segments that fluently lead from one topic to another, without seemingly random jumps between topics. Particularly abstractive models have shown dramatic improvement in coherence scores, having the increased ability to merge and re-formulate existing sentence structures (Fabbri et al., 2021). An important work detailing the power of recent automated evaluation metrics for system-level coherence is by Steen and Markert (2022), which extend the more simplistic definition by

[5] Confusingly, this is indeed referred to as *relevance* by Grusky et al. (2018); we exclusively use the term *factuality* or *factual consistency* for the remainder of this work.

Fabbri and collaborators. They find that most metrics are still unable to capture full document-level context, instead often only evaluating coherence at the level of individual segments.

Fluency

Within a summary, the correct syntactic use of language may be variable, and high-quality individual sentences may well be as important for a "good" generation result as the overall system-level coherence. For this purpose, the fluency can express exactly this fine-granular evaluation of the grammaticality of a prediction within a single sentence (or across shorter segments). This is particularly challenging to maintain when the original input documents contain frequent references between different sentences, such as co-references or the use of anaphora, but also requires a system to at least be able to coherently express individual sentences. While we generally may assume that extractive summaries score well in the faithfulness department, fluency may be more affected (Zhang et al., 2023). The aforementioned issues of missing co-reference/anaphora resolution steps can cause a semantic breakdown in a heavily condensed summary, in the worst cases to an active falsification. Particularly more recently proposed generation evaluation metrics building on heavily pre-trained generic language models are increasingly able to identify syntactically correct sentences by evaluating model perplexity of individual sentences (Zhang et al., 2020b; Thompson and Post, 2020; Yuan et al., 2021)

2.3.2 Evaluating Summaries

Aside from the challenges of *what* to ask about in summarization evaluation, the *how* of asking evaluators to rate generated text is almost as important; we refer to Ter Hoeve et al. (2022) for a more comprehensive analysis and practical guidelines of how to design a high-quality evaluation study. In general terms, recent progress in the Natural Language Processing (NLP) community has lead to a wider acceptance of the fact that singular ground truth labels may be insufficient to represent task-specific annotations (Abercrombie et al., 2022; Plank, 2022). For summarization, we expect this to be the case as well, given the conflicting definitions of what makes a summary "good". This problem is only exacerbated for settings that require task-specific insights; in some instances, almost no agreement can be found between the annotations provided by novices and task experts (Fabbri et al., 2021). Such insights are worrisome, as the majority of annotations is collected from platforms populated mostly by task-specific novices.[6] Another fallacy in evaluation is that small changes may have a huge penalizing impact on the summary quality. This is especially

[6] As a "late-breaking addendum", we also reference insights from the study by Veselovsky et al. (2023), who find that an increasing number of MTurk annotations are likely generated by systems such as ChatGPT. This creates a further spiral of deteriorating data quality.

relevant for evaluation of factuality, where minor changes in the generated text can dramatically change the faithfulness of a summary, e.g. the birthday or name of a person.

In the same vein, we may also address the frequent use of Likert rating scales (Likert, 1932), as it cannot be omitted how the particular choice of evaluation scales may affect the observed subjectivity in results. While Likert-style ratings generally offer a reliable (and consistent) way of judging quality differences different annotators' ratings (Croasmun and Ostrom, 2011), there are shortcomings when comparing the scores with longer-form explanations of particular ratings (Hu et al., 2018). In the context of text summarization, Tang et al. (2022) recently investigated the difference between Likert-style ratings and preference ratings (i.e., simply indicating whether a particular output A is preferred over an alternative B). Though it should be said that preference-based ratings are not free from issues, either. Particularly for different evaluation dimensions, human annotator consistency seems to be more erratic (Steen and Markert, 2021), and annotators are usually not given a choice to prefer "neither of the options" as an output. The latter is especially relevant for text summarization, as low-quality outputs are still a frequent occurrence, and as such should be able to be marked as "insufficient", even when comparing the output of two systems. Simultaneously, strength of preference is not (necessarily) considered for binary ranked ratings; especially for quantifying the level of improvement, it is therefore not ideal to use such ratings.

2.4 A Brief History of General-purpose Text Summarization Research

Before setting out with a more holistic exploration of the area of text summarization, we want to divulge the reader in a brief introduction to cover some of the central works throughout the field's history. Unlike the wider field of Natural Language Processing, which is primarily influenced by the Computational Linguistics community, text summarization also finds its origins in a related field, namely the library sciences. Similar to the history of Information Retrieval, early results were primarily motivated by a more systemic approach to library management, including the efficient access of information in large document collections.

Probably the first notable work on summarization of automated abstract generation is Luhn (1958), who already recognized that word centrality, i.e., the importance of a singular segment for the overall document meaning, can be used as a quantification–and subsequent ranking–of text segments.[7] From a vector of segment centrality scores, a "summary" can be easily constructed by choosing the particular highest-ranking segments. While we do not explicitly cover Luhn's

[7] Fun fact: Luhn's paper opens with a sentence familiar to the common summarization researcher: "*Th[e] widespread problem [of fast access to information, Ed.] is being aggravated by the ever-increasing output of technical literature.*" It seems little has changed in over sixty years.

method in this work, it has proven to be a central contribution to the early days of the field. In addition, later contributions by IBM, where Luhn worked at the time, saw the company cemented as the de-facto intellectual leader of the field (Resnick and Savage, 1960; Rath et al., 1961). Few works are worth noting in the broader context throughout the following decades; a notable exception is the work by H.P. Edmundson, who conducted a series of studies on "automatic abstracting" throughout the 1960s (Woolridge Inc., 1961; Edmundson and Wyllys, 1961). He was also the first to our knowledge to actively discuss limitations and problems in the area of text summarization (Edmundson, 1964). We can also observe some first rather theoretical papers appearing at academic conference, surfacing during the early 1980s (Fum et al., 1982; Marsh et al., 1984). The absence of further progress cannot be fully explained, but we may reason that the probable lack of computing power for such complex tasks as summarization played a large role.[8] During the 1990s, Karen Spärck Jones contributed seminally to the area, instigating (among other things) the 1993 Dagstuhl seminar on summarization (Endres-Niggermeyer et al., 1995). A thoughtful review of her further contributions can also be found in (Maybury, 2005). By the end of the 1990s, the field saw the arrival of more academic interest, and consequentially some first attempts at consistent evaluation of summarization systems (Mani et al., 2002). This culminated in the series of workshops hosted at the Document Understanding Conference (DUC), which established the first formal track for summarization and ran between the years of 2001 and 2006 (Over et al., 2007). Here, we have arrived at what can be considered the "new age of text summarization", which will be too much to cover in detail. Instead, we individually examine some of the primary contributions since the early 2000s. We begin by analyzing key algorithmic concepts, followed by a brief review of available (English) evaluation resources, as well as an overview of relevant automatic evaluation metrics.

2.4.1 Extractive Summarization Algorithms

In no particular order, we introduce a number of central algorithms and baselines that we will continuously use throughout this work. Partially, these make use of statistical information hidden inside the training data, such as lead sentence-based methods, but also approaches that satisfy the basic "test of time", and are still used today. Note that all of the following systems are extractive approaches, and as such are only covering a part of the wider progress in literature, which we will analyze in more detail later on, see Section 2.4.2. We broadly distinguish the introduced extractive methods into graph-based (TextRank and LexRank), frequency-based (SumBasic), and heuristic (Lead-3/k) approaches. All of the discussed methods have the advantage of being entirely *un-*

[8] Circling back to the work by Luhn (1958), his system was implemented on an IBM 704 machine, which came with a total of 18,432 *Bytes* of main memory, see `https://en.wikipedia.org/wiki/IBM_704`, last accessed: 2023-07-25.

supervised, meaning they do not require any additional stochastic gradient-based learning step, which in turn would require annotated data. At most, the methods

TextRank

A central method, popularizing the use of graph-based approaches to keyphrase extraction as well as summarization, is TextRank. Introduced by Mihalcea and Tarau (2004), with similar ideas presented by Erkan and Radev (2004a) at the same time. It introduces an algorithm to compute summaries from a graph-based representation of input texts. In particular, each vertex of the graph represents words or sentences, which are then weighted with iterations of the Pagerank algorithm (Page et al., 1998) over the graph. This constitutes the computation of Eigenvector Centrality (Bonacich, 1987), where the mathematical relationship is further elaborated by Erkan and Radev (2004b). A final (extractive) summary can then be generated from the top-weighted nodes in the resulting network. Several subsequent works pick up on this idea of graph-based summary generation (Garg et al., 2009; Baralis et al., 2013; Parveen et al., 2015). The key difference between these works essentially lies in the underlying method for generating the final ranking of the graph nodes.

LexRank

With more modern systems available, TextRank and other graph-based methods have largely fallen out of favor as baseline systems. We instead present LexRank (Erkan and Radev, 2004b) as the surviving graph-based approach, which we also frequently use in our own experiments (with some modifications). Similar to the previously mentioned LexPageRank/TextRank, the LexRank algorithm can be broken down into the following steps.

1. Compute a vector representation for each segment,[9] utilizing term frequency – inverse document frequency (TF-IDF) for scoring. Words below a certain score threshold are ignored for efficiency and the resulting values stored in a sparse vector.
2. Based on these vector representations, a similarity between all segments is computed. Again, a threshold is used to sparsify the resulting $N \times N$ matrix $\mathcal{A}$.
3. $\mathcal{A}$ is then interpreted as a form of *adjacency matrix*. It can then be used to initialize a graph, i.e., interpreting the similarities as a form of edge weights between different nodes (the segments).
4. Similar to other graph-based methods, a centrality scoring method is applied to the graph, based on PageRank. The top-weighted nodes in the resulting network are again taken as the summary of an article.

[9] In the original implementation, the authors use sentences.

A major reason for the consistent popularity of LexRank over other methods is the original integration into the MEAD summarization toolkit (Radev et al., 2001), which included convenient pre- and post-processing tools, such as sentence segmentation functions, and re-ranking of top-weighted sentences with Maximal Marginal Relevance (MMR) (Goldstein and Carbonell, 1998), which, for example, allows the automated determination of summary lengths with minimal repetition. For the purpose of our experiments, we rely on a Python re-implementation of LexRank,[10] which does not provide any pre-/post-processing functionalities.

Modified LexRank (LexRank-ST)

LexRank has one inherent advantage over the other presented graph-based approaches to summarization. By relaxing the mathematical assumptions about the underlying centrality computation, LexRank can be adjusted to work with arbitrary metrics that operate on a potentially completely different similarity computation, even for metrics not based on a graph. Particularly because the original LexRank implementation uses a discrete bag-of-words approach, which is fairly susceptible to drastic score changes despite small syntactic adjustments, it is desirable to replace the intermediate matrix of segment vectors with a more robust continuous representation. As such, we utilize a modified variant of LexRank, which uses cosine similarity over segment-level language model representations for the calculation of the self-similarity instead.

Particularly, we use the `sentence-transformers` library (Reimers and Gurevych, 2019), which critically offers model variants suitable for multilingual applications (Reimers and Gurevych, 2020).[11] Importantly, however, LexRank assumes that the matrix of computed similarities represent a *stochastic matrix*. This requires the assertion that all entries of the similarity matrix $\mathcal{A}$ are *non-negative* and each row in the matrix represents a probabilistic distribution of the transition probabilities in the Markov chain associated with $\mathcal{A}$. To this extent, it may be required to adjust the (possibly negative) cosine similarity scores, e.g., by re-normalization.

A final critical choice is the determination of appropriate output lengths. For LexRank-based approaches, we differ in our approach, by either using an oracle-informed approach (Aumiller and Gertz, 2022a) or a pre-estimated length based on available training data (Aumiller et al., 2022b). For the oracle variant, we simply extract the same number of segments as present in the gold summary. This presents a pseudo-optimal solution, given that we have an optimal trade-off between precision/recall in n-gram-based metrics (cf. Section 2.4.5). Given that we do not have this information in "blind test settings", we instead opt to adopt a metric based on the average *compression*

[10] `https://github.com/crabcamp/lexrank`, last accessed: 2023-07-24.

[11] This is also one of the rare instances where we are able to attribute a tweet as the relevant reference, since this approach was originally suggested by Nils Reimers himself. See `https://twitter.com/Nils_Reimers/status/1488213682236661774`, last accessed: 2023-04-13.

ratio (Grusky et al., 2018). We define the mean *compression ratio* CR of the samples in a training corpus as

$$CR(\mathcal{S}, s) = \frac{|\mathcal{S}|_{token}}{|s|_{token}}, \tag{2.16}$$

in order to estimate an approximate number of segments in the target summary. We alternatively use implementations in later parts of this work that operate over the *character* level, but functionally stay the same otherwise. By defining the average training set compression ratio as

$$CR_{avg}^{train} = \frac{1}{N} \sum_{i=0}^{N} CR(\mathcal{S}_i, s_i), \tag{2.17}$$

we are further able to approximate the average individual target lengths. Let $|s_i^{test}|$ be the length of to the i-th test reference summary. To compute the estimated target length for this sample, we define

$$|s_i^{test}| :\approx \frac{\mathcal{S}_i^{test}}{CR_{avg}^{train}}. \tag{2.18}$$

Of course, the assumption that most articles follow a similar compression ratio breaks down once we consider datasets with a huge variance in the target lengths.[12]

SumBasic

Vanderwende et al. (2007) introduce a method that entirely relies on word frequency statistics, but does not construct the intermediate representation of a text graph. SumBasic shines by being explainable in five steps, without additional requirements. This makes it an ideal baseline candidate and a frequent implementation, but also showcases how automated systems do not necessarily reflect the true user-centricity required to return subjectively useful information. Given an input document collection $\mathcal{D}$, its associated vocabulary $V_{\mathcal{D}}$ and document superset $\mathcal{S}$, SumBasic computes a summary as follows (Vanderwende et al., 2007):

1. Each token t has its occurrence likelihood assigned as $p(t) = \frac{\text{freq}(t)}{|V_{\mathcal{D}}|_{token}}$, where $\text{freq}(t)$ counts the occurrences of t in $\mathcal{S}$.
2. For each segment d in the input collection, assign a score based on the average word likelihood as $\text{score}(d) = \frac{1}{|d|_{token}} \cdot \sum_{t \in d} p(t)$.
3. The highest-scoring segment d_{best}, *containing the most frequently occurring word*, is added to the intermediate summary.

[12] Or for domains where the output length is fixed to a particular parameter, see lead-3 as a baseline for news articles.

4. To discount previously selected content (and sentences), adjust the probabilities of all tokens present in d_{best}, by computing $p(t) = p(t) \cdot p(t), \forall t \in d_{best}$.
5. Repeat steps 2.-4. until the desired summary length is reached.

Similar to our LexRank baseline, SumBasic only requires an existing tokenizer/segmenter for any particular language, but otherwise has no requirements and should be fairly flexible to apply to multilingual datasets.

Lead-3 and Lead-k

One of the most frequently used baselines in the literature is *Lead-3*. This approach simply takes the first three sentences of the reference text as the corresponding "summary". This specific approach achieves comparatively high scores on news-like articles Nallapati et al. (2017), which is the primary cause for it being dominantly referenced as the primary "naive" baseline. Interestingly, similar variants have already been introduced much earlier, including a variation by Resnick and Savage (1960). The authors use a combination of the first and final 5 percent of a technical report as its "auto-abstract" and observe reasonable empirical performance.

While three sentences are an appropriate length estimate for news articles (especially given the currently used datasets), it may be insufficiently long for other domain applications. To this end, we propose a variant which we title lead-k, instead taking the first k sentences of an article as its summary. Empirically, this provides a stronger baseline on Wikipedia data (Perez-Beltrachini and Lapata, 2021; Aumiller and Gertz, 2022a), but also for legal applications (Glaser et al., 2021b; Aumiller et al., 2022b). The main difference between approaches usually lies in the level used for length estimation (token or sentence length considered?), or alternatively in the estimation of the parameter k, similar to our mentioned approach for the modified LexRank algorithm.

2.4.2 Abstractive (Neural) Summarization

Compared to the extractive nature of most traditional summarization algorithms, abstractive variants are practically synonymous with the recent advent of neural networks. In fact, searching for the term "Abstractive Summarization" before the year of 2013 reveals only a singular quote by Erkan and Radev (2004b): "In fact, truly abstractive summarization has not reached a mature stage today." More than a decade later, Rush et al. (2015) were the first to transfer an attention-based recurrent neural model (Bahdanau et al., 2015) to the task of text summarization. Since then, we have seen ways to optimize training on larger data collections (Vaswani et al., 2017), allowing for an entirely different approach to model "recycling".

Recurrent Models Where extractive methods rely largely on a deterministic set of steps to formulate a summary, the training process of abstractive summarization systems relies largely on stochastic gradient descent. For any neural network-based approach, the training objective directly optimizes (or stochastically approximates) the conditional likelihood of next words, i.e.,

$$\operatorname*{argmax}_{\theta} p(t_i | T_{<i}; \theta). \tag{2.19}$$

Here, t_i refers to a single token in a generated sequence, and $T_{<i} = [t_1, t_2, ..., t_{i-1}]$ to the sequence of all tokens prior to the i-th one. Finally, θ expresses the (adjustable) parameters of the neural network. For a more formal view of optimization criteria beyond the scope given here, we encourage readers to have a look at the work by Goodfellow et al. (2016).

Notably, the training objective implies that any optimization is purely done at the level of *predicting subsequent tokens*, and does not allow for a larger, segment-level coherence optimization. We particularly invite readers to think about the consequences this has on the evaluation criteria we previously introduced in Section 2.3. The metrics there largely tie themselves to the document level of a generated summary, and (with the exception of fluency), rarely boil down to the much narrower token-level context. This problem inherently stems from the fact that optimization objectives need to be differentiable within the network architecture, which is possible for individual tokens, but less so for document-level targets. As such, there is no simple solution to "bake" the expected metrics into the training procedure, which is particularly troublesome.

The spike in attention on such particular neural architectures may also be partially caused by the increased availability of large-scale training resources, at least for English (cf. Section 2.4.4). While RNN architectures (later to be mostly replaced by the Long-Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997)) were a big step-up from previously achieved state-of-the-art methods, they soon proved to be limited in their own respective ways. The inherently recurrent nature of their design limits the training efficiency, and the particular implementation of recurrence makes long texts difficult to represent with stable gradients (Pascanu et al., 2013).

Transformers, Pre-training, and Fine-tuning The particular trend in the early neural boom era between 2013-2017 included largely training models from the ground up, discarding any previously spent computing time on models trained for different tasks. Aside from the obvious inefficiency in model re-use, this can be largely attributed to the lack of empirical understanding on training dynamics, as well as the prohibitively expensive (and thus slow) updates of LSTM-based models. The limitation here comes from the recurrent nature of LSTM units; the derivation of a gradient has to propagate "back in time" through all prediction steps, where usually individual steps are mapped to tokens. This non-linear dependency chain of computation is difficult to map linearly on physical hardware and thus hard to scale up (Vaswani et al., 2017).[13] As

[13] There are further practical concerns, such as the requirements on batches of sequences having the exact same length, that are partially alleviated by available model implementations.

an alternative, Vaswani and colleagues propose a model architecture that shifts from a sequential computation to a parallel implementation via inter-connected individual layers, set to a separate encoding and decoding unit. Their suggested architecture, the Transformer block, also proposes using a fixed-length "context window" and forcing sequences to either truncate to the window length, or otherwise pad with "empty text". These two changes in combination allow the training with much greater efficiency for large amounts of data, while also showing promising SotA results on individual tasks.

On top of the architectural departure from prior work, in early 2018 approaches emerged with the idea of using a *broader, much more generic* approach to Natural Language Processing. This is not unlike a similar development in neural Computer Vision research, where existing model weights are taken and adapted to other tasks than the original training objective (Razavian et al., 2014).[14] For Natural Language Processing, this concept naturally extends to models that can perform several text-based tasks at the same time Peters et al. (2018), potentially also including summarization (Howard and Ruder, 2018). This basic idea of "Transfer Learning" is a first attempt at addressing the previously mentioned inefficiency in training models from scratch. However, since the exact nature of each task is varying, it often requires task-specific data. Transfer Learning now describes a framework to train a model not from scratch, but rather some previously trained network, and adjust the weights such that the model can perform on the "transfer task" as well, without requiring nearly as much task-specific data. This is not to be confused with multi-task learning, where the objective is to train on multiple different task *at the same time*. Such setups do similarly benefit tasks where annotated data is hard to come by, and training on a smaller dataset would not work individually, see, e.g., Liu et al. (2016).

In an attempt to make the initial model as capable and flexible as possible, the so-called *pre-training* stage in a Transfer Learning should scale to the largest possible number of pre-training samples. Since human annotation of such large amounts of data is generally prohibitively costly, the usual choice is to collect samples and design a very generic objective, for which no manual annotation is required. This extends the idea behind modern word embeddings, where Mikolov et al. (2013) introduce a similar automatically derived task objective in their landmark word2vec paper. For generative pre-training, the is to focus on a model's ability to generate *language*, by reconstructing a partially obfuscated piece of text (Devlin et al., 2019; Raffel et al., 2020), or an auto-regressive (i.e., one-sided) next token prediction task (Radford et al., 2018). For the sake of conciseness, we will only briefly consider the latter, as defined by Radford et al. (2018).

[14] Despite these rather recent developments, Transfer Learning as a concept is another hot topic that dates all the way back to the 1970s, see Bozinovski (2020).

Given a document $D \in \mathcal{D}$, we consider the task of next token prediction by proxy of maximizing the log-likelihood of the current token, given a (limited) context of the previous tokens. Formally,

$$L(d,k) = \sum_{i=0}^{|d|_{token}} \log P(t_i|\, t_{i-k}, ..., t_{i-1}; \Theta), \tag{2.20}$$

where Θ represents the learned representation of a neural network, and k the token window size for the left-sided context. From this formulation, it should also become apparent how unlabeled pieces of text can simply be used during the pre-training phase without further annotation. Simple segmentation of the "current" token and its context allow for practically unlimited training data sourced from any form of text, which can then easily be parallelized with chunks of the same context size.

As we previously alluded to, this objective in itself is already quite powerful and results in models that can complete a piece of text reasonably well (given sufficient training data and time). However, the limitation also lies within the myopic formulation of the next token alone – sequence-level information (both syntactic and semantic in nature) are not explicitly captured by this way of training a model, and only by way of sequence context can generate a reasonable continuation.[15] Some models specifically augment the generic loss terms with additional task objectives (Lewis et al., 2020a; Zhang et al., 2020a), although these seem to have largely fallen out of fashion in more recent years. Major advancements since the inception of the Transformer can be roughly categorized into the usage of more sizable encoders/decoders (Liu et al., 2018a; Radford et al., 2018; Raffel et al., 2020), architectural modifications for specific limitations of Transformers (Shazeer et al., 2017; Su et al., 2024), improved alignment with human preferences to circumvent the previously mentioned issue of myopic optimization (Stiennon et al., 2020), and generally more awareness of training data curation (Kreutzer et al., 2022; Longpre et al., 2023).

2.4.3 Commercial Tools For Text Summarization

Given the driving motivation of investigating why there is a relative scarcity of *usable* text summarization tools, we also take the time to look at some of the existing commercial solutions using automated approaches to text summarization.[16] In contrast to the previous sections, we will be unable to provide much detail on the underlying algorithmic choices made, but do our best to provide some educated guesses of the underlying technologies. This is in no way meant as an ex-

[15] Interestingly, even during model fine-tuning, the same loss function is applied. We can only speculate where the improved performance is coming from. One reasonable explanation could be the narrower in-domain distribution or more uniform task contexts.

[16] For non-automated creation of text summaries, we can notably highlight Blinkist (`https://www.blinkist.com/`, last accessed: 2023-04-19), which provides human-curated summaries of (mostly non-fiction) books.

haustive list; especially with recent advancements in the large language model space, funding has become readily available for start-ups boasting the usage of "Artificial Intelligence", which also broadly touches upon the summarization space. We expect several (arguably more or less relevant) additions to this list in the near future.

Summly An early, and rather notable, exception to the otherwise sparse commercial success of summarization tools is Summly[17], an app designed for the bullet-style summarization of web content. Primarily intended to accelerate the digestion of news-style articles, it reached around half a million downloads within the first few months of its publication, and was famously acquired by Yahoo! in 2013. Unconfirmed speculations argue that the central method of Summly was then incorporated into the Yahoo! News Digest app[18], which was ultimately retired in 2017. Little is known about the algorithm used by Summly, although based on the available information it seems to be an extractive method utilizing some form of semantic parsing. However, the tool originally worked for several languages, which made it arguably the most impactful solution so far.

Cohere Summarize Another more recent addition, Cohere.ai[19] released a beta version of a summarization endpoint in February 2023[20]. Based on available information in the original blog post, it seems that Cohere is internally using a hybrid summarization system, which increases the possible context length of their endpoint to "up to 50,000 characters". The abstractive stage of their model is likely based on a version of their instruction-tuned generation endpoint, given that additional prompt texts are allowed. While Cohere positioned itself as a potential solution for long-form summarization tasks (as they are present, e.g., in the legal domain, or for scientific use cases), there are no major players reporting the usage of said endpoint, and it has been officially declared as a deprecated endpoint in upcoming releases.

Birch AI Unlike other model providers which provide a generic endpoint and leave task definitions up to the users, Birch AI[21] provides a rather domain-specific application for the automation of call center operations. More specifically, this includes the additional modality of audio inputs, which is first mapped to an internal textual transcription, before additional summarization operations are performed. Given that their focus is primarily on dialogue summarization, which goes beyond the scope of this work, we cannot comment on the relevant architectural choices. How-

[17] https://www.crunchbase.com/organization/summly, last accessed: 2023-04-19.

[18] https://www.thedrum.com/news/2014/02/14/case-study-development-summly-mobile-news-app, last accessed: 2023-04-19

[19] https://cohere.ai, last accessed: 2024-04-02.

[20] https://txt.cohere.ai/summarize-beta/, last accessed: 2024-04-02.

[21] https://birch.ai/, last accessed: 2023-04-19

ever, they claim to train and host their own models,[22] which is a critical requirement for their target industry (healthcare).

Google Docs TL;DR In early 2022, Google introduced a new feature for its product Google Docs, which allows the automated creation of a summary for sufficiently long documents.[23] Given the wide adoption of Google's GSuite, this may be the most-used commercial application of text summarization at the moment. An interesting choice is the underlying model, which is described as a "hybrid architecture of a Transformer encoder and an RNN decoder.", distilled from a more complex Pegasus model (Zhang et al., 2020a) for faster inference.[23] It is not said what document lengths this model can handle, but the targeted length for output summaries is between one and two sentences only. Similar to Cohere, the efforts have failed to make a big splash, and Google has since renewed its efforts with Gemini in the direction of more generic systems (Anil et al., 2023), which most recently announced successful internal tests with context windows of up to 10 *million* tokens.[24]

OpenAI GPT Most prominent in mainstream media are the recent GPT variants proposed by OpenAI, especially since the release of ChatGPT, originally an instruction fine-tuned variant of GPT-3.[25] OpenAI's models are also capable of following instructions for summarization-style tasks, even in non-English languages. This behavior is reasonable, given the inclusion of summarization-related instruction tasks in derived fine-tuning sets[26] GPT-4-Turbo increased the available context window size to 128,000 tokens, being suitable even for extensive document summarization tasks.[27]
There are a number of works evaluating the efficacy of these models for summarization-specific use cases, with Goyal et al. (2022) being the most thorough in analyzing not only the performance in terms of automated scores, but also human preference ratings. While they focus on the particular domain of news summarization (likely due to the context window constraints at the time), it demonstrates the potentially orthogonal nature of human and score-based preferences. The findings were that humans tend to prefer GPT-3-generated summaries over other methods, although the specific ratings were highly subjective, adding to the difficulty of evaluation. None of the compared approaches were simple baselines, which takes slightly away from the findings, especially with respect to the importance of factuality (likely prevalent across methods).

[22] https://birch.ai/news/ia-summit-fireside-chat-yinhan-liu-from-birchai, last accessed: 2023-04-19.

[23] https://ai.googleblog.com/2022/03/auto-generated-summaries-in-google-docs.html, last accessed: 2023-04-19

[24] https://blog.google/technology/ai/google-gemini-next-generation-model-february-2024/#context-window, last accessed: 2024-04-03

[25] https://openai.com/blog/chatgpt, last accessed: 2023-04-19

[26] See, e.g., the distribution of tasks in the fine-tuning set used for Stanford's "Alpaca" model: https://github.com/tatsu-lab/stanford_alpaca/blob/main/assets/parse_analysis.png, last accessed: 2023-04-19.

[27] https://platform.openai.com/docs/models/gpt-4-and-gpt-4-turbo, last accessed: 2024-04-03

Other Model Providers Similar to OpenAI, there are by now a range of generic neural models that can be utilized for summarization tasks as well. Anthropic's Claude 2 model family has been adopting an increasing context window size of 100,000 tokens and were the first to provide working context windows of that size.[28] However, the initial use case described by Anthropic itself is not necessarily one for a summarization system, but rather a closed Question Answering tool. Importantly, they still use singular documents for context, and did not explicitly allow for multiple document uploads at the same time. This was later fixed with the availability of Claude 3, which also increased the token limit even further, up to 1 *million* tokens.[29]
Perplexity.ai[30] targets a niche of mostly retrieval-augmented generation use cases, and heavily focuses on scholastic use cases. They internally provide access to different model providers and limit the context window, but also expect most of the context to be provided during the retrieval step. Similarly, You.com[31] provides grounded generations, although focusing more on providing search results, which can be seen as a weak and unguided form of summarization.

2.4.4 Data Sources for Text Summarization

When talking about academic approaches to text summarization, it has to be mentioned *what* data sources are used to evaluate (or, for neural approaches, train) models. Conveniently for us, Carbonell et al. (2000) present an early vision for assessing the properties of summarization datasets that goes well beyond many of the more recent works that present novel resources. After introducing said quality assessment structure and adjusting it to a more modern frame of reference, we spend the second half discussing various popular summarization resources and highlighting particular advantages and disadvantages.

Dataset Quality Assessment

As a general evaluation rubric for assessing the quality of datasets, we use the categorization by Carbonell et al. (2000), which uses three broad categories to analyze a dataset: **input characteristics**, detailing high-level properties such as the domain or language, the **summary purpose**, looking at the intended audience, and **output characteristics**, specifically for the textual properties of provided references. In particular, each of the three areas defines a more nuanced list of points to analyze, and we want to commend the original authors of the work for curating a list of properties that are still highly applicable today, and should be more actively used in the description of newly introduced dataset, in our opinion. Some of these points have been indirectly

[28] `https://www.anthropic.com/index/100k-context-windows`, last accessed: 2023-10-17.
[29] Likely by using Ring Attention (Liu et al., 2023a).
[30] `https://www.perplexity.ai/`, last accessed: 2024-04-03
[31] `https://you.com/`, last accessed: 2024-04-03

discussed as properties of summarization systems already, but we reiterate them in the context of a particular *document collection* at this point in time.

Input Characteristics

1. **Text Size:** This property is indicating whether a summary relies on a single document or several texts. Importantly, they also acknowledge "indirect multi-document settings", where the context of a single summary practically requires the inclusion of knowledge from other sources.
2. **Specificity:** Indicating whether a corpus handles a generic summarization settings, or a narrower domain-specific area. Many research projects nowadays are targeting a generic summarization system, but often fail to recognize that evaluation corpora are primarily of a domain-specific nature.
3. **Genre and Scale:** To differentiate the type of summary, one may look at the *genre* of a text corpus. Particularly for domain-specific datasets, it is important to further distinguish *which* domain (or "genre") the texts are from. Related to this is also the *scale* of input texts, which can range from short news articles up to entire books as references.
4. **Language:** We pay a particular focus to this aspect in our work, comparing whether resources are monolingual (in practice, primarily English), or multilingual.

Summary Purpose

1. **Situation:** Carbonell et al. (2000) mention "tied" and "floating" situations as the opposites end of the scale. Whereas tied summaries target a specific environment where the audience and the intended purpose of the summary is known, floating situations are more open-ended. This is another major point where popular resources are not properly put in context.
2. **Audience:** As a second purpose criterion, the specificity of a target audience can make a big diffefrence in how a summary may need to be written. E.g., addressing a summary to a primarily technical audience with educational background in the topic of an article may look vastly different from a broader and more accessible summary of the same text.
3. **Use:** Aside from the audience, it is also important to consider the intended *use* of a summary. For news summaries, for example, the intended use is to convince readers to read the full article. The consequential structure and length of a summary drastically varies based on this point.

Output Characteristics

1. **Degree of Extractiveness:** Determining whether a summary can be answered by simply restating existing content from an original document can play a huge role in how difficult it is to obtain a valid summary. On the other hand, coming up with re-written content that aggregates points from multiple parts of a text is more challenging.
2. **Coherence:** Where coherence (and the implied fluency of text) is also cited as an evaluation metric for the *assessment of generated summaries* (cf. Section 2.3), it can be important to realize that the ground truth summaries may not be very fluent to begin with. This is the case, for example, in the CNN/DailyMail dataset, where individual sentences for the gold summary are sourced from bulleted lists on the news websites.
3. **Partiality:** Any summary naturally lends itself to some form of bias, but some more so than others. The authors originally wanted to highlight potential reporting biases, but we believe that this also extends to naturally arising biases, such as positional preference, when creating a gold summary.

Document Understanding Conference (DUC)

The shared task of DUC 2001 is probably the first concentrated effort of evaluating (and providing data for) computational summarization settings. There are several iterations of the conference, starting with the initial 30 train and 30 test document clusters provided in 2001. Given that the DUC organizers were likely heavily inspired (if not directly involved) in the discussions from Carbonell et al. (2000) from the previous year, there is a visible imprint of several characteristics left on the distributed data and the task setup. While notably one of the smallest resources, DUC data comes with the advantage of having hand-curated resources.[32]

Track 1 consists of a single-document summarization task, where the target summary is of a fixed length of 100 words. **Track 2** introduces multi-document summaries with four different target lengths, between 50 to 400 words. We omit further discussion of the blue-sky **Track 3**. For both Track 1 and 2, the evaluation was done partially by humans at NIST, who organized this conference series. The scoring criteria where based on “grammaticality, cohesion, organization”, as well as coverage recall.[33] These were evaluated at the level of Elementary Discourse Units (EDUs) per the NIST guidelines, which limits comparability with modern, sentence-based discourse units. In terms of summary purpose, the shared task setting provides a very specific use (academic comparison of models) with a targeted “audience” (evaluators at NIST). It thus only serves as a proxy of an actual usable system, but at least with a decent evaluation setup in mind.

[32] `https://www-nlpir.nist.gov/projects/duc/guidelines/2001.html`, last accessed: 2024-02-29

[33] `https://www-nlpir.nist.gov/projects/duc/pubs/2001slides/pauls_slides/sld013.htm` and `https://www-nlpir.nist.gov/projects/duc/pubs/2001slides/pauls_slides/sld020.htm`, last accessed: 2024-04-03

More popular are nowadays the later extensions of the DUC challenges, which include "focused summary generation" from DUC 2003 and larger multi-document training resources from DUC 2004. Already for DUC, the documents were largely centered around resources derived from news articles, implying a fairly high specificity and narrow genre within the corpus. It should be noted, however, that DUC already included first resources for cross-lingual summarization, such as the Arabic-English Tracks 3 & 4 during DUC 2004. We refer the reader to Dang (2005) and Over et al. (2007) for a more detailed background on the design decisions behind DUC.

CNN/DailyMail, XSUM and NY Times

While DUC, and subsequently the Text Analytics Conference (TAC) carried out their regularly occurring shared tasks until 2011, the downside was always the rather limited data availability. As we have previously pointed out, the "data-hungry" nature of emerging neural methods required training data at a much larger scale as previously available, which ultimately inspired the creation of the two most popular training (and evaluation) datasets at the time of writing: CNN/DailyMail (or CNN/DM) (Hermann et al., 2015; Nallapati et al., 2016) and XSUM (Narayan et al., 2018).

CNN/DailyMail was originally designed as a corpus for Question Answering (Hermann et al., 2015), but already captured the article summary provided by a series of news bullets from article websites of the CNN and DailyMail. Nallapati et al. (2016) repurposed the available resource to a fully fledged summarization dataset shortly after, and demonstrated that neural systems can utilize this dataset to train SotA models. For reference, where the first DUC dataset provided around 600 samples, the CNN/DailyMail corpus boasts a total of 300,000 instances.

When it comes to the characteristics of the dataset, it should be mentioned that the text size and genre are particularly uniform.F Focusing entirely on news articles from only two sources, the dataset is automatically obtained and relies on a "soft alignment", expecting the news bullets to be a content-wise summary of an already short piece of text. The authors also do not differentiate the purpose of the summary, which in this case is specifically targeted towards *engagement*, by enticing users to read the full article text over just the summary bullets. This stands in juxtaposition to the intended goal of summarization systems, which primarily should serve to *avoid* any additional information digestion. A similar problem arises when thinking about the coherence of all bullets taken together (the target of our prediction setting), which again stands in contrast to the intended goal of a fluent and cohesive prosaic summary.

On the other hand, the audience of news websites such as the CNN is very broad, and could lead to a more differentiated view on how to generate summaries. As far as we are aware, it has not been explicitly researched whether tailoring CNN/DM summaries to specific target audiences has a noticeable effect on the summary generation.

Many of the same arguments can also be made for the later XSUM dataset (Narayan et al., 2018). Spanning roughly 226,000 articles, it follows the same construction logic as Hermann et al. (2015), scraping articles and teaser summaries from the BBC. XSUM claims a slightly higher level of abstractiveness (defined by novel n-grams in the target summary), but otherwise shorter input texts compared to CNN/DailyMail.

A third resource which has striking similarity to the previous resources, but remaining a less popular choice is the NY Times corpus by Sandhaus (2008). Interestingly, it predates the other tow resources by a few years, contains more data (around 650,000 samples suitable for summarization), including human-written gold reference summaries written by librarians. We find no apparent reason why it has been less popular, aside from the slightly more restrictive licence provided by its publisher, the LDC. We will discuss a number of more recent and diverse summarization datasets in later chapters, which address some of the concerns regarding the content diversity and text size we mentioned here.

2.4.5 (Automatic) Evaluation Metrics

Measuring usability of systems in the real world is difficult enough as it is, but for summarization systems, we are still not seeing any large-scale adoption that would enable verifying the efficacy of certain models in such a practical setting. In constrained academic settings, on the other hand, the limited funding and oftentimes tight deadlines force researchers to rely on faster and cheaper evaluation metrics. Various groups have long since worked on establishing automated evaluation of summarization systems in order to reduce the associated effort and costs (Resnick, 1961; Nenkova and Passonneau, 2004; Lin, 2004). We briefly discuss some of the prevailing evaluation strategies in this section, ranging from fairly simplistic and partially interpretable, to drastically more complex black-box evaluation models.

We also want to reiterate the distinction between reference-based evaluation metrics (i.e., some form of gold summary is available) and reference-free settings, in which only the original input text is available as an input. Without weighing in on the availability bias of reference-based systems, it should be mentioned that, while reference-free metrics are easily applicable to settings without further annotations, they themselves may exhibit inherent biases (Deutsch et al., 2022b).

ROUGE

What the evaluation metric BLEU (Papineni et al., 2002) is to Machine Translation, ROUGE (Lin and Hovy, 2002) is to text summarization.[34] As an approximation of human judgments, both BLEU and ROUGE rely in some form on the matching of tokens between a

[34] In some instances, BLEU is even considered as a separate evaluation metric in summarization research, see, e.g., Graham (2015).

system-generated summary and a (usually human-generated) gold reference, therefore constituting reference-based evaluation metrics. The argument for using a token-based matching function arises naturally from the previously introduced Document Understanding Conference's (DUC) summarization task (Lin and Hovy, 2002), where manually curated references already existed, but had not been previously utilized for automated "matching".

Let us assume that $\mathcal{D}_{gold} = \{g_1, ..., g_m\}$ refers to the collection of provided reference summaries g_i, and s a single prediction by a summarization system. One of the beneficial parts about working with reference-based corpora is also that the original task setup (e.g., the number of input documents in $\mathcal{D}$) does not matter for the eventual evaluation, and is entirely based on the references themselves.

Following are the definition of the simplest, n-gram-based ROUGE scores, commonly referred to as ROUGE-N, R$_n$ or R-n.

$$R_n^{prec}(\mathcal{D}_{gold}, s) = \max_{g \in \mathcal{D}_{gold}} \frac{\sum_{gram \in g} \min(count(gram, g), count(gram, s))}{\sum_{gram \in G} count(gram, s)}, \tag{2.21}$$

$$R_n^{rec}(\mathcal{D}_{gold}, s) = \max_{g \in \mathcal{D}_{gold}} \frac{\sum_{gram \in g} \min(count(gram, g), count(gram, s))}{\sum_{gram \in g} count(gram, g)}, \text{ and} \tag{2.22}$$

$$R_n^{F1}(\mathcal{D}_{gold}, s) = 2 \cdot \frac{R_n^{prec}(\mathcal{D}_{gold}, s) \cdot R_n^{rec}(\mathcal{D}_{gold}, s)}{R_n^{prec}(\mathcal{D}_{gold}, s) + R_n^{rec}(\mathcal{D}_{gold}, s)}. \tag{2.23}$$

The function $count(t, d)$ returns the exact number of occurrences of a particular n-gram t within the tokenized document d. We want to point out that the exact definition of ROUGE-N is disputed; the original paper provides conflicting formulations on whether to aggregate best scores across multiple gold summaries, or simply to take the maximum ROUGE-N value across different hypotheses.[35] The associated Perl script released by Lin has in fact a parameter to decide on which variant is to be used.[36] Given the default choice of averaging across hypotheses in this script, and the predominant adoption of that particular variant, we only introduce the averaging procedure here. We furthermore extend the commonly assumed separation into precision, recall, and F1 scores. Without further clarification, we will be referring to the F1 scores when reporting R_n without specification, even though the original acronym has the "Recall-Oriented Understudy" in the name, and thus exclusively refers to the recall-oriented variant in Equation (2.22).

Aside from reporting the uni- and bi-gram scores (R-1 and R-2), the third popular variant is ROUGE-L (or R-L/R$_L$ for short). In contrast to the n-gram-based overlap counts of the pre-

[35] We refer to a related discussion on the web, see https://stats.stackexchange.com/questions/558777/rouge-n-for-multiple-references, last accessed: 2023-04-14.

[36] A passed-down version of the original script can be found here: https://github.com/li-plus/rouge-metric/blob/master/rouge_metric/RELEASE-1.5.5/ROUGE-1.5.5.pl#L1171-L1183, last accessed: 2023-04-14

viously defined variants, ROUGE-L attempts to identify the Longest Common Subsequence (LCS) between the gold summary (or summaries) and a system output.

We follow the definition of Cormen et al. (2009) by defining the LCS problem as an optimization to return the longest possible sequence of tokens that appear in non-decreasing order in two different documents d_i and d_j, formally

$$\begin{aligned} LCS(d_i, d_j) = & \underset{V_T}{\operatorname{argmax}} \, [t_a, t_b, ..., t_c], \text{ s.t.} \\ & t_a, t_b, ..., t_c \in d_i \wedge \; t_a, t_b, ..., t_c \in d_j \wedge \\ & idx(t_a, d_i) < idx(t_b, d_i) < ... < idx(t_c, d_i) \wedge \\ & idx(t_a, d_j) < idx(t_b, d_j) < ... < idx(t_c, d_j) \end{aligned} \tag{2.24}$$

Here, $V^{\mathbb{N}}$ refers to an arbitrarily long token sequence constructed from the underlying vocabulary V and $idx(t, d)$ returns the index position of token t in document d. In practice, the LCS of two sequences can be determined in quasi-quadratic runtime $\mathcal{O}(|d_i|_{token} \cdot |d_j|_{token})$ via dynamic programming (Cormen et al., 2009). We assume that a somewhat efficient implementation is chosen, but also point out that the computation is generally performed on relatively short text segments (namely, the summaries), which means an evaluation with ROUGE remains fairly efficient. ROUGE-L extends the analogous separation of different precision/recall-focused variants in ROUGE-n, but substituting the token matching with our previously defined LCS:

$$R_L^{prec}(\mathcal{D}_{gold}, s) = \max_{g \in \mathcal{D}_{gold}} \frac{|LCS(g, d)|_{token}}{|s|_{token}}, \tag{2.25}$$

$$R_L^{rec}(\mathcal{D}_{gold}, s) = \max_{g \in \mathcal{D}_{gold}} \frac{|LCS(g, d)|_{token}}{|g|_{token}}, and \tag{2.26}$$

$$R_n^{F1}(\mathcal{D}_{gold}, s) = 2 \cdot \frac{R_L^{prec}(\mathcal{D}_{gold}, s) \cdot R_L^{rec}(\mathcal{D}_{gold}, s)}{R_L^{prec}(\mathcal{D}_{gold}, s) + R_L^{rec}(\mathcal{D}_{gold}, s)}. \tag{2.27}$$

Again, we slightly deviate from the original formulation by Lin, in that we assume ROUGE-L to operate similarly as a maximization objective over multiple existing reference summaries, which is not explicitly spelled out in the paper itself. We finally note that there exists another variant of ROUGE-L, which calculates the LCS at the sentence level and only then combines the scores across the number of sentences. Our argumentation for focusing on the summary-level LCS is again the predominant utilization of this version in related work.

With the formal definitions spelled out, we continue to elaborate on why ROUGE has become the de-facto choice for automated metrics: Lin (2004) reports Pearson correlation of ROUGE measures with human judgments on several DUC evaluation corpora, and finds a strong association between the two. While this assumption is nowadays heavily disputed (Graham, 2015; Kryscinski et al., 2019; Fabbri et al., 2021), it may still remain a competitive choice even with many newly proposed measures available (Deutsch et al., 2022a). The algorithmic simplicity of ROUGE has even some unintended consequences with respect to the evaluation of non-English data. Given that the only requirement for adopting ROGUE to another language is the existence of a proper tokenizer, it can be easily transferred to evaluate scenarios in multiple languages. This is also an inherent advantage over some of the more complex models, which generally require huge amounts of resources in each respective language.

Even conceptually, one can intuit why ROUGE has a strong correlation with human judgments: particularly the in-order requirement of LCS matches has a strong relation to the *coherence* evaluation dimension; high ROUGE-L scores therefore may implicate a grammatically strong summary. Similarly, the uni-gram matches of ROUGE-1 point towards a shared vocabulary (particularly the precision-oriented formulation of R_1^{prec}), precisely what is needed for a high *relevance* in summaries. However, the main complaints about ROUGE are generally related to the evaluation of *semantically consistent* summaries. Here, token-based metrics alone cannot score overly well.

Another fact that should give food for thought for the reader is a worrisome development in recent years that recent works produce new evaluation corpora with only single reference texts per sample. Instead of having multiple gold summaries available, alleviating the strictness of exact token matching present in ROUGE-based evaluation, the automated nature in which new resources are often produced leads to a less diverse evaluation setting with only singular references.[37] Coupled with the insufficiency of incorporating multiple gold answers during neural training routines, we have seen a (presumably indefinite) departure from the more holistic generation setting, which is hampering the requirement of "subjectivity" in summary generation in particular. Finally, many authors are also confusing ROUGE as a complete replacement of manual summary inspection, instead of a more complementary (large-scale) study. This can lead to overstated claims about state-of-the-art performance of particular systems, which is detrimental to meaningful progress in generalized settings, as many of the "default evaluation corpora" are focused on a narrow setting, see Chapter 3.

The Pyramid Method

An interesting variant of a semi-automated evaluation metric is the Pyramid Method, proposed by Nenkova and Passonneau (2004), which incorporates the computation of automated scores based on further manual analysis. The authors claim that the earlier evaluation procedure in-

[37] To be completely transparent, this also includes our own proposed corpora for summarization, see Aumiller and Gertz (2022a) and Aumiller et al. (2022b).

troduced by the DUC annotation guidelines has significant design flaws, ultimately leading to a rather random agreement among human evaluators when it comes to scoring respective summaries. Subsequently, the central argument is that this may also cause low agreement for the evaluation scores of machine-generated summaries, where texts might receive a low score despite a relevant content. Instead, the proposed method identifies sub-sentence level phrases (so-called "Summary Content Units", or SCUs for short), which aim to identify phrases that appear in multiple reference summaries.

The key problem is that – while scientifically sound and reasonable in its assumptions – the annotation of sub-sentence units across *multiple reference summaries* is prohibitively expensive to perform. Even assuming the existence of a multi-reference datasets, this approach still requires more human intervention (or an otherwise sufficiently "good" approach to annotate the Summary Content Units), which is why we do not consider this approach further. However, we use this as a further example to illustrate the unfortunate disconnect between an "optimal" and "feasible" evaluation scenario. Partially because of the recent trend towards predominantly single-referenced evaluation corpora, approaches like the Pyramid Method are often not applicable, although it would likely provide a better evaluation bed for many research settings.

Language Models as Evaluators

More recently, language models (LMs), particularly large language models (LLMs), have become a staple in the NLP community, and allowed for significant progress not just on the modeling side, but also have been increasingly useful for evaluation purposes. Intuitively, a language model's internal representation is used to assess the quality of a text, based on the likelihood of a given system-generated text being produced by the language model itself. Primarily, this builds on the concept of **perplexity** (PPL), defined as the product of (negative log) token likelihoods by a model M[38],

$$\mathrm{PPL}(D_i, M) = \exp\left(-\frac{1}{|D_i|_{token}} \sum_{i}^{|D_i|_{token}} \log p_M(t_i|t_{<i})\right). \tag{2.28}$$

$t_{<i}$ refers to the token sequence up to the i-th token.[39] However, perplexity alone is not necessarily suited to directly evaluate the quality of a summary, given that the summary's ratings largely depend on the additional conditional of the input documents $\mathcal{D}$. In the following, we present two popular methods that take the previously stated limitation into account.

[38] We follow the formulation by Huggingface, see `https://huggingface.co/docs/transformers/perplexity`, last accessed: 2023-04-14.

[39] This formulation slightly differs from the contextual loglikelihood loss term defined in Equation (2.20) by considering the full previous context and normalization over steps.

BERTScore   Presented by Zhang et al. (2023), BERTScore is an automated reference-based evaluation metric which builds on the pairwise similarity of embeddings generated by Transformer Encoders, e.g., BERT (Devlin et al., 2019). A visual representation of the method can be seen in Figure 2.3. We follow the theoretical notation of (Zhang et al., 2023), and assume a neural model M, which is able to turn a token-level segment $[t_1, ..., t_m]$ into a vectorized representation of the same segments, $[\mathbf{t}_1, ..., \mathbf{t}_m]$, where each token is represented by a normalized k-dimensional vector $\mathbf{t}$. Formally

$$M(d_i^{tok}) = [\mathbf{t}_1^i, ..., \mathbf{t}_m^i], \mathbf{t}_1^i, ..., \mathbf{t}_m^i \in \mathbb{R}^k, \ |\mathbf{t}_j^i| = 1 \, \forall j \in [1, m]. \tag{2.29}$$

Assuming a gold reference segment g and a system-generated candidate segment s, the score is computed by greedily matching the most similar tokens between the two sequences, using cosine similarity.[40] The authors arrive at three formulations, focusing on the precision (i.e., normalizing by the candidate segment length), recall (normalizing by the reference length) and the harmonic mean of the two values (F-score).

Assuming two embedded documents $\mathbf{d_i} = M(d_i^{tok})$ and $\mathbf{d_j} = M(d_j^{tok})$, we can define BERTScore as follows:

$$BERTScore_{rec}(d_i, d_j) = \frac{1}{|d_i|_{token}} \sum_{t_m^i \in d_i^{tok}} \max_{t_n^j \in d_j^{tok}} \mathbf{t}_m^{i\top} \mathbf{t}_n^j, \tag{2.30}$$

$$BERTScore_{prec}(d_i, d_j) = \frac{1}{|d_j|_{token}} \sum_{t_m^j \in d_j^{tok}} \max_{t_m^i \in d_i^{tok}} \mathbf{t}_m^{i\top} \mathbf{t}_n^j, \tag{2.31}$$

$$BERTScore_{F1}(g, s) = 2 \cdot \frac{BERTScore_{prec}(d_i, d_j) \cdot BERTScore_{rec}(d_i, d_j)}{BERTScore_{prec}(d_i, d_j) + BERTScore_{rec}(d_i, d_j)} \tag{2.32}$$

Similar to ROUGE, it is in fact unclear which variant people refer to when they are “using BERTScore”, as multiple implementations exist, including a TF-IDF-normalized version. Given the evaluation results in the original paper, coupled with the general preference of F1 metrics as a “balanced” representation, we argue that it is mostly the basic $BERTScore_{F1}$ variant.[41] Another problematic fact is that people generally do not disclose the underlying language model that was used to obtain the vector representations. According to the authors’ repository, their own im-

[40] Zhang et al. assume a normalized vector length, which simplifies the similarity computation to a single dot product. We adopt this notion for improved readability.

[41] This is despite the fact that the recall-oriented variant seems to fare much better on average in the evaluation benchmark by Fabbri et al. (2021).

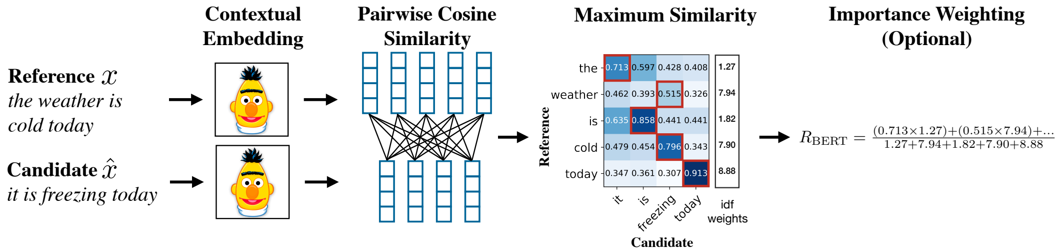


Figure 2.3: Schematic view of the computation of BERTScore. Alignments are created between tokens of the two inputs, based on the embedding similarity. Source: Zhang et al. (2020b)

plementation supports over 130 models by now,[42] with a particular recommendation for a variant of DeBERTa (He et al., 2021).

BARTScore Where BERTScore has the disadvantage of having to rely on embeddings that may not be directly related to the original training objective of the underlying language model, BARTScore (Yuan et al., 2021) utilizes a more direct approach to evaluation: depending on the specific setting, BARTScore uses an autoregressive language model (in this case, BART (Lewis et al., 2020a)) to obtain the conditional modeling probability $P(D_j|D_i)$. Formally, BARTScore can be defined as the log likelihood over prior sequences, or

$$\text{BARTScore}(D_i, D_j) = \frac{1}{|D_j|} \cdot \sum_{t=1}^{|D_j|} \omega_t \log P(D_{j,t}|D_{j,<t}, D_i, \theta), \tag{2.33}$$

where $D_{j,t}$ refers to the t-th token in D_j, and $D_{j,<t}$refers to the sequence of all tokens until position t, and ω to the parameters of the underlying language model. ω_t can be used to define more specific token weights, such as determined by IDF weighting or similar approaches, but is generally left at $\omega_t = 1, \forall t < |D_j|$ in the default implementation, as no empirical benefit can be observed from adjusting the weights. The normalization factor is required to discourage the model from preferring shorter outputs, as (negative) log likelihoods would otherwise diminish the score for longer sequences.[43]

It is yet uncertain whether these metrics provide a meaningful improvement in terms of correlation with human annotations, specifically in generalized summarization settings (Deutsch et al., 2021b; Fabbri et al., 2021; Deutsch et al., 2022a), but they do generally complement a ROUGE-focused evaluation. Similar to ROUGE, we can again relate back to the evaluation dimensions of Section 2.3. As we will later show in Section 3.5, BARTScore indeed has some correlation with hu-

[42] https://github.com/Tiiiger/bert_score/blob/master/README.md, last accessed: 2023-04-18.

[43] Note that the original authors have a discrepancy in that regard between the (unnormalized) equation in the paper, and the (normalized) implementation in their code repository.

man factuality ratings, but still remains far from perfect, and general findings point to BARTScore being better suited to summarization-specific use cases than BERTScore (Yuan et al., 2021; Liu et al., 2023b). Both of the LM-based metrics show an exceptional ability to model textual coherence, but they also exhibit the undesirable self-preference for outputs generated by similarly trained models, see Deutsch et al. (2022b) and Liu et al. (2023b). This means that outputs generated by LMs similar to the ones used for evaluation will achieve higher scores than should be otherwise assigned.

A major limitation of the applicability of LM-based evaluation criteria, which is particularly relevant for the contents of this work, is the narrow focus on English. While it is theoretically possible to utilize the introduced metrics for non-English settings, this requires the practical availability of *high-quality* trained language models in the evaluated language. Even for German, where there exists a fairly decent representation of monolingual models (Chan et al., 2020),[44] there is no empirical evidence that evaluates the suitability of LM-based metrics for the (fair) analysis of summarization results. While it may be a reasonable assumption that the correlation is somewhat consistent between languages, this is not necessarily the case. Specifically differences in the morphological structure of languages can cause a drastically different behavior of, e.g., the token-alignment procedure in BERTScore, leading to detrimental results. We did not further investigate this issue in our own work, but encourage interest readers to conduct a feasibility study for (particularly multilingual) models that go beyond just English evaluation.

2.5 Domain-specific Distinctions for Summarization

In Section 2.3, we were primarily concerned with a more psychological separation of the human preferences across different evaluation dimensions. However, in domain-specific applications of summarization, we may well encounter users that have particular (and strong) preferences for what exactly a summary should be structured like, or what contents need to be included. In this section, some of the relevant domains for text summarization will be introduced, which we will revisit periodically throughout the remainder of this work. In particular, this includes a brief analysis of the specific requirements for some prominent use-cases within the various domains. In general, this section serves as a critical differentiation on why singular *general-purpose* summarization systems may not be suitable, or certainly much harder to achieve than some related works make us believe.

We begin by evaluating the focus on news-related use cases, and contrast it by analyzing two highly specified domains, with medical and legal summarization scenarios. We also try to shape this

[44] Also see `https://github.com/bminixhofer/gerpt2`, last accessed: 2023-04-19, or `https://github.com/dbmdz/berts`, last accessed: 2023-04-19, for available models.

within a broader category of "business" use cases, which has an arguably similarly broad conceptual view as a "general-purpose" system. The following domains do not pose an exhaustive list – in fact, we will briefly reference scenarios that are not considered in this thesis, but still of high impact, such as scholarly summarization scenarios.

2.5.1 News Domain

We start by analyzing the setting in which most academic news-related summarization is focused, which includes the construction of a generic single-document summary. Aside from these, we argue that there are many more settings that have not received as much attention, and may inadvertently be more closely tied to Information Retrieval settings, given a broader scope of input documents.

Generic Single-Document Summaries

Due to the popularity of previous news-related corpora (see Section 2.4.4), much time and effort has been spent on identifying the particularities of such news-focused summary generation. During the meeting series preceding the eventual DUC conference, it was already discussed what implications the various summarization scenarios could be (Carbonell et al., 2000).[45] When we are talking about the basic task (also evaluated in the aforementioned datasets) of generic text snippet generation, there is also an associated limit on the length, usually implying a target of less than 5 sentences.

We want to start by briefly investigating the domain's most central drawback: Much of the news-related content follows a structure referred to as the "inverted pyramid" (Scanlan, 2000), which entails that relevant content should be contained within the beginning of an article. Consequentially, when optimizing a summarization system towards the news domain, we observe the resulting systems primarily picking up on content elements at the beginning of a text (Kryscinski et al., 2019; Zhu et al., 2021b). However, summarization of news articles also has a distinct advantage over other domains (particularly for research settings): we can assume that summaries ultimately only need to have a narrow *purpose*. Following the subdivision of purpose factors by Scanlan (2000), news articles can be categorized as follows:

1. **Situation** Summaries of news articles are always tied (opposed to "floating"), meaning they serve a particular environment where the "who, why and when" of a readership is defined. Ultimately, serving a summary to news readers should provide the gist of a new story, po-

[45] This paper is in our opinion also one of the criminally underrated works of the early 2000s. Having several of the most influential summarization researchers of the time as authors, it conveys several ideas about Question Answering and text summarization that we have seen implemented since; truly a visionary report.

tentially interest the reader in reading more about the topic, and be relevant immediately after the release of a news story.

2. **Audience** We may assume that the readership of a news publisher is the primary intended audience, and as such has a rather unspecified prior knowledge about the topic. Yet, the primary interest in an *extremely short* summary is shared between readers.
3. **Use** As expressed before, the primary use of the summary is to interest readers in the full story, or otherwise capture their interest.

For any of the following domains, it is much harder to establish such a streamlined vision of *what* a summary should be; this makes a homogeneous evaluation setup infinitely harder, as it requires the consideration of a multitude of different aspects to be properly represented. Despite (or maybe, because of) this simplicity in the task setup, the observations made by Over et al. (2007) about the evaluation results across DUC's competitions are striking:

> *Automatic summaries seldom performed better than simple baselines based on the structure of news articles.*

While algorithms have since improved drastically, particularly with the introduction of neural methods (Nallapati et al., 2017; See et al., 2017), it still remains a considerable challenge to beat available baselines that are much simpler to compute.

News Aggregation Scenarios

Aside from the generic summarization case, we argue that practically relevant use cases may also extend to multi-document summarization for news articles. Particularly news aggregator sites, such as they are offered by Google, Apple and Yahoo!, are generally dealing with incoming document streams from multiple sources. To present readers with relevant information content, it is crucial for these sites to aggregate (and also summarize) similar articles into one digestible result snippet. We re-iterate the importance of content de-duplication and the particular challenges of open multi-document summarization (Giorgi et al., 2022). Notably, many news(paper) organizations are also accredited to a particular end of the political spectrum (Schudson, 2002). To avoid subversive messaging contained in aggregated articles, and in order to represent a less biased report, it may be required to further disentangle the various input documents with respect to their particular framing. While it may initially seem that in such cases the simple approaches of taking opening segments no longer work, empirical results during the evaluation of DUC 2006 have shown the opposite, and human annotators strongly prefer the simple baseline to more complex systems (Dang, 2006).

Closely related is the task of providing direct answers in search engine result pages (SERPs), which are usually motivated by a user-specified interest. These can also be considered as a case of the open-domain multi-document summarization scenario, although it frequently suffices to find relevant passages (i.e., purely extractive settings).

2.5.2 Medical Domain

Particularly with recent adversarial trends in the healthcare industry, practitioners are facing increased workloads in their day-to-day business (Torjesen, 2021). There are several attempts to remedy the information overload during administrative tasks for medical professionals by employing some form of summarization system. For this domain, we assume a restriction on text-only inputs, which is not necessarily the case here: patient conversations may be available as audio / video, which first has to be transcribed (see our previous discussion of companies, such as Birch AI), and decision support systems frequently need to account for multi-modal inputs, such as images from X-ray or MRT scans. It may also include otherwise time-sensitive information (e.g., repeatedly conducted blood tests or other vital measurements in the form of time series data).

We distinguish between a number of tasks that have been introduced in the literature as potential avenues for application scenarios. Even with the limitation of only considering textual data, there is a broad spectrum of different medical summarization settings. We begin with the automated generation of conclusion sections for medical analyses, such as it is the case for radiology reports, and also on dialogue summarization systems with assumed transcription data available. But summaries can also be extremely valuable for patients themselves. We argue for the particular usability of layman-focused summarization tasks (also applicable for similar legal use cases), where technical content is simultaneously simplified and summarized. This offers intriguing challenges, as the user background is implicitly modeled as a "knowledge prior" in such cases. Mostly, there exists no further differentiation than an *expert* and *novice* user, although in practice we may encounter further distinctions.

On the other hand, medical use cases have hard constraints, which differentiate it from other summarization domains. Many countries require specific certification of products used for medical purposes, which includes inspection of technological solutions. The US Food & Drug Administration (more commonly known as the FDA) and its European equivalent, the European Medicines Agency, or EMA, impose strict rules on the reliability of system outputs. Making an incorrect prediction in a medical system can, after all, cause irreversible and lasting damage to a human being. As automated decision systems (which includes summarization models) are currently still prone to make factual mistakes, e.g., changing up the name of medication or other central aspects, it remains uncertain whether they would be admissible for patient care solutions.[46]

For this particular reason, applications of text summarization systems in medical settings should

[46] A draft proposal for future regulation within the United States is proposed here: https://www.fda.gov/media/122535/download, last accessed: 2023-04-25.

be followed with specific care for the underlying systems. A good indicator is generally whether medical professionals were included in the study design and evaluation setup, as they have a better understanding of the actual obstacles for deploying practical solutions.[47] We note that even large corporations, such as Google, struggle to utilize their systems in practice due to the regulatory approval requirements.[48]

Even from a Computer Science point-of-view, though, medical NLP applications offer a range of technical challenges. One of the primary concerns is often the use of highly technical vocabulary, which is the cause of issues in general language models. For this purpose, it may be required to pre-train domain-specific variants which are more adept at representing complex medical (or broadly, technical) vocabulary (Beltagy et al., 2019). Furthermore, previous studies find that medical experts use copy-pasting as a means to accelerate the writing of highly templated task settings (Liang et al., 2022), which can cause high redundancy in the generated outputs, or a significant number of train-test leakage due to near duplicates.

Automated Medical Conclusion Generation

Regulatory grievances aside, we take a brief look at one of the more straightforward application scenarios, which is the generation of summarizing conclusions in analytical reports. A much-studied example is the automated creation of summaries in radiology reports (Goff and Loehfelm, 2018; Zhang et al., 2018b; Liang et al., 2022), with similar multi-modal tasks, such as chest x-ray interpretations (Jing et al., 2019).

It poses as one of the most similar task settings compared to general-purpose summarization scenarios, such as discussed before. In the simplest setting, singular input documents are compressed into a series of conclusive findings, which often contains a recommendation for the next steps in medical treatment. Importantly, the target audience of such reports are generally other personnel familiar with the medical domain. This consequently allows authors (and models) to write summaries that are fairly complex and may contain direct references to the original input without necessarily providing much further explanation. In our opinion, the main challenges in conclusion generation settings are:

1. The implicit optimization of factually accurate answers (Zhang et al., 2020c). Although it is notable that primarily abstractive systems were utilized in the mentioned works, which tend to suffer more from such problems compared to their extractive counterparts.

[47] For the interested reader, we recommend the comparison of two concurrent studies in for radiology report findings, mentioned in the following section. We can observe a striking difference in the methodology and evaluation employed by medical professionals (Goff and Loehfelm, 2018) and computer scientists (Zhang et al., 2018b).

[48] See `https://blog.google/technology/health/ai-llm-medpalm-research-thecheckup/`, last acessed: 2023-04-23.

2. On a related note, it may be helpful to ground medical expressions in available knowledge bases or ontologies (MacAvaney et al., 2019; Sotudeh Gharebagh et al., 2020). This grounding may indirectly alleviate issues with factual inconsistencies, as entities are a frequent issue. A weaker form of external referencing is also employed by the Pointer Generator architecture, introduced by See et al. (2017).
3. Lastly, reports in general may exhibit a relatively rigid structure, which can be exploited for a more targeted generation (Jing et al., 2019). Critically, the document structure may be more similar within a particular application setting, compared to general-purpose applications. As such, important document information can be repeatedly found in similar portions of a document, leading to better generation results. This is also strongly related to other settings, particularly legal domain scenarios.

Outside of these strict requirements, we also highlight the particularly abstractive nature of findings, which may be grounded in the *experience* of professionals. As an example, we may have varying conditions that lead to the exact same conclusion: a phrase, such as "tumor detected", may be stemming from various implicit findings in the document, without ever being directly mentioned. This lack of expressiveness is particularly challenging for models to learn on, as it requires the formation of an implicit (and potentially highly complex or ambiguous) cause-consequence relationship.

A further necessary requirement may be the inclusion of several respective documents, such as earlier diagnoses, or related findings of other patients, effectively turning this into a form of multi-document setting. These cases do not necessarily share the same prerequisites as the aforementioned single-document scenario, particularly with respect to the shared content structure, while simultaneously providing more external information about patient condition.

Patient Dialogue Summarization

While we do not explicitly cover the related field of dialogue summarization, we briefly mention it due to the abundance of potential application scenarios. In contrast to the comparatively "linear" structure of analysis documents, written by medical professionals, a turn-based interaction may be observed during doctor-patient consultations (Song et al., 2020; Joshi et al., 2020; Navarro et al., 2022; Savkov et al., 2022). For such settings, it already becomes more interesting to consider the intended audience of the summaries. While related works assume that interaction results should be summarized into bulleted lists of key findings (or the free-form text equivalent of such), we can also imagine systems that summarize the interactions for *patients*. This consequently requires a very different level of detail in the provision of conclusions, or otherwise needs to indicate how questions lead to particular insights by a doctor. Similar to the previously mentioned setting of report generation, we again can imagine instances of highly abstractive conclusions being drawn from an interaction, without them ever being explicitly spelled out.

Interestingly, existing patient interaction protocols can also be redirected towards efforts of creating doctor-like medical chatbots (Singhal et al., 2023). These settings generally focus less on a summarization of previous interactions, and rather a compression of information for a layman user (see subsequent section). Furthermore, we already discussed the implications of required medical licenses, which makes the (presumably unsupervised) interactive setting for patients unlikely to be deployed anytime soon.

Layman Summarization in the Medical Field

As mentioned, not only medical experts may be interested in a summarized finding of conversations or medical reports. In fact, explainability of findings seems to be correlated with patient trust, particularly when replacing doctors with expert systems (Alam and Mueller, 2021). For the effective communication of findings towards patients, however, it is necessary to choose an appropriate level of text complexity, based on the patient's technical understanding. As we will be evaluating this notion of simplifying text in our own work, we briefly highlight the recent progress in what is known as "layman summarization" (Grigonyte et al., 2014; Goldsack et al., 2022; Jeblick et al., 2023). Here, it is assumed that complex concepts (including highly domain-specific vocabulary) have to be adjusted for a less adept audience. Notably, these are also fields that have recently seen expansions to non-English settings, see, e.g., Trienes et al. (2022) and Dercksen et al. (2023).

Prominently, though, the perception of "reading difficulty" is highly subjective, and may differ between patients (Gooding et al., 2021; Gooding and Tragut, 2022). Imagine, for example, the simplification necessary to explain the common cold to an adult versus a pre-teenage child, which expands greatly on the difficulty of generating appropriate output. Such meta-contexts are rarely available in existing solutions, and definitely not in the fairly recently developed systems in the medical field. It remains to be seen whether these will play a larger factor in future work, but adaptive solutions can be considered superior to a "one-size-fits-all" approach.

Interestingly, layman summarization offers a previously untouched scenario, in which contextually implied information may be explicitly stated in a "summary". This is a rare scenario in which the output text adds previously unseen information, in this case for the purpose of semantic clarification. Unlike other settings, where most content can either be attributed directly from the source text, or constitutes a fairly surface-level paraphrasing of existing semantic information, the simplification setting introduces the concept of additions in the context of summarization as well.

2.5.3 Legal Domain

The legal industry is a similarly sensitive domain which has a theoretical abundance of practical use cases which currently fail at the strict requirements imposed on employed software solutions. While there is no overseeing body similar to the FDA/EMA in the legal domain, the direct implication of being *legally responsible* for client outcomes makes it notoriously difficult to employ approaches which potentially omit important details or results. For a sober look at the applicability within the broader legal NLP space, we also recommend the study of our previously published (German) commentary, see Gertz and Aumiller (2022).

Generally, it can be noted that, unlike the medical domain, differences in local legislature can have a severe impact on the underlying document structure, or logical reasoning. Even when considering documents of the same language, e.g., English, relevant elements may look completely different in a court document in the US, compared to one in the United Kingdom, due to differences in their judicial systems. A secondary factor is the temporal dimension, playing a highly relevant role in legal use cases. Because newly introduced changes to the law code can affect future outcomes, it has to be considered which documents are *most appropriate* by their respective temporal ordering, with the most recent document likely being more applicable. Finally, we highlight the extreme verbosity and length of legal documents. This distinction makes the application of existing "general-purpose" systems oftentimes impractical, if not outright impossible, as they are not designed to work with context windows sufficiently large for legal documents (Aumiller et al., 2022b).

We again introduce a variety of scenarios that benefit from the application of text summarization systems, and discuss their respective challenges. One relatively broad scenario is related to court documents and associated summarization cases, which includes, e.g., Argument Retrieval, but also decision predictions. On the other hand, preparatory work in the legal industry requires reliable and fast access to a number of related works, usually through the use of citations of past cases. This also poses as a great example of contextually dependent summarization, enabling readers to quickly get key concepts of related works, even without the necessary familiarity of the referenced legal texts.

Not mentioned again because of large overlap with the medical domain is the aspect of layman-focused (simplified) summaries of legal texts (Manor and Li, 2019; Chandrasekaran et al., 2020). This nonetheless presents a relevant topic of active research and has practical use cases in activist causes for better understanding of licensing and agreement terms.[49]

[49] See, for example, the "Terms of Service; Didn't Read" project: `https://tosdr.org/`; last accessed: 2023-04-25.

Court-related Summarization

Oftentimes, relevant document collections in the legal domain may have some association with courts. They can consist of (more or less direct) transcripts of court proceedings, supplementary documents related to hearings themselves, or simply the abridged court rulings based on some prior decision. Given the dependence on prior rulings, it may be required for lawyers or judges to have a grasp of an ever-growing body of literature, which can benefit from a simplified (or at least, accelerated) access through summarized content.

When summarizing court decisions, it may be necessary to distinguish between the scenario of compressing information for future readers (Xu et al., 2021a), versus the more prevalent setting in academic works of *predicting a verdict* (Glaser et al., 2021b), which can also be seen as a form of informed summarization.

We particularly also highlight the benefit for cases spanning multiple hearings, where it may be necessary to rebut previously made statements by the opposing party in court. For this particular purpose, we redirect the reader to the field of Argument Mining (Lawrence and Reed, 2019), which has previously seen application in the legal domain as well (Conrad et al., 2009; Elaraby and Litman, 2022).

In cases spanning several hearings, the complexity of interactions along multiple temporal axes can also not be understated. In these particular scenarios, we may encounter an ordering of *historical events* which is not necessarily aligned with the *order of discussion* in the actual hearings. To this end, we have also previously introduced a model for quickly analyzing the "historical temporal ordering" dimension, irrespective of the underlying mention order (Hausner et al., 2020a,b). We expand on this notion in Chapter 6.

Citation Summarization

Outside of the court houses, there are plenty of other application areas for summarization systems as well. One notable example, which we will detail in Section 3.2, is the summarization of relevant legislature or related commentary. Oftentimes, this problem is deeply intertwined, with the German judicial system relying strongly on arguments outlined in commentary works, instead of basing it solely on past court decision, such as it is more common in the United States, for example. This generally requires systems to be able to handle a *diverse* range of documents. Furthermore, identifying relevant passages requires a sufficiently good retrieval system designed for legal users, which has also attracted a growing amount of attention in recent years (Van Opijnen and Santos, 2017; Verberne et al., 2023).

Notably, citation summarization (or related span-identification settings) are not exclusively relevant to the legal domain. We highlighted similar concepts present in the medical field, where it focused on the inclusion of external knowledge bases and ontologies, but there also exists a whole body of literature for the extraction of appropriate information for literature surveying (Aumiller et al., 2020; Chandrasekaran et al., 2020).

As a sub-task relevant for the correct identification, it is necessary to segment a document into (semantically) coherent units, which we have previously introduced as *segments*. Interestingly, this in itself already poses a non-trivial challenge in legal docouments (Aumiller et al., 2021; Glaser et al., 2021a). Because of the frequent use of non-standard textual elements, such as lists, enumerations, paragraphs (including subsections), etc., traditional sentence boundary detection methods work poorly on the more complex structure of legal documents, which makes it challenging to even select relevant extractive content.

2.5.4 Business Domain

Identifying relevant applications of summarization in the more general business domain is a lot trickier. This is due to the more heterogeneous nature of tasks, many of which are kept away from public dissemination as "trade secrets". As an extrapolation of sorts, many of the existing academic works are related to the summarization of financial documents. To what extent these find practical application, can only be guessed, but we assume that the low reliability of (abstractive) systems still poses a drastically reduced value gain for high-stakes industry settings. More practical, but less directly a form of summarization, are structured generation settings, which oftentimes include the transcription of tables into more actionable insights.

Financial Document Summarization

A central pillar of the available documents are the annual (or quarterly) reports of companies, which have been extensively studied in the context of summarization as well (La Quatra and Cagliero, 2020; Abdaljalil and Bouamor, 2021). Notably, the Financial NLP workshop series (Chen et al., 2021) and Financial Narrative Processing workshops (El-Haj et al., 2022a) have centered around these particular topics. Notably extensions for shared tasks beyond monolingual processing (El-Haj et al., 2020) have been proposed recently (El-Haj et al., 2022b), although the evaluation settings of both are fairly limited in their (manual) analysis of results. For a more practical perspective, we recommend the work by Leidner (2020), who shapes his opinion from a wealth of history in the industry.

A key problem is the frequent limitation to singular input documents; practical scenarios usually combine information from a variety of sources, instead, which causes a discrepancy between the investigated settings and the actually relevant scenarios. This underestimates the relevance of processing metadata, which may include sensitive information about the origin or temporality of, e.g., news coverage. A related topic will be the entity-centric summarization of content, see (Filippova et al., 2009; Maddela et al., 2022), which can be considered another form of context-sensitivity.

Structured Text Generation

As a weaker form of summarization, and more related to the broader setting of Natural Language Generation (NLG), we mention data-to-text scenarios, which oftentimes include the multi-modal extension to various levels of structured data. The most prominent example is the translation of tabular input information into abstractive insights (Parikh et al., 2020). Notably, though, models trained on the ToTTo dataset by Parikh et al. suffer from practical shortcomings (Sundararajan et al., 2022). Highlighted can be the more thorough evaluation of data-to-text systems by Ehud Reiter and collaborators (Inglis et al., 2017; Thomson and Reiter, 2020; Thomson et al., 2023). The main difficulty for translation from structured data is the question of just how static the underlying data structure really is; for variable schemata, this problem already becomes significantly more difficult. Extending it to semi-structured (or unstructured) only adds to this level of difficulty. Another problem is the question of representation for structured information. Despite the implied presence of some internal representation (i.e., the structure itself), most existing work in the NLP/NLG area is focusing on purely unstructured inputs. Consequently, suitable models may not support the input of more structured information. Unlike other settings described in the previous sections, though, structure generation scenarios may have much shorter inputs, and are therefore easier to model (Parikh et al., 2020).

2.5.5 Other Domains

While we attempt to consolidate a diversified picture of summarization and its application in various domains, there are certain instances we will largely omit for the sake of brevity. This includes primarily scholarly application scenarios, since they do have a niche impact on the overall applicability, and relate in several aspects to settings that have been detailed before. Aside from this, we have talked about potential use cases that extend to other modalities or input structures, such as dialogue-focused input documents, or inclusion of images/tables as other forms of references.

Automated Generation of Scientific Abstracts

An active research area in the early days of summarization (Luhn, 1958; Resnick and Savage, 1960), automatically creating technical abstracts has been an obvious application given text summarization's roots in the library sciences. Even today, this is practically relevant as a research topic, given the familiarity of many researchers with the underlying corpus data, as well as the broad availability of papers through platforms like arXiv (Cohan et al., 2018). It furthermore poses a fairly generic summarization setting, requiring little external information beyond the immediate document, as well as targeting a wide audience (i.e., not necessarily adjusting to individual user needs). Ultimately, the most relevant information in scientific articles can often be found at positions early on (introduction) or at the very end (conclusion). This combination of factors draws many parallels between scientific document summarization and existing works on news summarization, such that approaches working well on one area can be reasonably extended to the other one.

Notable is the introduction of system-generated automated "Too Long; Didn't Read" (TL;DR) sections in Semantic Scholar.[50] This is a direct derivative of the work by Cachola et al. (2020) and among the only successful implementations of such kind. Even then, the system is limited in the types of papers it will annotate (primarily papers from the natural sciences), only uses particular sections to limit the input length, and requires large-scale annotation of summaries to work well.

Literature Summarization

Slightly related, but a more complex setting, es the task of literature surveying, or citation relationee Here, the task is related to either summarizing content from multiple works (Portenoy and West, 2020), or relating which sections of other works are particularly relevant for a current context window (Chandrasekaran et al., 2020; Aumiller et al., 2020). These settings exhibit strong ties to the use cases mentioned in the medical and legal domain, and as such find themselves basis for some of the previously mentioned applications for literature surveying, e.g., Perplexity.ai. They particularly focus on providing scholarly citations and back-linking to claims within generated answers, but overall still require strong steering through inputs from humans.

[50] https://www.semanticscholar.org/product/tldr, last accessed: 2023-04-25.

3 Limitations of Current Directions in Summarization Research

"All models are wrong, but some are useful."

George E.P. Box

We have now spent considerable time building up a list of desirable attributes for a *practically useful* text summarization system and discussing some of the innovations within the field over the years. While several end-to-end systems have been built, we are still seeing a shortage of systems actively used by substantial user base to address a summarization-related need. To answer *why* existing systems are insufficient for practical purposes, we specifically want to address some of the remaining shortcomings within the area of text summarization. While these limitations are relatively well-known within the community, they are oftentimes ignored in academic settings, which severely limits the practical relevance of recent works.

We begin by discussing some of the existing works on limitations in summarization research in Section 3.1. Here, we primarily focus on three broader categories: 1) Issues related to the narrow focus on English and disregard for multilingual systems, 2) length limitations of existing text generation systems with respect to document length, and 3) the availability of meaningful evaluation metrics. Given our own focus on summarization for German, we further elaborate on some existing tools that deal with text summarization in non-English languages, either in mono-, multi-, or cross-lingual settings.

As a solution to the data scarcity problem of multilingual (high-quality) evaluation corpora, we propose a new dataset, called *EUR-Lex-Sum*, in Section 3.2. It provides long-form textual descriptions of legal acts relevant to the European Union, including associated human-written summaries of these documents. As the EU has 24 official languages, we are able to collect a highly multilingual dataset *including sentence-level alignments between languages*.

During the creation of our dataset, we notice that the quality of training resources heavily affects downstream performance and overall generalization. In particular, discrepancies between system-generated responses and the human expectation of a "meaningful" summary become apparent for systems trained on automatically obtained corpora. Section 3.3 introduces a more detailed account of hypotheses regarding the (data) quality issues present in existing systems. These are centrally related to the domain-specificity of prominent datasets and issues herein, which limit their applicability outside the intended domain.

To address basic data quality concerns in summarization resources, we formalize a series of automated detection mechanisms in Section 3.4, which are able to reduce the error rate in existing summarization datasets. While we demonstrate the applicability of these methods on German datasets, the approach is largely language-agnostic, and it encourages the future application to other datasets as well. Worryingly, we also observe that model performance on standardized test sets drops significantly when filtering test splits of public benchmark datasets. Based on our analysis, this implies that model performance is generally overestimated, and the distribution skewed due to low-quality outliers.

To also help quantify issues present in *model outputs*, we finally introduce a new metric to analyze the factual consistency of generated text in Section 3.5. Our algorithm represents textual elements as a series of "factual tuples", which are comparable across input texts and corresponding generated outputs. Our experiments indicate that the method is comparable to current state-of-the-art methods and allows for a more linguistically grounded evaluation of factuality.

The contents of this chapter are based on the following peer-reviewed publications:

Dennis Aumiller, Ashish Chouhan, and Michael Gertz. EUR-Lex-Sum: A Multi- and Cross-lingual Dataset for Long-form Summarization in the Legal Domain. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7626–7639, Abu Dhabi, United Arab Emirates, December 2022b. Association for Computational Linguistics. URL https://aclanthology.org/2022.emnlp-main.519

Dennis Aumiller, Jing Fan, and Michael Gertz. On the State of German (Abstractive) Text Summarization. In Birgitta König-Ries, Stefanie Scherzinger, Wolfgang Lehner, and Gottfried Vossen, editors, *Datenbanksysteme für Business, Technologie und Web (BTW 2023), 20. Fachtagung des GI-Fachbereichs „Datenbanken und Informationssysteme" (DBIS), 06.-10, März 2023, Dresden, Germany, Proceedings*, volume P-331 of *LNI*, pages 195–220. Gesellschaft für Informatik e.V., 2023. doi: 10.18420/BTW2023-10

Jing Fan, Dennis Aumiller, and Michael Gertz. Evaluating Factual Consistency of Texts with Semantic Role Labeling. In Alexis Palmer and Jose Camacho-collados, editors, *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 89–100, Toronto, Canada, July 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.starsem-1.9

3.1 Related Literature

We are not the first to critically evaluate progress in text summarization research. Zhang et al. (2018a) were the first to notice that outputs from "abstractive" summarization systems regularly turn out to be suspiciously extractive in nature and spent time to address these by more focused evaluation efforts. Kryscinski et al. (2019) move beyond simply focusing on the evaluation of generations, and criticize the dimensions of datasets, evaluation metrics, and model choices in their analysis of research progress within the field. Extending the comparison of various text summarization innovations to large pre-trained language models, Huang et al. (2020) again study the relative performance differences of various approaches. All three works confirm our believe that there are severe shortcomings in the current state of evaluating new approaches, and that some of the more popular neural models reveal a rather limited improvement over much simpler heuristic approaches in certain cases.

There are several additional facets to the limitations of current systems and multiple ways to address the shortcomings that we wish to introduce. In an effort to place our own contributions within the larger field, we divide the following sections into an analysis of existing approaches going beyond English (Section 3.1.1), as this poses a biased reporting of "innovations" on approaches that are essentially custom-built for English texts. We further focus specifically on an analysis of resources and models for German, which in our opinion serves as an upper-bound on research progress in a non-English language, due to a relatively active German-specific research community. Aside from the scarcity of (multilingual) resources, we also highlight the recent works on evaluation with longer input documents, challenging the relatively limited context windows of many published end-to-end systems (Section 3.1.2). These reveal another shortcoming in the practical applicability of models, oftentimes focusing on generating summaries from relatively short inputs. Section 3.1.3 provides some more context for recent evaluation metrics in addition to the methods already presented in Section 2.4.5. We then focus on the evaluation of *factuality* in system generations (Section 3.1.4), which generally has emerged as a popular focus dimension in recent evaluations, due to the problematic nature of faithfulness in neural systems.

3.1.1 Summarization Resources Beyond English

In our discussions of text summarization research so far, we have implicitly made the assumption that the language of a particular piece of text does not play any larger role in the effect it has on the system function. However, in practice, a majority of research (and practical works) are focusing primarily on English, creating a sort of self-reinforcing bias towards that language. Methods that work well for English do not necessarily translate well to other languages, be it because of the grammatical structure or a different alphabet, for example. Subsequently, transferring meth-

ods that have been designed for English is also non-trivial. More traditional linguistic approaches often require software tooling (such as language-specific stemming or tokenization approaches), which might simply not exist, and more modern neural systems are notoriously hard to adjust for other languages due to their inductive training biases on monolingual (English) corpora.[1]
We specifically want to highlight some of the active efforts that help widening the accessibility by focusing on non-English application scenarios in text summarization. This starts with the availability of language resources beyond English, or cross-lingual datasets. Even there, we can see similar biases, with European languages (having the academic funding and general "online presence") far outweighing other language families in terms of available research. Categorically, we split existing approaches in monolingual non-English/multilingual and cross-lingual works.

Multilingual Summarization Datasets

With a particular eye on large-scale multilingual *resources*, most of the datasets are from the past few years.[2] MLSUM (Scialom et al., 2020) is based on news articles in six languages, however, without cross-lingual alignments. Similarly without alignments, but larger in scale is MassiveSum (Varab and Schluter, 2021). XL-Sum (Hasan et al., 2021) does provide document-aligned news articles in 44 distinct languages, where the authors extracted data from translated articles published by the BBC. In particular, their work also provides translations for several lower-resourced Asian languages. Less popular is MLGSum (Wang et al., 2021), which also boasts availability in several languages. The authors utilize available news aggregators and extend the crawling to news outlets from various languages in a multilingual equivalent to the method behind CNN/DailyMail summaries.[3] WikiLingua (Ladhak et al., 2020) borders the multi- and cross-lingual domain; some weak cross-lingual alignment is constructed by the authors, but is limited to the English reference texts. Specifically for Indian language pairs but also using multilingual parallel websites as a source, Urlana et al. (2023a) provide a parallel resource across fourteen languages, and also evaluate cross-lingual setups in their analysis.

Cross-lingual Summarization Datasets

To our knowledge the earliest explicit setup for cross-lingual summarization is introduced by Saggion et al. (2002), utilizing a parallel corpus of Cantonese and English newspaper articles. The

[1] Although there exist approaches to attempt a post-training transfer to other languages for neural models, see, e.g., Minixhofer et al. (2022).

[2] Select language-specific implementations of earlier text summarizations exist, e.g., for German (Reithinger et al., 2000). The same goes for non-English monolingual corpora, which we omit here for conciseness.

[3] Our primary reason for excluding this dataset in further experiments is the inaccessibility of the processed dataset. The authors only provide a crawler for users to obtain the over one million (!) articles themselves, leading us into questionable legal territory when it comes to website fair use.

task of cross-lingual summarization was more broadly popularized by the later DUC 2004 Arabic-English summarization task (Over et al., 2007). While there are several works on new models for multilingual summarization (Lim et al., 2004; Litvak and Last, 2013), to our knowledge only smaller cross-lingual corpora have been proposed during previous decades.

More recently, Wang et al. (2022c) provide an extensive survey on the currently available methods, datasets, and prospects. Modern resources for cross-lingual summarization can be divided into two primary categories: synthetic datasets and web-native multilingual resources. For the former, samples are created by directly translating summaries from a given source language to a separate target, which can carry potential negative effects for the subsequent evaluation due to translation errors (Zhang and Toral, 2019). Examples include English-Chinese (and vice versa) by Zhu et al. (2019), and an English-German resource (Bai et al., 2021). Both works utilize news articles for data and neural MT systems for the translation. In contrast, there also exist multilingual datasets with naturally aligned data, where both references and summaries were obtained primarily from parallel websites. Global Voices (Nguyen and Daumé III, 2019), XWikis (Perez-Beltrachini and Lapata, 2021), Spektrum (Fatima and Strube, 2021), and CLIDSUM (Wang et al., 2022b) represent instances of datasets for the news, encyclopedic, and dialogue domain, with differing numbers of supported languages. Notable is also the effort by Zheng et al. (2023), who focus on providing a resource of 94,000 document pairs with longer contexts. Their dataset is built from Chinese papers with English abstracts as the target summary, with the average input document length exceeding 2,800 character symbols.

German Text Summarization

As a case study of an individual language, and a basis of related work for some of our later discussions, we specifically take a look at German resources for text summarization. This includes both available systems, but also training datasets and their respective focus. Compared to some other languages, German has a dedicated community that has been previously investigating methods specifically in a monolingual context (Frefel, 2020; Frefel et al., 2020). In addition, there exist other works where it is discussed as one of several languages in a multi- or cross-lingual context, per our previous discussion.

While we are slowly starting to see a greater diversity in the available training resources for German text summarization, it comes as a small surprise that the availability of trained system is much less diverse. As will become more apparent in later sections, the primary focus for training systems is a combination of a pre-trained checkpoint and one predominant training resource ("MLSUM", focusing on the ´German subset). Below, we elaborate on considered model properties, differentiating between the availability levels of related works in context. A summary of known properties can be seen in Table 3.1. For a more focused overview of LLM summarization performance, we

Model	Training data	Test Set	Evaluation	Filtering	Public	Reprod.
mrm8488/bert2bert[4]	MLSUM	MLSUM	ROUGE	None	✓	✓
ml6team/mt5-small[5]	MLSUM	MLSUM	ROUGE	Length	✓	✗
T-Systems/mt5-small[6]	CNN/DailyMail, MLSUM, XSum, Swisstext	MLSUM	ROUGE	Length & Overlap	✓	✗
Shahm/t5-small[7]	MLSUM	MLSUM	ROUGE	None	✓	✗
T5-base[8]	?	?	ROUGE	?	✓	✗
german-t5[9]	Swisstext	MLSUM	ROUGE	?	✗	✗
Aksenov et al. (2020)	Swisstext	Swisstext	ROUGE & manual	?	✓	?
Parida and Motlícek (2019)	Swisstext & CommonCrawl	Swisstext	ROUGE & manual	None	✗	✗
Venzin et al. (2019)	Swisstext	Swisstext	ROUGE & manual	None	✗	✗
Fecht et al. (2019)	Swisstext	Swisstext	ROUGE & manual	?	✗	✗
Glaser et al. (2021b)	LegalSum	LegalSum	ROUGE	?	✓	?
Liang et al. (2022)	Radiology	Radiology	ROUGE & manual	?	✗	✗

Table 3.1: List of German neural abstractive summarization models, divided into systems available on https://huggingface.co and academic artifacts. We detail their known properties from provided training recipes or published papers. If we have access to models, we denote whether public scores are reproducible within ±0.5 ROUGE points ("*Reprod.*"); ? in the reproducibility column indicates that models were available, but we were unable to successfully run their code.

refer the reader to Schubiger (2024), who evaluates various LLM systems, although he does not specifically fine-tune models for the task and relies on in-context or zero-shot setups.

Publicly Available Systems The primary source for available models is the Huggingface Hub[10], which allows filtering by supported language and appropriate task (in our case "summarization"). We note that some of the available models are not properly tagged, but spent considerable time to ensure no models were accidentally ignored. For users who have uploaded several different versions, we selected the model with the highest self-reported evaluation scores.
Given that users on the platform are likely familiar with other services of Huggingface (including

[4] https://hf.co/mrm8488/bert2bert_shared-german-finetuned-summarization, last accessed: 2022-10-06
[5] https://huggingface.co/ml6team/mt5-small-german-finetune-mlsum, last accessed: 2022-10-06
[6] https://huggingface.co/T-Systems-onsite/mt5-small-sum-de-en-v2, last accessed: 2022-10-06
[7] https://huggingface.co/Shahm/t5-small-german, last accessed: 2022-10-06
[8] https://huggingface.co/Einmalumdiewelt/T5-Base_GNAD, last accessed: 2022-10-06
[9] https://github.com/GermanT5/german-t5-eval, last accessed: 2022-10-06
[10] https://huggingface.co/models, last accessed: 2023-01-14

their datasets browser), it comes as no surprise that the diversity of chosen models is low. Available systems either choose checkpoints of mT5 (Xue et al., 2021) or variants of T5 (Raffel et al., 2020) as a basis for fine-tuning experiments. In our investigation, we found that alternatives based on (m)BART (Lewis et al., 2020a; Liu et al., 2020) are consistently outperformed according to self-reported metrics. In order to train effectively on large quantities on data, most approaches use one of the smaller checkpoints, referring to model variants with fewer parameters. Outside of the model hub, code repositories exist for the BERT-Copy architecture by Aksenov et al. (2020) and Encoder-Decoder models used by Glaser et al. (2021b). However, we were unable to set up inference for custom datasets based on the respective code bases.

Private Models A further selection of models has been published in response to the Swisstext 2019 summarization challenge (Parida and Motlícek, 2019; Venzin et al., 2019; Fecht et al., 2019). However, neither team has published any associated public repository. Similarly, no models are available from Liang et al. (2022) who work on radiology reports. As the only one of the major cloud providers, Microsoft offers a dedicated extractive summarization service through Azure that supports German.[11] One of the first commercial solution providing a platform for abstractive summarization also supporting German texts was Aleph Alpha,[12] with other chat model providers by now supporting multiple languages as well, frequently explicitly mentioning German. Official support is provided by, e.g., OpenAI (OpenAI, 2023) and Cohere.[13] Anecdotally, several more systems exhibited decent cross-lingual capabilities in our own experiments with German prompts.[14]

German Data Sources for Summarization

In our experiments, we focus on datasets across a variety of domains. To our knowledge, these cover the most prominent publicly available sources used for training German systems, in particular for single document summarization. Notably, some of the mentioned corpora have different derivations stemming from either the same base corpus, or subsequent re-crawls of a resource. Where necessary, we indicate the existence of such a dataset.

MLSUM (Scialom et al., 2020) This multilingual dataset was presented as one of the first efforts in making larger-scale training sets available for multiple languages that also include Ger-

[11] `https://learn.microsoft.com/en-us/azure/cognitive-services/language-service/summarization/language-support`, last accessed: 2022-10-06

[12] `https://www.aleph-alpha.com/use-cases/conversion#trilingual-summary`, last accessed: 2022-10-06

[13] `https://docs.cohere.com/docs/command-r`, last accessed: 2024-04-04

[14] Most notably, most systems are able to semantically parse a request in German. While some will respond in German (e.g., Perplexity.ai), others instead default to an English response instead. This includes You.com and Cohere's models in RAG mode.

man as a language. MLSUM is constructed by extracting news articles and associated summary sections as generation targets. We use the German subset in this work, which is by far the most popular dataset used for training and evaluating resources in German, based on our survey. Despite the popularity, Philip May was the first to report issues in the quality of summaries,[15] an aspect we will analyze in more detail later.

MassiveSumm (Varab and Schluter, 2021) The construction of this particular dataset is similar to MLSUM and focuses on a large number of automatically extracted summaries from web articles in multiple languages. The authors perform some rudimentary filtering with respect to empty samples and even go as far as avoiding similar issues to MLSUM by removing what they call "ellipsoid summaries", i.e., fully extractive summaries that appear at the beginning of the reference text. While the quality of the samples is comparatively low due to the automated extraction process, this corpus is by far the largest readily available resource considered in our experiments. It has the potential to improve existing training setups with its sheer number of samples.

MLGSum (Wang et al., 2021) The German subset of this multilingual resources provides almost 500,000 data points, but does not have a usable variant available (see previous discussions). The data consist of news articles from German outlets, where editorial bullets constitute the target summaries.

Swisstext (Frefel et al., 2020) In contrast to the –generally shorter– news articles available in MLSUM, the Swisstext dataset provides longer-form summaries based on German Wikipedia pages, which has been later extended to the GeWiki corpus (Frefel, 2020). For the construction, the central argument is that the introductionary paragraph serves as a "summary" of the remaining article text. The provided dataset comes with a training portion and a private test set, meaning no ground truth summaries are available for the test samples. A multilingual variant of this idea, the XWikis corpus, was introduced shortly after (Perez-Beltrachini and Lapata, 2021). While the XWikis corpus contains more samples per language, including German, monolingual sample alignments are not readily available for download. Adding the fact that German summarization works primarily deal with the Swisstext dataset, we choose the Swisstext variant as our Wikipedia-based source.

Klexikon (Aumiller and Gertz, 2022a) Another Wikipedia-related resource, but with different target summaries. This dataset has much longer summary lengths compared to the Swisstext dataset, but covers a much smaller subset of only around 3,000 samples. Given the secondary

[15] https://may.la/blog/2022/02/23/anomalies-in-the-mlsum-dataset/, last accessed: 2022-10-06

focus on simplification in the target summaries, this corpus requires a considerably higher level of abstractive reformulations during the generation. See Section 5.2 for a more detailed analysis.

WikiLingua (Ladhak et al., 2020) As the third multilingual resource, summaries in this corpus are extracted from the WikiHow platform. Here, Ladhak et al. consider short instruction summaries of individual steps in WikiHow guides and align those with the referenced paragraphs. The general tone of the dataset is rather informal and is in a more imperative style in comparison to other data sources. To align samples across languages, images within articles of different languages are matched to identify presumably parallel paragraphs. Importantly, this means that for German articles, frequently only *some* of the article's paragraphs are actually contained in the dataset.

LegalSum (Glaser et al., 2021b) Another area benefiting enormously from high-quality summaries is the legal domain. LegalSum is the first German resource providing summaries of around 100,000 court rulings. On average, these samples require the highest amount of compression across evaluated datasets.

EUR-Lex-Sum (Aumiller et al., 2022b) A smaller but high-quality resource built on top of EU legal data available in 24 languages, including German. We discuss the challenges of creating said resource further in Section 3.2.

20 Minuten (Kew et al., 2023b) This resource is based on the content from the Swiss news outlet of the same name. Interestingly, the content provides different granularities of summarization targets. Ranging from headline generation to bullet summaries, there are between 22,000 to 41,000 samples depending on the specific target.

Further Resources In addition to these datasets, we are aware of several other resources that are, however, not publicly available. A news-related datasets for German summarization can be found in experiments by Nitsche (2019), where data was supplied by the German Press Agency, but no public record of the corpus exists otherwise. A second news-related resource is hinted at online by users on Huggingface's platform.[16]
For clinical summarization, Liang et al. (2022) work with a dataset of about 11,000 radiology reports; given the sensitive nature of the data, no publicly available version exists as of now. We are also aware of a secondary source of the WikiLingua dataset curated by the GEM community,[17] which provides additional samples, as well as a pre-split validation and test section not provided in

[16] A rather large one, with around 400,000 articles is indicated here: https://huggingface.co/Einmalumdiewelt/PegasusXSUM_GNAD/discussions/1#6308eb5037556c4ab03258df

[17] https://gem-benchmark.com/data_cards/wiki_lingua

the original dataset. In preliminary experiments, we found that $> 99.89\%$ of the data were valid samples for the GEM source (see Section 3.4.1). Most problematic is their choice to automatically aggregate different paragraphs into one summary, which can cause disjoint referencing in the (also aggregated) input texts, especially for the parallel multilingual subsets.

Finally, all of the discussed corpora so far are types of single document summarization resources. Datasets for training summarization systems that consider multiple source texts exist at smaller scales (Benikova et al., 2016; Zopf, 2018). More recent experiments with neural models on top of the latter corpus have been conducted by Johner et al. (2021) and Mascarell et al. (2023).

3.1.2 Long-form Text Summarization

Mentioned as a key limitation of earlier systems, research has recently focused on extending the context length of Transformer models, which were by default trained for context windows no longer than 512 tokens.[18] Popular early approaches utilize sparse attention mechanisms, which enable transformer-based models to handle longer documents (Beltagy et al., 2020; Zaheer et al., 2020). However, the document structure is not explicitly considered in current models, and sparse attention is generally considered as a lossy approximation of the exact computation leading to eventual degradation. Yang et al. (2020) propose a hierarchical Transformer model, SMITH, that incrementally encodes increasingly larger text blocks. The latest developments instead focus again on scaling exact computation of attention windiws with Rotary Positional Embeddings (RoPE) (Su et al., 2024) and better hardware support with FlashAttention (Dao et al., 2022), which enable context lengths that can represent contents of entire books.

Given the lengthy nature of legal texts, we previously investigated different approaches for separating content into topically coherent segments, which can benefit the processing of unstructured and heterogeneous documents in long-form processing settings with limited context (Aumiller et al., 2021). From a data perspective, Kornilova and Eidelman (2019) propose BillSum, a resource based on US and California bill texts, spanning between approximately 5,000 to 20,000 characters in length. A similar corpus based on the EUR-Lex platform appeared around the same time as our initial result, but focuses exclusively on English documents (Klaus et al., 2022). They utilize an automatically aligned text corpus for fine-tuning BERT-like Transformer models on an extractive summarization objective. Their best-performing approach is a hybrid solution that prefaces the Transformer system with a TextRank-based pre-filtering step. As a similarly long resource of few, but high-quality documents, Kryscinski et al. (2022) present BookSum, based on books from Project Gutenberg.[19] Their dataset provides both chapter- and book-level summaries, although

[18] For reference, the rendered text on this page alone equals already about 650 tokens using the BERT tokenizer. This means that the majority of early Transformer systems are unable to handle the full content of even a single page of text.

[19] `https://www.gutenberg.org/`, last accessed: 2024-05-02

the authors themselves mention severe quality differences between different samples, as they are likely written by several users without any shared guidelines on how to create the respective synopsis.

3.1.3 Generic Evaluation Metrics

As an alternative to the controversial use of ROUGE as an evaluation metric (Lin, 2004), more recently proposed alternatives rely on score computation from a single gold summary only (Ermakova et al., 2019). As a more generic criticism, Ter Hoeve et al. (2022) note that this direction of relying on a single perspective during evaluation setups is overall detrimental for the generalization ability of abstractive summarization systems. However, just how strong this reference overfitting is in existing systems has not been studied, much less for non-English languages.

Examples of more recent evaluation metrics include primarily neural similarity scoring between a generated summary and a gold reference (Sellam et al., 2020; Zhang et al., 2020b). Ultimately, neural methods are also incredibly expensive to employ for evaluation settings, potentially taking several days to evaluate a single experiment on a test set (Nan et al., 2021). Besides the cost factor, the main issue with such alternative scores is two-fold: On the one hand, a distinct advantage of co-occurrence-based metrics such as ROUGE is the simplicity in transferring the score computation to another language. Even basic extensions, such as stemming algorithms, are generally available in several languages. Trained metrics, such as BERTScore (Zhang et al., 2020b) or QAEval (Deutsch et al., 2021a), however, are severely limited in their transferability to other languages, and would require dedicated efforts to port them to German, for example. On the other hand, recent statistical analyses have shown that when accounting for annotator expertise, correlation of evaluation metrics and human preference can vary significantly (Fabbri et al., 2021). When additionally controlling for variance and confidence intervals, correlation with human judgments only rarely improves statistically significantly over ROUGE (Deutsch et al., 2021b). A particular investigation on metrics for German summarization was conducted during the second Swisstext challenge (Frefel et al., 2020). Submitted resources were only marginally better than ROUGE baselines for judging system quality (Paraschiv and Cercel, 2020; Biesner et al., 2020), reinforcing our point that a focus on more languages is necessary. For crowd-sourced evaluation approaches, Iskender et al. (2020) further elaborate on the importance of survey setups and considerations for expert annotators to ground evaluation results.

Beyond evaluation-focused studies, some works have previously attempted to redefine the desirable properties in a summary (Fan et al., 2018; Steen and Markert, 2021; Ter Hoeve et al., 2022), where the main findings generally agree that existing evaluation metrics fall short of the incorporation of *semantically consistent* and *subjectively varying* summaries.[20] Data-centric approaches to remediate the shortcomings of evaluation metrics have also been proposed (Clark et al., 2023).

3.1.4 Analyzing Factual Consistency of Summarization

Analyzing the outputs of summarization systems has become a more active area of research, in part due to the influx of generic summarization systems becoming available (Nallapati et al., 2016; See et al., 2017; Lewis et al., 2020a). As discussed in Section 2.3.1, factuality is oftentimes a key limitation of existing systems, and largely dictates the utility of a generic summary. It is also an evaluation dimension where correlation with existing metrics is not very high, especially compared to other quality factors, such as relevance or coherence. Goodrich et al. (2019) were the first to propose a reference-based estimator to specifically gauge the factual consistency of generated summaries with their gold reference. The proposed approach is based on a tuple representation of atomic "facts" across both pieces of text, which are compared on the basis of matching fact arguments. The authors use a triplet representation of *(Subject, Predicate, Object)*. Subsequent work has proposed alternative metrics based on textual entailment (Falke et al., 2019; Mishra et al., 2021) and Question Answering (QA) (Wang et al., 2020; Durmus et al., 2020), where agreement of answers to questions on the reference and summary are used for estimating factuality. However, especially QA-based metrics require additional fine-tuning on task-specific datasets, which makes the adoption to new domains (or languages) fairly expensive and prohibitive for broader application scenarios.

The only other work to our knowledge that uses a SRL-based factuality estimation is presented by Fischer et al. (2022) and the most similar to the approach we present in Section 3.5. In comparison to our method **SRLScore**, the authors aggregate "role buckets" across the entire text instead of sentence-specific tuples and do not necessarily differentiate between the semantic context of individual sentences. Empirically, their implementation has lower correlation with human ratings than compared approaches, which is contrary to our own findings. Li et al. (2022) frame factuality estimation as an in-filling task, where fact statements are withheld as masked tokens in a generated summary, and a separate model is trained to predict missing "facts". Notably, this approach relies on the assumption that the majority of factual mistakes stems from noun phrases and entity mentions (Pagnoni et al., 2021).

An alternative body of literature has explored the possibility to exploit Language Models (LMs) directly for estimating factual consistency: The previously discussed metric BertScore (Zhang et al.,

[20] We crucially do not focus on the (very limited) analysis of human evaluation setups, although we highlight their importance as an additional qualitative feedback mechanism. From personal experience, human-in-the-loop experiments are less comparable due to nuances in annotator background and variation within annotations over multiple runs. This sentiment is also echoed in studies on the reproducibility of human evaluation studies (Belz et al., 2023).

2020b) uses LM-generated representations to generate alignments for scoring that also exhibit some correlation with textual factuality. In comparison, PRISM (Thompson and Post, 2020) or BARTScore (Yuan et al., 2021) directly use model perplexity as a factuality estimate. Xie et al. (2021) explore masking approaches that fall somewhere between the works of Li et al. (2022) and BARTScore; their framing of counterfactual estimation still relies on model-based likelihood scores for computation.

The majority of prior work expresses metric performance in terms of correlation with human factuality ratings. Notably, annotations exist for subsets of the popular CNN/DailyMail (Wang et al., 2020; Fabbri et al., 2021) and XSUM summarization corpora (Maynez et al., 2020). Where Wang et al. (2020) collect user annotations from crowd workers, Fabbri et al. (2021) additionally sample expert judgments, and find that expert ratings tend to be more reliable. Maynez et al. (2020) study several aspects of summarization evaluation beyond just factuality, but do not disclose the background of annotators for evaluation.

Generally, reliably evaluating correlation of summarization metrics with human preferences is no easy task, either: Deutsch et al. (2022a) show that system-level evaluation metrics for text summarization rarely outperform simplistic metrics, such as ROUGE (Lin, 2004), to a statistically significant degree. Partially, the low confidence of improvement can be attributed to the small number of human-annotated samples available, which is due to the expensive annotation process required to obtain conclusive (and agreed-upon) annotations for factuality evaluation. Existing datasets contain a total of fewer than 2000 different instances.

3.2 Multilingual Long-Document Summarization: The EUR-Lex-Sum Dataset

When discussing current limitations, it becomes quite apparent that the diversity and availability of data resources is one of the major contributing factors. To combat this, we present a new resource of high-quality and domain-specific texts including human-written summaries for the legal domain, titled **EUR-Lex-Sum**. We detail some of the practical aspects of curating a new dataset, and manage to create a resource that has parallel availability for 24 European languages. Afterwards, we further explore performance of baselines on this dataset, and showcase that existing methods are limited in their applicability due to the average document length in EUR-Lex-Sum.

Our motivation to create a new resource stems in part from the homogeneity of existing summarization datasets and extraction processes: frequently, these are either collected from news articles (Lin and Hovy, 2002; Sandhaus, 2008; Hermann et al., 2015; Narayan et al., 2018; Grusky et al., 2018; Hasan et al., 2021) or wiki-style knowledge bases (Ladhak et al., 2020; Frefel, 2020), where alignment with supposed "summaries" is assumed over fragments from the original source documents (e.g., news bullets serving as a summary of an article on the same page). Domain outliers do exist, e.g., for scientific literature (Cachola et al., 2020) or the legal domain (Gebendorfer and Elnaggar, 2018; Bhattacharya et al., 2019; Kornilova and Eidelman, 2019; Manor and Li, 2019; Klaus et al., 2022), but are primarily restricted to the English language or do not contain finer-grained alignments between cross-lingual documents.

Reasons for the usage of mentioned predominant domains are manifold: Data is reasonably accessible throughout the internet, can be automatically extracted, and the structure naturally lends itself to the extraction of excerpts that can be seen as a form of summarization. For news articles, short snippets (or headlines) describing the gist of main article texts are quite common. Wikipedia has an introductionary paragraph that has been framed as a "summary" of the remaining article (Frefel, 2020), whereas others utilize scholarly abstracts (or variants thereof) as extreme summaries of academic texts (Cachola et al., 2020).

For a variety of reasons, using these datasets as a training resource for summarization systems introduces unwanted biases. Examples include extreme lead bias (Zhu et al., 2021b), focus on extremely short input/output texts (Narayan et al., 2018), or high overlap in the document contents (Nallapati et al., 2016). Models trained in such a fashion also tend to score quite well on zero-shot evaluation of datasets from similar domains, however, poorly generalize beyond immediate in-domain samples that follow a different content distribution or longer expected summary length.

Simultaneously, high-quality multilingual and cross-lingual data for training summarization systems is scarce, particularly for datasets including more than two languages. Existing resources are often constructed in similar fashion to their monolingual counterparts (Scialom et al., 2020; Varab and Schluter, 2021) and subsequently share the same shortcomings of low-quality alignments.

Our main contribution in this work is the construction of a novel multi- and cross-lingual corpus of reference texts and human-written summaries that extract texts from legal acts of the European Union (EU). We provide a paragraph-aligned validation and test set across all 24 official languages of the European Union[21], which further enables cross-lingual evaluation settings.

3.2.1 The EUR-Lex-Sum Dataset

Our dataset is based on available multilingual document summaries from the EUR-Lex platform. After processing, the final resource consists of up to 1,500 document/summary pairs per language. For comparable validation and test splits, we identified a subset of 375 cross-lingually aligned legal acts that are available in all 24 languages. In this section, the data acquisition process is detailed, followed by a brief exploratory analysis of the documents and their content. Finally, key intrinsic characteristics of the resource are compared with relation to existing summarization resources.

[21] https://eur-lex.europa.eu/content/help/eurlex-content/linguistic-coverage.html, last accessed: 2022-06-15

In short, we find that the combination of human-written summaries coupled with comparatively long source *and* summary texts makes this dataset a suitable resource for evaluating a less common summarization setting, especially for long-form tasks.

The EU as a Data Source

Data generated and provided by the European Union has been utilized extensively in other subfields of Natural Language Processing. The most prominent example is probably the Europarl corpus (Koehn, 2005), consisting of sentence-aligned translated texts generated from transcripts of the European Parliament proceedings, frequently used in Machine Translation systems due to its size and language coverage.

In similar fashion to parliament transcripts, the European Union has its dedicated web platform for legal acts, case law and treaties, called EUR-Lex (Bernet and Berteloot, 2006),[22] which we will refer to as the *EUR-Lex platform*. Data from the EUR-Lex platform has previously been utilized as a resource for extreme multi-label classification (Loza Mencía and Fürnkranz, 2010), most recently including an updated version by Chalkidis et al. (2019a,b). In particular, the MultiEURLEX dataset (Chalkidis et al., 2021) extends the monolingual resource to a multilingual one, however, does not move beyond the classification of EuroVoc labels. To our knowledge, the only other resource utilizing document summaries of legal acts from the platform is the monolingual English resource by Klaus et al. (2022).

Dataset Creation

The EUR-Lex platform provides access to various legal documents published by organs within the European Union. In particular, we focus on currently enforced EU legislation (legal acts) for the 20 domains from the EUR-Lex platform.[23] From the mentioned link, direct access to lists of published legal acts associated with a particular domain is available, which forms the starting point for our later crawler. Notably, each of these domains also comprises of different topics and regulations, providing a high level of diversity within the resource itself.

A legal act is uniquely identified by the so-called Celex ID, composed of codes for the respective (industry) sector, publication year and the document type. The ID is consistent across all 24 languages, which makes it possible to align articles on a document level. Across all 20 sectors, the website reports a total of 26,468 legal acts spanning from 1952 until our crawling date in 2022. However, as legal acts may be assigned to multiple domains, only about 22,000 *unique* legal acts can be extracted from the platform. We do not consider EU case law and treaties, which are also available through the EUR-Lex platform, but provided in a slightly differing document format.

[22] most recent URL: `https://eur-lex.europa.eu`, last accessed: 2023-04-15

[23] `https://eur-lex.europa.eu/browse/directories/legislation.html`, last accessed: 2022-06-21

Crawling

The web page of a particular legal act contains the following page content relevant for a summarization setting:

1. The published text of the particular legal act in various file formats,
2. metadata information about the legal acts, such as published year, associated treaties, etc.,
3. links to the content pages in other official languages, and
4. if available, a link to an associated summary document.

This work contributes to preparing a dataset with the legal act content and their respective summaries in different languages. Therefore, crawling over the entirety of published legal acts gives access to all relevant information needed to extract source and summary text pairs. Since a single legal act requires 50 individual web requests to extract files across all languages, we have a total of around 5.5 million access requests, distributed across the span of a month between May and June 2022. We dump the content of all accessed acts in a local Elasticsearch instance, and separately mark documents without existing summaries. This allows the resource to be continually updated in the future.

Filtering

For further processing, we filter the documents available through our offline storage. First, some article texts may only be available as scanned (PDF) documents, which compromises text quality and is therefore discarded. For the most consistent representation, we choose to limit ourselves to articles present in a HTML format, which provides us with additional cross-lingual paragraph alignments. Availability of HTML documents generally correlates with the publishing year, see Section 3.2.2, presumably due to the emergence of the world wide web during the 1990s. Similarly, a document is not required to have an associated summary, limiting sample availability. A full distribution of available reference/summary pairs can be found in Figure 3.4. We could not identify any particular pattern what qualifies documents for an explicit summary, but we suspect the overall importance of a legal act as a leading factor.

More problematic is the fact that between 20-30% of the available summaries (depending on the language) are associated with *several* source documents, essentially turning this into a multi-document summarization setting.[24] Since this work focuses exclusively on single document summarization, we pair the summary with the longest associated reference document to maximize

[24] We further acknowledge that legal acts frequently reference *external* knowledge as well, in the form of hyperlinks. A comprehensive inclusion of this background information is not considered in our setup, and thus implicitly also assumes some "world knowledge" for a perfect system. On the other hand, we argue that the *core contents* (constituting the eventual legal act summary) are likely to be stated explicitly in the document itself, thereby drastically reducing the severity of this issue.

	n-gram novelty			
Subset	1-gram	2-gram	3-gram	4-gram
All samples	42.25	64.07	77.34	83.73
Single-reference subset	41.74	63.52	76.87	83.33
Longest available document	46.77	68.83	81.44	87.18
Concatenated documents	41.03	63.06	76.38	82.77

Table 3.2: Comparison of n-gram novelty for the English subset, depending on reference document processing. *Longest available subset* considers samples with multi-reference documents but only taking the content of the longest document as a reference; *Concatenated documents* uses the concatenation of all associated references. In comparison, *single-reference subset* refers to document that naturally only have a singular source document.

availability. Table 3.2 details the impact of considering only the longest document in terms of n-gram novelty; we observe a consistent increase of novel n-grams by about 5 percentage points over the subset of single-reference documents. While the concatenation of all associated reference documents would eliminate any difference in n-gram overlap between the summary and reference texts, having a single reference document conserves the correct order of processing and avoids the artificial mitigation of implicit lead biases in the text. Further, concatenation leads to ambiguous text orderings, which may change summarization outcomes based on different aggregation strategies. However, the subset of these multi-document samples could be a challenging extension based on our available corpus that may be explored in future work. Finally, we filter out all document pairs where the reference text is shorter than the input document. This occurs only for multi-document summary pairs, where sometimes several short acts are aggregated into a single summary.

After filtering out invalid samples, between 391 (Irish) to 1,505 (French) documents remain; the full list of samples broken down by language can be found in Table 3.6. Across all languages, we manage to extract a total of 31,987 reference/summary pairs.

Data Split

To ensure a suitable (and comparable) validation and test split across different languages, all documents having sample pairs available in 24 languages (375 total) are taken out of the available respective subsets. Of the 375 documents, 187 samples are randomly divided into a validation set, and the remaining 188 as our test set. All other documents are assigned to the language-dependent training sets. No guarantee for cross-lingual availability is provided for the training set, however, most documents do appear in several of the languages. We will use these filtered data splits for all experiments, unless explicitly mentioned otherwise.

3.2.2 Exploratory Analysis

An exploratory analysis of the dataset is conducted to confirm the resource's viability for automatic summarization and overall data quality. Aside from a qualitative view of the resource and an analysis of the temporal distribution of our samples, we provide a comprehensive look at intrinsic metrics commonly used for summarization datasets.

Data Quality

Documents of the EU are generally held to a high standard, and the legal acts are no exception. This also extends to the summaries, which follow a particular set of guidelines for their creation process.[25] In particular, guidelines for drafting summary texts are detailed in Technical Annex I, which specify several key instructions for generating human-written summaries of an underlying legal act. Most prominently, they recommend a target length for key point summaries between 500-700 words and formulate a template structure for the overall text outline. An example of a typical summary structure can be seen in Figure 3.3. Aside from the key points, this includes, e.g., references to the main documents or specific act-related key phrases. We want to highlight that the generation guidelines changed over time. Since we do not have access to previous versions of the guidelines, we manually probed comparisons between older and newer documents, which exposed a highly similar structure despite changes in guidelines.

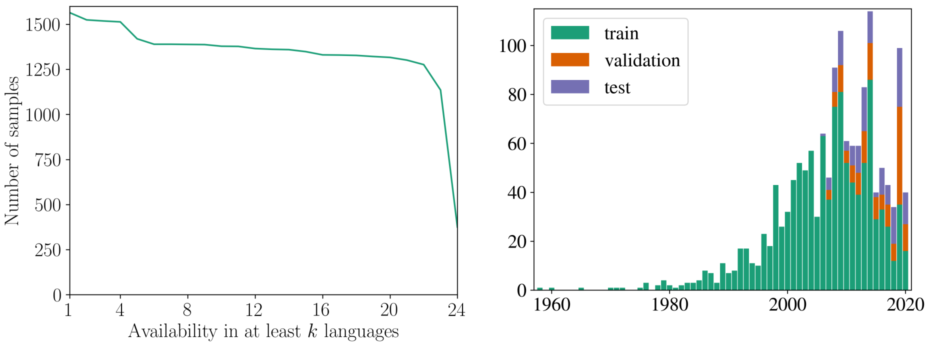
The published documents and summaries offer further peculiarities in both their content structure as well as the creation process: First, the multilingual versions of both documents and summaries are always translated from the original English legal act (or English summary thereof),[26] which ensures strict content similarity of the same text across all available languages. Second, due to their HTML representation, it is possible to extract *paragraph-aligned* texts between language-specific versions. This is a well-known property of EU-level data, most notably exploited in the Europarl corpus (Koehn, 2005) for automatic alignments of machine translation training data. We similarly maintain this alignment structure during our extraction process in order to make it available for later stages, e.g., for specific evaluation setups or cross-lingual pre-training.

Temporal Distribution

Figure 3.1 displays the distribution of filtered documents by year of publication. The amount of available samples increases after 1990, which likely coincides with more member states joining, as well as a shift to digital archiving (compared to OCR scans of PDF documents, which are excluded from our corpus). Compared to other European resources, such as Multi-EURLex Chalkidis

[25] `https://etendering.ted.europa.eu/cft/cft-documents.html?cftId=6490`, last accessed: 2022-06-15

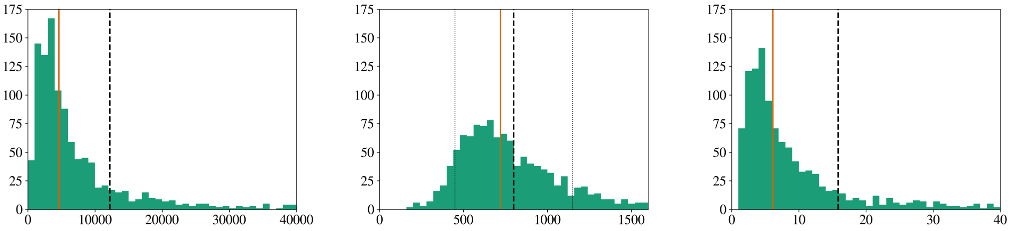
[26] This has been confirmed by the Publications Office of the European Union in private correspondence.



(a) Language availability

(b) Temporal distribution

Figure 3.1: (a) Cross-lingual availability of individual documents. The vast majority of samples is available in at least 20 languages. (b) Distribution of the publishing year of unique legal acts included in the final dataset. The number of parsed documents increases after 1990.



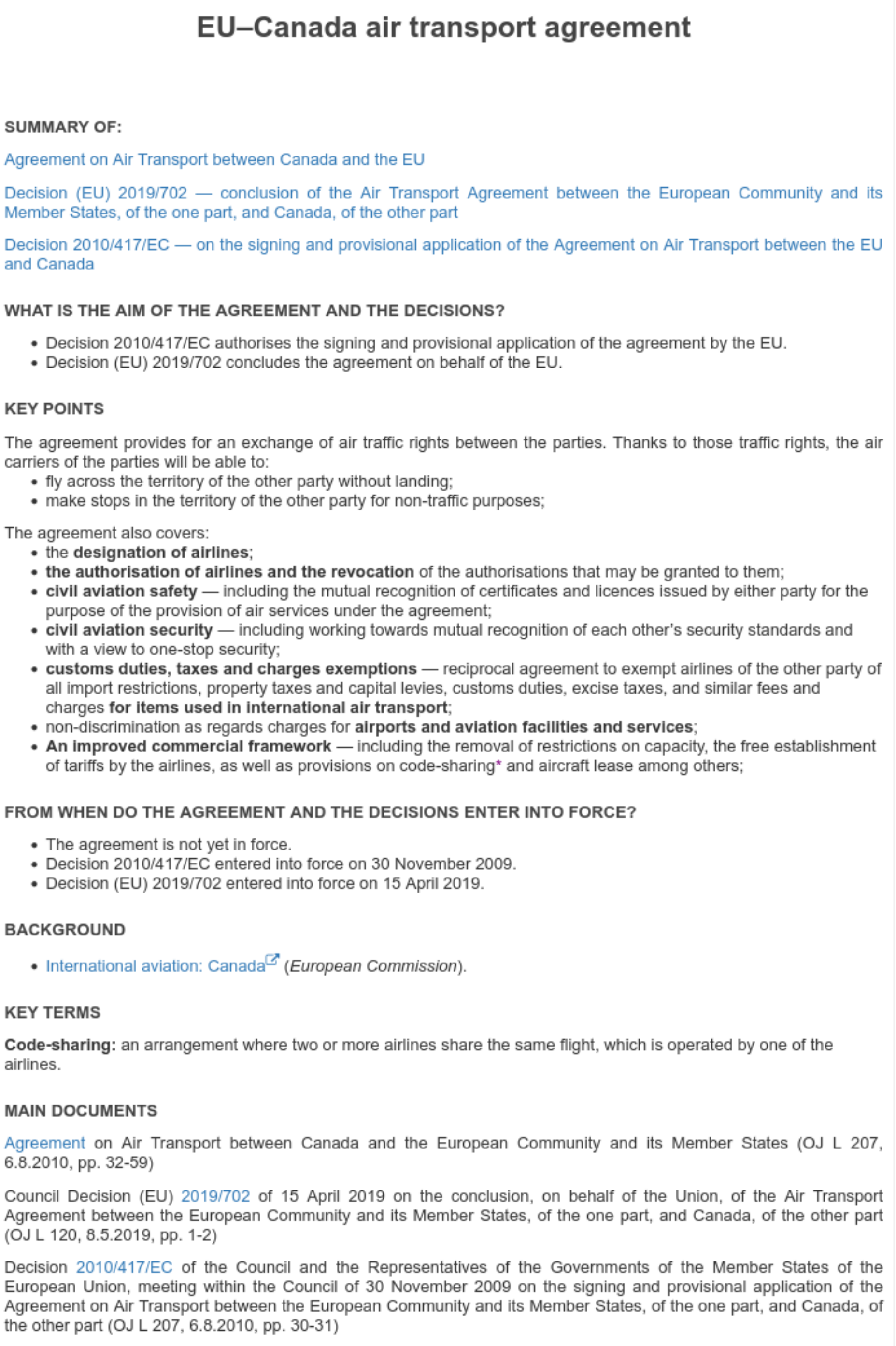
(a) Reference tokens

(b) Summary tokens

(c) Token compression ratio

Figure 3.2: Histogram of the English training set, comparing article token lengths. Displayed are the distribution for references (left), summaries (center), and compression ratios (right). Vertical lines show median length (continuous orange), mean length (dashed black), and standard deviation (dotted black lines). The latter exceeds display limits for reference length and compression ratio. The x-axis range is limited to the 95th length percentile for legibility in all plots.

et al. (2021), a lesser topical shift is expected, simply due to a more limited time frame. Notably, we also include the distribution by dataset split and observe an even stronger bias towards more recent legal acts for validation and test sets. This is a natural consequence of the requirement for validation and test sets that legal acts be present in all 24 languages, which includes more recent additions, such as Croatian (added in 2013) or Irish (added in 2022). We also want to mention that amendments to both reference and summary texts might be added (or revised) several years after their original publication, which is not reflected in our analysis.



EU–Canada air transport agreement

SUMMARY OF:

Agreement on Air Transport between Canada and the EU

Decision (EU) 2019/702 — conclusion of the Air Transport Agreement between the European Community and its Member States, of the one part, and Canada, of the other part

Decision 2010/417/EC — on the signing and provisional application of the Agreement on Air Transport between the EU and Canada

WHAT IS THE AIM OF THE AGREEMENT AND THE DECISIONS?

- Decision 2010/417/EC authorises the signing and provisional application of the agreement by the EU.
- Decision (EU) 2019/702 concludes the agreement on behalf of the EU.

KEY POINTS

The agreement provides for an exchange of air traffic rights between the parties. Thanks to those traffic rights, the air carriers of the parties will be able to:

- fly across the territory of the other party without landing;
- make stops in the territory of the other party for non-traffic purposes;

The agreement also covers:

- the **designation of airlines**;
- **the authorisation of airlines and the revocation** of the authorisations that may be granted to them;
- **civil aviation safety** — including the mutual recognition of certificates and licences issued by either party for the purpose of the provision of air services under the agreement;
- **civil aviation security** — including working towards mutual recognition of each other's security standards and with a view to one-stop security;
- **customs duties, taxes and charges exemptions** — reciprocal agreement to exempt airlines of the other party of all import restrictions, property taxes and capital levies, customs duties, excise taxes, and similar fees and charges **for items used in international air transport**;
- non-discrimination as regards charges for **airports and aviation facilities and services**;
- **An improved commercial framework** — including the removal of restrictions on capacity, the free establishment of tariffs by the airlines, as well as provisions on code-sharing* and aircraft lease among others;

FROM WHEN DO THE AGREEMENT AND THE DECISIONS ENTER INTO FORCE?

- The agreement is not yet in force.
- Decision 2010/417/EC entered into force on 30 November 2009.
- Decision (EU) 2019/702 entered into force on 15 April 2019.

BACKGROUND

- International aviation: Canada (*European Commission*).

KEY TERMS

Code-sharing: an arrangement where two or more airlines share the same flight, which is operated by one of the airlines.

MAIN DOCUMENTS

Agreement on Air Transport between Canada and the European Community and its Member States (OJ L 207, 6.8.2010, pp. 32-59)

Council Decision (EU) 2019/702 of 15 April 2019 on the conclusion, on behalf of the Union, of the Air Transport Agreement between the European Community and its Member States, of the one part, and Canada, of the other part (OJ L 120, 8.5.2019, pp. 1-2)

Decision 2010/417/EC of the Council and the Representatives of the Governments of the Member States of the European Union, meeting within the Council of 30 November 2009 on the signing and provisional application of the Agreement on Air Transport between the European Community and its Member States, of the one part, and Canada, of the other part (OJ L 207, 6.8.2010, pp. 30-31)

Figure 3.3: Excerpt of a short legal act (Celex ID 32019D0702). Visible are several distinct sections, with the majority of the document describing key points of the underlying legal acts. This particular summary aggregates content from several legal acts, of which we consider the longest one as the reference document.

	Ref tokens		**Summa tokens**		**Comp.**	**% novel n-grams in summary**			
	Min	**Max**	**Min**	**Max**	**Ratio**	**1-gram**	**2-gram**	**3-gram**	**4-gram**
Train	385	1,087,217	173	3021	16 ± 62	44.10	65.97	78.85	84.96
Val	1,143	199,405	354	5136	18 ± 17	36.65	58.23	72.74	79.96
Test	1,544	403,319	369	2987	18 ± 20	36.78	58.46	72.83	80.07

Table 3.3: Intrinsic dataset properties for the English subset splits. We report minimum and maximum token lengths of both reference texts and summaries in the data, as well as compression ratio of article pairs. Further listed are novelty n-gram shares in the gold summary based on whitespace tokenization.

Document Structure

An example of the content structure of a document summary is provided in Figure 3.3. The formatted text reveals cleanly separated sections of the summary, where the main content is usually a free-form text describing the key goals and highlights of the sub-points within a longer legal act. Other sections within the summary further describes which legal act (or several acts) are associated to this proposal. As previously described, we limit ourselves to linking the longest associated legal act for a summary referencing several acts.

While we provide raw text for the extracted legal act document in the proposed resource, example document in Figure 3.3 reveals a potential use case of semi-structured visual information from HTML tags (e.g., headline descriptors or bullet lists), which could be used for a fine-grained distinction between different content parts. In our preliminary experiments, we found that the used HTML tags for content elements can vary significantly between different legal acts (e.g., using modified `div` containers instead of `H3` for sub-headings) and therefore keep the inclusion of such features for future work.

Summarization-related Dataset Metrics

We adopt metrics from prior work to automatically analyze summarization datasets (Grusky et al., 2018; Zhong et al., 2019; Bommasani and Cardie, 2020). Our corpus reveals a high degree of abstractivity, which is surprising given the enormous length of input texts.

Length Distribution Based on the fact mentioned in Section 3.2.2 that documents are created as translations from the English original, we focus more on the distribution of legal acts and their summary lengths in English as a representative language. A more exhaustive overview can be found in Table 3.6, which gives more insight into language-specific length variations due to document availability, or simply morphological/syntactic differences, e.g., compound words.

Histogram plots in Figure 3.2 show a Zipfian distribution for reference text lengths, with a mean of around 12,000 tokens; however, exceptionally large standard deviation due to extreme outliers are present in the data, as mentioned in Table 3.3. In contrast, summary lengths exhibit closer to a normal distribution, which matches the suggested length of 500-700 words mentioned in the document guidelines. The observed mean is slightly higher at around 800 tokens, which includes other sections in the summary that refer to key phrases and document metadata and are not counted towards the actual summarizing content by annotators. One can similarly observe extreme outliers for summary text lengths, see Table 3.3.

Compression Ratio Comparing compression ratios reported by Zhong et al. (2019) reveals that EUR-Lex-Sum has a mean compression ratios similar to news-based summarization datasets, e.g., CNN/DailyMail (Hermann et al., 2015) or the NYT Corpus (Sandhaus, 2008).

n-gram Novelty To provide insight into the abstractiveness of gold summaries, we follow Narayan et al. (2018) in analyzing the fraction of n-grams not present in the original reference article. This metric is similar to content coverage metrics used by Grusky et al. (2018) or Zhong et al. (2019). When comparing novelty n-grams reported in Table 3.3, reported scores likely overestimate the realistic n-gram novelty slightly. This can be attributed to the use of whitespace tokenization, which can lead to n-grams being processed slightly differently due to the decreased tokenization accuracy. We further discuss the choice of tokenization in Section 3.2.3.

3.2.3 Experiments

As a reference for future work building on top of this dataset, we provide a set of suitable baselines and discuss limitations of methods and data. Notably, there are considerable challenges in constructing baseline runs with popular algorithms on this dataset particularly with respect to linguistic coverage.

Primarily, even just the length of a gold summary exceeds input limitations of popular abstractive neural models; as previously discussed, systems are generally limited to 512 (subword) tokens (Lewis et al., 2020a; Xue et al., 2021), and even length-focused alternatives generally boast only up to 4096 tokens (Beltagy et al., 2020; Zaheer et al., 2020), which is well below the median length of reference texts and prevents us from evaluating systems without further chunking the input text.

Less obvious, but no less problematic is the availability of tokenizers or sentence splitting methods in popular NLP libraries, affecting several lower-resourced languages in our corpus (for a more in-depth list of supported languages by library, see Table 3.6). This inherently prevents fair sentence-level evaluation (or extraction), as system performance is not guaranteed for underrepresented languages. Aside from a set of extractive baselines, we further evaluate a cross-lingual scenario in which summaries for the English reference text are generated and then translated into the target

	Validation			Test		
	R-1	R-2	R-L	R-1	R-2	R-L
English	25.99	13.34	13.30	26.68	13.65	13.58
French	32.18	18.03	15.15	32.35	18.00	15.16
German	26.00	13.12	12.24	26.72	13.75	12.56
Spanish	27.04	16.43	14.75	28.34	17.12	15.23
Italian	27.29	14.01	12.63	28.57	14.24	12.90
Portuguese	30.12	17.17	15.08	30.67	17.20	15.20
Dutch	29.07	14.92	14.66	29.62	14.76	14.73
Danish	28.78	13.90	13.14	29.22	13.86	13.19
Greek	24.42	9.77	15.46	24.79	9.45	15.46
Finnish	26.40	11.88	11.87	26.49	11.68	11.80
Swedish	30.25	15.40	14.27	30.67	15.47	14.35
Romanian	35.69	16.08	14.90	34.75	15.16	14.59
Hungarian	33.71	19.53	15.49	34.55	19.69	15.64
Czech	30.96	16.65	14.16	31.86	16.76	14.32
Polish	28.47	14.42	12.68	28.88	14.42	12.73
Bulgarian	26.36	9.15	16.54	25.58	8.40	16.13
Latvian	31.24	15.55	12.99	31.73	15.77	13.15
Slovene	26.75	12.25	11.64	27.19	12.34	11.79
Estonian	26.33	11.64	11.84	26.39	11.41	11.66
Lithuanian	26.79	12.43	11.44	26.76	12.45	11.59
Slovak	30.30	15.04	13.14	30.65	14.94	13.14
Maltese	29.71	14.55	12.73	30.51	14.62	12.86
Croatian	33.50	13.46	13.50	32.64	12.76	13.29
Irish	43.66	18.72	15.86	41.93	17.16	15.25

Table 3.4: Extractive summarization baseline using our modified LexRank-ST approach. We report ROUGE F1 scores for both the validation and test splits. We reiterate that scores are not comparable between languages, and should rather be seen as a baseline set of metrics to compare with other language-specific results.

languages. The hypothesis is that this provides insight into limitations of existing cross-lingual summarization systems discussed in Section 3.1.1 and also represents more realistic deployment scenarios where cross-lingual systems can be utilized as supportive summarizers for monolingual input texts.

Zero-shot Extractive Baselines

We utilize a modified variant of LexRank (Erkan and Radev, 2004b) that uses multilingual embeddings generated by sentence-transformers (Reimers and Gurevych, 2019, 2020) to compute centrality, see Section 2.4.1. Given the previously mentioned limitations of sentence splitting of input texts, we chunk the text based on existing paragraph separators (refer to Figure 3.3), and treat those segments as inputs to our baseline setup. Notably, this method does not require any form of fine-tuning or language adoption and works as a zero-shot domain transferred extractive model, which makes it preferable over methods such as SummaRuNNer (Nallapati et al., 2017) or extractive BERT summarizers,[27] which require training on (automatically extracted) alignments. To determine the output summary length, we calculate the average paragraph-level compression ratio on the language's as described in Section 2.4.1, and use it in conjunction with the number of paragraphs of the current input text to estimate a target number of paragraphs for the summary. For evaluation, we rely on ROUGE scores (Lin, 2004) with disabled stemming to conserve comparability between languages. We acknowledge that this is not a comprehensive measure according to our prior discussions and has distinctive shortcomings. However, it works as a language-agnostic baseline and has comparatively high signals at the paragraph level, as such units generally preserve both factual consistency and fluency.

Due to the paragraph-level consistency of generated summaries, this is a fairly strong baseline. Importantly, ROUGE scores remain consistent for languages between the validation and test set, although we do observe some languages with outlier performance: For Greek text, the model likely struggles with the representation of non-Roman alphabets, but still performs decently well at the ROUGE-L level. Otherwise, Irish has unexpectedly high ROUGE scores, which we were unable to explain. This is especially surprising given the fact that the language is not even one officially supported by the multilingual embedding model used for this experiment.

Cross-lingual Baselines

As a baseline for future cross-lingual experiments, we provide a simple two-step translate-then-summarize pipeline (Wang et al., 2022c). To generate summaries on longer contexts, we utilize a model based on the Longformer Encoder Decoder (LED) architecture (Beltagy et al., 2020), precisely a checkpoint previously fine-tuned on the English BillSum corpus (Kornilova and Eidelman, 2019). Translation from English to target languages is done with OPUS-MT (Tiedemann and Thottingal, 2020). To deal with long documents exceeding the particular model's window size, we greedily chunk text if necessary. To represent an upper limit of performance, we compare a translate-then-summarize setup from English to Spanish, which can be regarded as one of the language pairs with the highest MT performance, due to data availability and linguistic similarity of the source and target language.

As baselines, we provide translations of the English gold summaries into the target language (again with the Opus MT model), as well as a translation of the extractive LexRank summary from the previous experiment. Results seen in Table 3.5 are surprising: While the abstractive model seems to improve over the purely Spanish-based LexRank summary (LexRank-ES) by a significant mar-

[27] e.g., `https://pypi.org/project/bert-extractive-summarizer/`, last accessed: 2024-04-05

	Validation			Test		
	R-1	R-2	R-L	R-1	R-2	R-L
LED	31.67	13.00	16.17	31.14	13.01	16.20
LexRank-EN	39.42	20.03	18.53	39.44	20.02	18.73
LexRank-ES	27.04	16.43	14.75	28.34	17.12	15.23
Oracle	52.84	39.79	43.87	54.55	41.01	45.06

Table 3.5: Cross-lingual summarization setup for English-Spanish. We report ROUGE F1 scores for both the validation and test splits on the Spanish subset. The LED model has input length limitations at around 16k tokens, whereas our LexRank-based methods can handle any length. "LexRank-EN" translates a summary generated in English to Spanish. "LexRank-ES" uses the Spanish input text and directly generates a Spanish summary. "Oracle" is the highest possible ROUGE score achievable if selecting the best possible input paragraphs.

gin, it turns out that translating the English LexRank (LexRank-EN) baseline drastically *improves* results in terms of ROUGE scores. We assume that this is related to truncation and re-phrasing happening during the translation step.

Open Problems

The most obvious problem for this dataset is the extreme length, and also length disparity between documents. This is especially apparent when comparing the length to average samples in CNN/DailyMail Hermann et al. (2015), where the mean article length is about 16 times shorter; this makes content selection significantly more challenging. Secondly, incorporating hierarchical information about the reference text could greatly improve context relevance in such extensive settings. However, this is not only restricted to the reference document, but could also be considered for the (hierarchical) construction of long-form summary texts. Given that previous datasets do not come with such long output samples, this has to our knowledge not been previously tackled in the literature. Ultimately, the question of equal coverage for lesser-resourced languages is also not fully answered. While we attempt to treat languages in our dataset equally, this comes with its particular set of challenges and we acknowledge that better support for rare languages is needed.

3.2.4 Conclusion and Future Work

We contribute a new multi- and cross-lingual resource for text summarization research, and expose some of the limitations of existing approaches in handling long legal document contexts. We further provided a more detailed analysis of the underlying data and sample quality and hypothesized potential applications to open problems in the communtiy, such as long-form summarization or cross-lingual application scenarios. Our dataset is publicly available on the web, and comes with a set of monolingual extractive baselines that provide suitable reference points for any future work

	No. articles		**Availability**			**Avg article length (token)**		**Comp**	**n-gram novelty**	
Language	**before**	**after**	**S-T**	**spacy**	**nltk**	**Reference**	**Summary**	**ratio**	**1-gram**	**2-gram**
English (en)	1,974	1,504	✓	✓	✓	12206 ± 42429	799 ± 349	16 ± 62	44.10	65.97
French (fr)	1,969	1,505	✓	✓	✓	13192 ± 43950	892 ± 395	16 ± 63	45.07	64.13
German (de)	1,966	1,490	✓	✓	✓	11144 ± 41061	748 ± 330	16 ± 68	44.85	66.95
Spanish (es)	1,964	1,487	✓	✓	✓	13581 ± 44574	932 ± 420	15 ± 57	44.76	61.51
Italian (it)	1,867	1,403	✓	✓	✓	13152 ± 44641	845 ± 370	16 ± 67	44.77	67.00
Portuguese (pt)	1,845	1,376	✓	✓	✓	12629 ± 29921	896 ± 391	14 ± 38	43.84	64.00
Dutch (nl)	1,844	1,376	✓	✓	✓	13233 ± 44638	834 ± 362	17 ± 69	44.41	65.86
Danish (da)	1,843	1,377	✓	✓	✓	11947 ± 43155	717 ± 308	18 ± 71	46.96	68.27
Greek (el)	1,837	1,366	✓	✓	✓	13609 ± 45411	863 ± 369	17 ± 64	44.86	66.70
Finnish (fi)	1,825	1,366	✓	✓	✓	9792 ± 41021	575 ± 247	18 ± 93	53.41	77.26
Swedish (sv)	1,822	1,362	✓	✓	✓	10796 ± 26923	718 ± 305	15 ± 40	46.74	69.62
Romanian (ro)	1,817	1,353	✓	✓	✗	13646 ± 45644	826 ± 356	17 ± 67	45.42	67.80
Hungarian (hu)	1,813	1,336	✓	✗	✗	12230 ± 46764	702 ± 298	19 ± 84	53.23	75.68
Czech (cs)	1,812	1,359	✓	?	✓	12469 ± 46640	715 ± 307	18 ± 77	46.75	71.89
Polish (pl)	1,811	1,353	✓	✓	✓	11560 ± 33296	739 ± 324	16 ± 48	46.69	71.01
Bulgarian (bg)	1,792	1,332	✓	✗	✗	13397 ± 45578	819 ± 350	17 ± 69	47.00	68.44
Latvian (lv)	1,790	1,334	✓	?	✗	11841 ± 46552	670 ± 289	19 ± 83	50.23	74.55
Slovene (sl)	1,789	1,332	✓	✗	✓	11357 ± 32842	712 ± 305	16 ± 48	47.28	71.57
Estonian (et)	1,788	1,332	✓	✗	✓	10778 ± 45157	581 ± 249	20 ± 94	52.20	77.46
Lithuanian (lt)	1,788	1,335	✓	?	✓	11943 ± 46673	669 ± 290	19 ± 88	47.79	74.00
Slovak (sk)	1,788	1,325	✓	?	✗	11600 ± 32968	729 ± 319	16 ± 47	48.20	73.42
Maltese (mt)	1,770	1,315	✗	✗	✗	12711 ± 48156	685 ± 299	20 ± 85	54.77	81.43
Croatian (hr)	1,762	1,278	✓	?	✗	10051 ± 19390	712 ± 307	14 ± 28	48.62	72.61
Irish (ga)	427	391	✗	?	✗	28152 ± 63360	948 ± 385	46 ± 137	45.89	70.38

Table 3.6: Supplementary statistics of the EUR-Lex-Sum corpus across languages. We list the number of available articles (before and after filtering), and whether a particular language is supported by `sentence-transformers` multilingual models ("*S-T*"), or has available language-specific processors in `spaCy` Montani et al. (2023) or `nltk` Bird et al. (2009), respectively. "?" indicates potential support through general-purpose models with uncertain segmentation quality. We also provide abridged statistics similar to Figure 3.2 and Table 3.3 for each language's training partition.

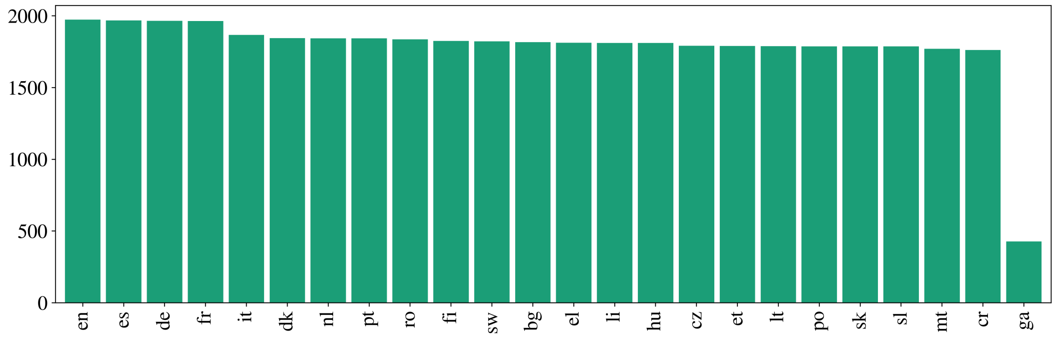


Figure 3.4: The number of all crawled document/summary pairs across the 24 official EU languages *before* filtering. Irish has only recently been added as an official language and thus has fewer documents available.

in this direction.

Potential expansions in future work include the exploitation of structural elements in the original HTML code of the summary texts to generate more guided target summaries for particular sections. Especially for extremely long legal texts, templated generation could also be utilized to improve the uniformity of generations. Given that it is not possible to train any models to respect the specific output format in that way, templates or few-shot prompts for expected output structures may improve the results. On a more general level, we expect that progress in long-form models is required to achieve remotely sensible results on extreme-length generative tasks with neural models. Even with more extended context windows becoming available in chat systems, the longest documents within EUR-Lex-Sum would still exceed the commercially available limits (currently around 128k tokens, versus the longest document being over 1 million tokens long).

3.3 Usability of Existing Summarization Systems

So far, we have taken a macro-level view of summarization systems through the lens of data issues, particularly revolving around the limited availability in non-English setups. To switch perspectives, we want to use this section to investigate some of the more narrow problems encountered in the space of text summarization, and elaborate on the key limitations that we encounter here. In Section 3.3.1, we hypothesize about the impact of data quality issues in existing summarization corpora on downstream performance, which we later substantiate in Section 3.4. The central arguments in our opinion are the limited variance in length and domain exposure due to the uniformity of existing summarization datasets. Coupled with the limited data quality standards for automatically extracted text summarization resources, this leads to subpar training results and limited generalization of systems. Evaluating models touted as "generic summarization models" on data that is even slightly out-of-domain already reveals critical failures (Section 3.3.2).

3.3.1 Data Quality Issues in Summarization Corpora

Given the data-hungry nature of Machine Learning approaches that have been popularized in the Natural Language Processing community due to their strong results, we have seen an increased interest in the curation of new training resources for this purpose, also in the summarization community. We have witnessed the creation of various new corpora, for example in the news domain (Narayan et al., 2018; Grusky et al., 2018), community knowledge website (Koupaee and Wang, 2018), for scientific abstract summarization (Cohan et al., 2018; Cachola et al., 2020), legal use cases (Kornilova and Eidelman, 2019) and even several multilingual extensions discussed in previous sections (Ladhak et al., 2020; Scialom et al., 2020; Aumiller et al., 2022b).

Since the majority of these datasets contain data that has been obtained following a pre-defined set of extraction rules,[28] most corpora are obtained in an unsupervised fashion. This in itself is not necessarily a problem, **if** necessary precautions are taken to ensure a high quality in the resulting corpus. Realistically, however, most authors skip even on simple checks to verify the data quality, subsequently leading to a poor data sanity in the respective resource. We argue that there exist several problem dimensions in the current versions of summarization corpora, data quality being one of the most obvious ones. For related domains, for example, previous work has also shown the existence of outright label errors (Northcutt et al., 2021). It is reasonable to assume that these problems exist in summarization corpora as well, which we show in later sections, see Section 3.4. Unfortunately, identifying issues related to the *semantic* relationship of input and output text snippets is arguably much harder to perform automatically, but remains out of scope for most parts in this work.

3.3.2 Domain Adaptability of Existing Solutions

One key problem of current research work is the unwillingness to explicitly state the nature of domain-specific vs generic summarization systems. Of course, the holy grail of any research work is to present a summarization system that is capable of delivering perfect summaries across a *wide variety of settings*. Realistically, the contributions of individual works rather demonstrate an improvement on a select choices of evaluation datasets (or, more often than not, a singular dataset). Despite this, the general storyline often talks about improving on the task of "summarization" as a whole, which can be interpreted as a misleading statement.

[28] There are a few notable exceptions that curate data manually, e.g., the SQuALITY corpus by Wang et al. (2022a) and (to a lesser extent) the aforementioned SciTLDR corpus of Cachola et al. (2020).

	MLSUM			Klexikon		
Model	**R-1**	**R-2**	**R-L**	**R-1**	**R-2**	**R-L**
lead-*k*	15.08	4.17	11.45	28.34	5.50	12.50
mrm8488/bert2bert	44.05	33.44	40.36	15.41	3.26	9.30
ml6team/mt5-small	28.51	19.52	26.53	13.45	2.98	8.49

Table 3.7: ROUGE F1 scores for the test set of in-domain evaluation sets (MLSUM Test) and out-of-domain data (Klexikon Test). Models fail to outperform our simple lead-based heuristic.

We have already demonstrated that the requirements for a summary can hugely vary depending on the task and user context, see Section 2.5. As an illustration, during a student research project led by Jing Fan, we have conducted a series of experiments to test the domain generalization of two German summarization systems listed in Table 3.1: mrm8488/bert2bert and ml6team/mt5-small. Both of these systems were trained on MLSUM (Scialom et al., 2020), with the ml6team additionally training on English CNN/DailyMail (Nallapati et al., 2017).

Table 3.7 shows the ROUGE F1 scores illustrating a relative performance difference across corpora. While the expressiveness of ROUGE scores itself is limited of course, it already showcases a grave problem for both neural models. The domain focus on news texts translates extremely poorly to the Klexikon data that is considerably longer in its summary. In a preliminary qualitative analysis, we further noticed that the stylistic properties of generated text largely depend on the training corpus as well. In that case, having a model specifically fine-tuned on, e.g., headline summarization tasks will ultimately lead to models generating exactly this kind of style – irrespective of the actual inputs being a news article or not. Fine-tuning with a broader and more diverse pairing of input and output texts seems the most obvious way to combat this issue, which again requires better datasets.

In the meantime, we advocate that researchers should accurately state the full context of what their provided models were trained on and thus are likely to perform well with. Understandably, stating that a model *"can accurately summarize short news articles"* instead of presenting a *"generic summarization system"* sounds less impressive to reviewers, but would allow users to better know what quality to expect.[29] We later also discuss this under the context of missing baselines and limitations of the evaluation metrics in Section 3.4.3 and demonstrate that there are certain setup choices that affect the interpretation of results from the perspective of evaluation metrics as well.

[29] We acknowledge that any proposed changes to model architectures might well be able to solve more generic summarization problems and deliver promised improvements. However, unless empirical evidence is shown that gives credibility to these claims, we remain firmly of the opinion that such statements are over-generalizing from limited evaluations.

Issue	Reference	Summary
Short text	Wir verwenden Cookies, um unser Angebot für Sie zu verbessern. Mehr Informationen [...].	–
Duplicates	‘Virtuelles Bergsteigen mit dem Project360 [...]	Leben und Kultur in Europa
	Historische Dokumente: Bilder der Wende [...]	Leben und Kultur in Europa
Relative Length	Chef-Sprüche: “Ich sehe meine Kinder auch nur im Urlaub.”	Die besten Chef-Sprüche zum Thema Überstunden.
Extractiveness	Neuigkeiten aus dem Berliner Zoo: Der Berliner Zoo [...]	Neuigkeiten aus dem Berliner Zoo.

Figure 3.5: Categories of error types for summarization samples. Examples for the first three types are directly taken from erroneous instances contained in the MassiveSumm dataset. Despite not directly measuring the semantic quality of samples, we notice a trend where filtered instances are of especially low semantic quality, too.

3.4 Quantifying Data Quality Issues for Summarization

With the established conclusion that data quality issues *are*, in fact, a prevalent issue in text summarization research, and likely impact the resulting performance of systems trained on such corpora, we set out to investigate methods to automatically detect and remove low-quality data instances from existing resources. We introduce a series of simple heuristics for the automated filtering of text summarization datasets, which require no additional language-specific software and are thus widely applicable. To categorize prevalent issues in existing datasets, we subsequently perform a more detailed analysis of filtering results on German summarization data. By first filtering evaluation splits of popular datasets and then re-evaluating the performance of existing models on these subsets, we are finally able to give more conclusive evidence whether limited model performance is due to failed generalization, or a lack of data care during the pre-processing. Our results indicate that the latter scenario is the prevailing cause for poor performance of German summarization models.

3.4.1 Automated Metrics for Dataset Filtering

The best strategy to achieve decent experimental results is ensuring high quality in the training data – in line with the popular Machine Learning wisdom of “garbage in, garbage out”. We present a number of data quality checks for individual samples, as well as a way to address the problem of potential data contamination, particularly for text summarization datasets. These measures are fully automated and at most require single hyperparameter settings to filter a dataset. We again demonstrate their performance at the example of filtering German summarization datasets, which we previously introduced in Section 3.1.1.

Intrinsic Dataset Metrics

We propose that intrinsic dataset metrics (Narayan et al., 2018; Grusky et al., 2018; Zhong et al., 2019; Bommasani and Cardie, 2020) should more regularly be used as a preliminary gauge for text quality in comparison to the original input. They are especially useful to estimate *abstractiveness* of summaries, essentially constituting the number of novel n-grams in summaries. This generally indicates a change in the used vocabulary between references and summaries, and as a proxy can also serve to detect summaries that are completely unrelated to the reference (e.g., because they only mistakenly have been aligned). We notice this issue particularly for web-scale datasets, such as MassiveSumm. However, there are also several shortcomings with existing intrinsic metrics, such as the semantic coherence score introduced by Bommasani and Cardie (2020), which essentially measures the likelihood over all segments and their predecessors in a summary. Similarly to the previously discussed shortcomings of many summarization systems, this coherence metric relies on a readily available and pre-trained system for the English language which are not given for many other languages and also expensive to run. We aim for a more fundamental list of checks that are even easier to compute than metrics like the semantic coherence[30] that will allow dataset creators to verify the basic utility and quality of their datasets.

Minimal Length and Empty Rows

The most trivial sanity check is verifying that *both* the reference text and summary are non-empty for all samples. While this is the most prevalent check implemented by other authors, in our experience we still find some cases where samples remain invalid. This is especially the case when we compare the varying definitions of "emptiness". For example, one could consider a sample empty, even if whitespaces (or whitespace-like symbols, such as `\t`) are present, but no other content symbols. Extensions are, for example, faulty encodings or only special characters in a text (cf., data audit insights by Kreutzer et al. (2022)).

As a superset of "empty samples", we may also impose a required minimum text length, which presents a slightly stricter filtering criterion for sample validity. Where empty texts are universally to be avoided, hard length requirements are difficult to determine, since the appropriate cutoff depends strongly on the dataset domain. For domain-specific datasets, e.g., the instruction-like texts in the WikiLingua dataset Ladhak et al. (2020), having extremely short summaries with only a few characters (and comparatively short references) may make sense. For summaries stemming from news articles, however, length requirements imposed on the reference might ensure a longer minimum text length for quality control. Importantly, however, we generally want to impose a slightly longer minimum length for the input texts compared to the summaries.

[30] More likely approaches like this measure something closer to "textual fluency", which is not necessarily relevant for specific summary styles, e.g. bullet point summarization.

Compression Ratio

Another key metric we previously introduced is the *Compression Ratio (CR)*, defined as the relation between reference text length and summary length. We follow the definition by Grusky et al. (2018), see Equation (2.16), which is the inverse fraction of the similarly named metric introduced by, e.g., Bommasani and Cardie (2020).

By using character-level length estimates, we avoid several issues at the same time: Compared to the usage of a token-level length estimators, evaluating the number of characters of a segment (or several ones) is trivial to implement in any language and relatively comparable. Tokenization, on the other hand, may be much harder to implement. Using a crude whitespace tokenization approach gives relatively decent approximations for English and other European languages, but may still fall short for any language using a different alphabet, such as Japanese or Chinese. Evaluating the length in sentences is also not necessarily a reliable evaluation metric in the context of summarization, where segment-level changes, such as sentence-splitting (distribution of information across multiple segments) or deletion of sub-phrases may affect the density of content.

For exemplary purposes of filtering impact in this chapter, we argue that a reduction of at least 20% in the summary length (relative to the input text) is appropriate for a summary, which equals a compression ratio of $CR \geq 1.25$. We note that this is not a strict requirement per se and may again depend on domain-specific factors, but has proven to eliminate a large number of subpar samples in analyzed datasets. In all cases that we considered it could always be argued that samples with summaries longer (or equal) than their respective references (i.e., $CR \leq 1.0$) pose an inadequate sample and must therefore be filtered.

Some related work takes a more drastic approach to compression ratio filtering, arguing that extreme content reduction may result in a lossy summary and should therefore also be avoided (Urlana et al., 2022); this simply equals a strict upper limit imposed during the filtering.

Extractiveness Filtering

To train a summarization system that is in fact able to generate *abstractive* summaries, it is required to train with samples that present at least a similar level of abstractiveness, instead of simply regurgitating particular segments from the input text.[31] As a consequence, we may want to filter out examples where the summary (or parts of it) are present in the original reference document. As another language-agnostic check, it is sufficient to verify whether the entirety of the gold summary is present in the input – we refer to this as *"fully extractive"*. As a more flexible measure to also uncover "near fully extractive" examples, we may want to use a slightly more forgiving metric.

[31] Of course, this is different for training an extractive system; in such cases it would be a perfectly fine output to repeat input segments. For the sake of our argument (and this work), we assume that such extractive tasks can be done by using much more efficient algorithms in the first place.

One idea is to compute character-level edit distances, which has worked reasonably well in some of our preliminary analysis. Given that the runtime of efficient algorithms for edit distance scale quasi-quadratically ($O(m \cdot n)$, where m and n are reference and summary texts, respectively), the performance degradation on longer inputs can be significant. Irrespective, it should be respected that an absolute edit distance automatically biases the detection of shorter samples unless specifically using a normalized edit distance.

Finally, we also theorize that ROUGE-L (in itself a problem variant related to edit distance) can be abused to uncover especially extractive samples. By computing the LCS between the reference and summary, extractive summaries will generally have a much higher score than abstractive samples. Compared to the previous two approaches, this method requires an additional tokenizer that again limits the flexibility of said method. Aside from this, computing the LCS suffers from similar scalability issues as the edit distance approach mentioned before due to its algorithmic complexity. As such, unless specified otherwise, we restrict ourselves to the detection of fully extractive samples in our experiments, but encourage others to analyze the filtering quality when expanding to a more complex analysis method.

Duplicate Filtering

One of the – surprisingly – less popular quality checks during dataset creation seems to be checking for duplicates, which is an issue that is also applicable in more general ML settings. However, given that each sample for summarization comes with two separate text snippets (the reference and associated gold summary text, respectively), there are further distinctions between different instances of sample duplication.

Trivial to consider are instances of what we call **exact duplicates**, i.e., samples that have the exact same combination of reference and summary appearing as another tuple in the dataset.

This idea can be further expanded by three more considerations, which we call **partial duplicates**. These are instances where one may find either the reference or summary in other dataset instances. Finally, it could also occur that both summary and reference are duplicated, but across different samples.

To understand why duplicates, including partial ones, can be considered harmful as a training resource, we need only look at the potential effects during training or evaluation. For exact duplicates, no real gain is achieved by including one sample several times in the training data and in the worst case can lead to overfitting of a trained generative model. Worse yet, if we encounter exact duplicates *across different data splits*, this can cause active falsification of evaluation results (commonly referred to as "train-test leakage"). While partial duplicates are less severe, we still encourage removal, as they can cause confusion during the learning process: cases where different input texts should generate the same summary hamper generalization of models, and the reverse case of similar input texts generating different summaries conveys unclear learning signals during training.

While *spotting* duplicates is fairly straightforward, removing duplicate content is often non-trivial, as there are several valid solutions to a de-duplication problem. In an attempt to reduce impact on smaller test and validation sets, we adopt an "additive" de-duplication strategy for the remainder of this work:

Starting with an empty dataset, and iteratively add new samples if (and only if) neither the reference nor the summary have been previously included. Particularly for datasets with multiple splits, we first iterate through the test and validation splits, and only then cover the training section. In this way, the relevant evaluation portions of datasets are less likely to be affected by "de-duplication purges", and instead samples will be taken from the original dataset.

So far, however, we have again only considered examples where input or summary texts are *exactly* equal. Instead, we may also adopt approaches that consider approximate equality between samples; while it is much more expensive to obtain, such information can also yield near duplicates, which potentially affect a larger subset of the corpus. The same ideas of approximate extractiveness filters can be applied to duplicate filtering as well (see previous section). The difference being that now two reference texts or two summaries are compared instead of using the reference/summary pair. Especially when using approximate filters, we want to point out that setting reasonable thresholds is critical for a sensible result. As such, it may be relevant to inspect a series of samples that fall into the borderline region of being a clear mistake or an accidental false-positive.

Neural Scoring Methods

As a side-note, there are also a number of works investigating data pruning strategies on a more general level for LLM data selection (Marion et al., 2023). These are generally similar in spirit to the previously mentioned semantic coherence metric (Bommasani and Cardie, 2020) and perplexity-based scoring methods like BARTScore (Yuan et al., 2021). Marion et al. find that using perplexity of a trained model can be used to estimate the utility of a particular training sample, or, more likely, to filter out samples with particularly low-quality sample alignments. This phenomenon seems to be more effective for bigger LLMs, as they evaluate models of up to 52 billion parameters. We therefore discount a similar approach for testing summarization systems discussed in this chapter, since none of the evaluated systems even exceed the threshold of one knowledgebillion parameters.

Similar ideas of using neural methods to filter datasets exist for text summarization as well, see Guo et al. (2022). The authors use factuality estimation metrics (cf. Section 3.5) to filter out the bottom 25% of training instances while retaining most of the system performance. In both cases, these methods again rely on trained systems or metrics being available for a particular language, and thus do not generalize as well. With more multilingual LLMs becoming available, it might be possible (but quite expensive) to perform data filtering based on neural models alone.

3.4.2 Manual Inspection of Samples

Even with all of the proposed automated measures, nothing can ensure data quality quite as well as the manual inspection of data.[32] All previously discussed measures can point to systemic failures in the data collection process, but may ignore more localized quality issues for particular samples. While a manual analysis step is not feasible at scale, often enough reviewing few samples will already reveal tendencies about the underlying data quality. We generally differentiate between the following strategies to inspect data samples and their respective up- and downsides:

1. **Reviewing samples in order:** A linear sequence of samples may reveal particular issues in sample consistency, which may be linked to the data acquisition processes. We emphasize that "linearity" can follow many particular axes, not just the order in which data is stored. Further possible orderings can be based on available metadata descriptors, such as sortings by timestamps, data source, or length of samples. In-order traversal of samples is the most likely approach to uncover systematic issues, such as incorrect alignment settings that span several samples.

2. **Reviewing random samples:** Another popular approach is to shuffle data and randomly select instances for review. This is fairly easy to implement and does not require iterating over the full dataset or sorting operations. Advantages of random reviews are a more holistic coverage of the data distribution, but requires potentially more manual reviews to find systemic failures.

3. **Outliers and representative samples:** If data statistics are already known (or comparatively easy to obtain), a more targeted approach is to look for distributional outliers. There are again a variety of metrics that can be considered, with the most obvious being text length and compression ratio of individual samples. Manually reviewing outliers can also refine specific requirements of sample properties, e.g., the minimum/maximum length of a summary in relation to the input text. Related are *representative samples*, which constitute instances close to the mean or median of a distribution.

4. **Comparing similar samples:** Specifically for filtering out near-duplicates, it can be helpful to decide on a reviewing strategy that matches two samples that have a high similarity.

[32] This may seem like stating the obvious. Despite this, we want to encourage the reader to think about how many datasets reveal issues that would have definitely been caught, had the authors of the datasets manually inspected some of their samples. As such, we feel it is more than justified to spell out even such basic requirements.

> For use cases like this, simple neural embeddings tend to work well enough to get extremely similar samples grouped together and filter them out manually. Subsequently, this also gives reviewers usually a better understanding of the general diversity of a dataset, by checking the sum number of "similar" pairs exceeding a threshold determined by these manual reviews.

3.4.3 Analysis of German Summarization Systems

Given the presented set of filtering methods, we set out to put current models' capabilities into a sobering context. To this end, we conduct a set of four experiments, specifically for German models and datasets: We start by applying the filters introduced in Section 3.4.1 to existing datasets, noting varying levels of subpar samples. To analyze the changes introduced by our filtering approach on the evaluation splits of certain datasets, we re-compute a set of strong baselines as updated results for datasets with available validation and test splits. Further, given the previously uncovered discrepancies in some datasets, we repeat more comprehensive experiments on the German subset of MLSUM (Scialom et al., 2020) and MassiveSumm (Varab and Schluter, 2021) across the pre- and post-filtered dataset to highlight the effect of filtering on ROUGE scores. We are able to show that this change in data quality also significantly impacts the reproducibility of results. Finally, we provide a small case study in which we examine a subset of generated samples that highlight some of the particular model-centric issues. Importantly, all our findings are conducted on top of existing fine-tuned models. Alternative experiments are possible by training with filtered data sources, which could ultimately yield models that are better able to generalize across datasets.

Filtering Datasets

> **Key Finding 1:** German subsets of two popular multilingual resources (MLSUM and MassiveSumm) have extreme data quality issues, affecting **more than 25% of samples** across all splits.

Table 3.8 presents our findings for filtering the available German summarization datasets; hyperparameters for filters are specified in the table caption. We refrain from imposing any particularly strict filtering metrics, prominently for the length of texts. Most concerning is the fraction of affected samples in MLSUM, given its popularity as a training resource for many public models. While a strong lead bias is to be expected from news articles as a training source, the eventual performance of models trained on the unfiltered dataset is severely impacted; a finding that we confirm in subsequent experiments. Primarily, it indicates that for fully extractive samples, summaries can be generated by directly running an extractive summarization system, and thus obtain similar (or better) quality at a much lower cost. For MassiveSumm, a large fraction of invalid samples can be attributed to duplicate content; manual inspection reveals that there are frequent generic references or summary texts, such as "*Read more after logging in!*". We assume the reason to be a faulty extraction logic.

The remaining inspected datasets were affected at a much lower rate; we see several subsets that have only a handful of faulty instances. Depending on the overall size of the dataset, this implies that evaluation scores will differ less between unfiltered and filtered splits of largely unaffected datasets.

For context, we also evaluate the most popular dataset for summarization (in English), CNN/-Dailymail.[33] Our findings show that around 1.7% of samples are filtered out with the same filtering criteria applied before. Most of the filtered instances are duplicates, which are harmful for training, as they encourage memorization of text sequences, rather than generalizing. We do not explicitly differentiate whether those duplicates constitute train/test set cross-contamination.

Consistent Results and Baseline Runs

> **Key Finding 2:** Existing evaluation scores are hard (if not outright impossible) to reproduce, even with model weights publicly available.
> **Key Finding 3:** Authors frequently fail to put scores into context, not comparing their own results against baseline methods for further scrutiny.

Another worrying trend we observe in the "reproducibility" column of Table 3.1, is the consistent inability to even approximately reproduce self-reported scores for any of the evaluated models. In our reproduction attempts, we employed no particular further filtering, and observe scores that are anywhere from 5 points worse to 3 points better than self-reported scores on the test set. Only a singular result was reproducible within 0.5 ROUGE points of the expected results. In particular, we find that implementation details on filtering steps and other subselection criteria are rarely (if ever) included in the documentation of training procedures. While the usage of so-called "model cards" (Mitchell et al., 2019), i.e., dedicated documentation pages for particular training results, has improved the availability of at least *some form* of documentation, these descriptions are still insufficient to fully reproduce results. As a side note, it should also be mentioned that multiple implementations for the ROUGE evaluation metric exist[34], which may result in scoring differences by utilizing different text processing tools or implementations, see also our discussion in Section 2.4.5. To ensure reproducibility of our own scores, we mention that scores were com-

[33] We use the corpus available at `https://huggingface.co/datasets/cnn_dailymail`, last accessed: 2024-04-04. In particular, the non-anonymized version 3.0.0.

[34] e.g., rouge-score (`https://pypi.org/project/rouge-score/`, last accessed: 2022-10-06) or pyrouge (`https://pypi.org/project/pyrouge/`, last accessed: 2022-10-06), to only name a few.

			Min Length			Min	Fully	Duplicates			
Dataset	**Split**	**Samples**	**Ref**	**Summ**	**Id**	**CR**	**Extr**	**Exact**	**Ref**	**Summ**	**Valid Samples**
MLSUM	Train	220,887	0	0	39	30	**126,204**	31	45	105	94,433 (42.75%)
	Val	11,394	0	0	0	0	**3,285**	1	1	5	8,102 (71.11%)
	Test	10,701	0	0	0	0	**3,306**	1	5	2	7,387 (69.03%)
MassiveSumm	Train	478,143	253	**16,294**	0	**33,959**	0	805	**73,886**	4,882	348,064 (72.79%)
Swisstext	Train	100,000	0	0	0	0	3	0	0	2	99,995 (100.00%)
WikiLing	Train	58,341	11	0	0	**1,435**	0	4	2	52	56,837 (97.42%)
Klexikon	Train	2,346	0	0	0	10	0	0	2	0	2,334 (99.49%)
	Val	273	0	0	0	1	0	0	0	0	272 (99.63%)
	Test	274	0	0	0	1	0	0	0	0	273 (99.64%)
EUR-Lex	Train	1,115	0	0	0	18	0	0	0	0	1,097 (98.39%)
	Val	187	0	0	0	0	0	0	0	0	187 (100.00%)
	Test	188	0	0	0	0	0	0	0	0	188 (100.00%)
LegalSum	Train	79,937	0	2	0	12	326	233	95	**3,106**	76,163 (95.28%)
	Val	9,992	0	0	0	4	32	14	2	157	9,783 (97.91%)
	Test	9,993	0	0	0	7	33	8	1	59	9,885 (98.92%)

Table 3.8: German text summarization datasets in numbers. Given are the original sample count and breakdown of filtered samples by automated assessment (cf., Section 3.4.1) for all provided splits. We set the *Minimum Length* to 20 characters for summaries and 50 for references, except for WikiLingua, which has limits of 8 and 20 characters, respectively, due to a domain-specific differences in writing style. *Id* refers to samples with same reference and summary text, *Min CR* ensures references are at least 25% longer than summaries, and *Fully Extr* identifies consecutive segments that are used as fully extractive summaries. For duplicates, we differentiate between both reference and summary appearing in the corpus (*Exact*), versus partial duplicates where only one of reference (*Ref*) *or* summary (*Summ*) are appearing elsewhere. Numbers in bold highlight issues affecting more than 2% of the data.

puted with help of the `rouge-score` package, version 0.1.2. We further replaced the default stemming algorithm with the German Cistem stemmer (Weissweiler and Fraser, 2017) to provide a reasonable upper-bound of scores and use the provided bootstrap sampler with $n = 2000$.

Aside from the lack of reproducible results, we also noted that only few public models report against a set of (consistent) baselines, with the most commonly compared approach being lead-3. Given that we have also presented a cleaned portion of popular datasets, we strive for a more comprehensive comparison of actual results, and investigate resulting implications that were omitted in the original evaluation settings.

In our particular setup, we compare against three mentioned extractive baselines already introduced in Section 2.4.1 and report scores in Table 3.9. We compare against lead-3, lead-k, and modified LexRank using SentenceTransformers (LexRank-ST), given their flexibility and great scalability for longer inputs.

Dataset	Method	Validation Set R-1	Validation Set R-2	Validation Set R-L	Test Set R-1	Test Set R-2	Test Set R-L
MLSUM	lead-3	19.06	5.58	13.21	18.90	5.47	13.04
	lead-k	14.93	4.12	11.31	15.08	4.17	11.45
	LexRank-ST	15.78	3.36	11.52	16.04	3.30	11.55
Klexikon	lead-3	15.19	3.46	9.10	15.87	3.64	9.35
	lead-k	28.11	5.51	12.43	28.34	5.50	12.50
	LexRank-ST	27.23	4.63	11.48	27.42	4.58	11.55
LegalSum	lead-3	16.72	2.80	10.51	16.74	2.86	10.53
	lead-k	14.34	2.27	8.78	14.36	2.34	8.78
	LexRank-ST	21.54	6.22	12.97	21.35	5.99	12.74
EUR-Lex-Sum	lead-3	3.31	2.25	2.72	3.31	2.19	2.67
	lead-k	41.74	17.77	16.04	39.42	17.08	15.52
	LexRank-ST	39.37	15.13	15.26	38.48	15.18	15.19

Table 3.9: Baseline results for all datasets with available validation and/or test splits. We report ROUGE F1 scores on the filtered datasets.

Depending on the dataset, the choice of a baseline can heavily skew the interpretation compared to neural methods. For example, on the Klexikon dataset, using lead-3 can lead to a roughly 12-13 point drop in ROUGE-1 scores compared to scores by the lead-k or LexRank-ST baseline. On the other hand, for lead-heavy and short texts in MLSUM, lead-3 serves as the best baseline method. Our recommendation is therefore to similarly use multiple (different) baseline approaches, resulting in a more defined context for evaluation based on ROUGE scores. While it may be easier to simply copy results from prior work, we highly recommend the reproduction of these results first, as scores may ultimately vary between different experimental setups.[35]

Effect on Automatic Evaluation Metrics

> **Key Finding 4:** After filtering, scores can drop by more than 20 ROUGE-1 points on the MLSUM test set.

To illustrate the effect of dataset filtering on downstream performance, we further compare results on the two most-affected datasets (MLSUM and MassiveSumm). Without any additional training, we run all available public models on the validation and test portion of MLSUM, for which we also obtain scores on the original unfiltered sets. Our findings can be seen in Table 3.10, where one can observe a performance drop in *every model*, even those that were not originally trained on the MLSUM dataset itself (*t5-base* and our baseline approaches). By far the worst affected are the two baselines constructed from leading sentences, as well as the mT5-small models

[35] For this purpose we have collected a number of baseline evaluation scripts in a code repository, see: `https://github.com/dennlinger/summaries`, last accessed: 2024-04-06

	MLSUM Validation Split						**MLSUM Test Split**					
	Unfiltered			**Filtered**			**Unfiltered**			**Filtered**		
Model	**R-1**	**R-2**	**R-L**	**R-1**	**R-2**	**R-L**	**R-1**	**R-2**	**R-L**	**R-1**	**R-2**	**R-L**
Lead-3	36.22	26.24	31.89	19.06	5.58	13.21	37.15	27.48	32.94	18.90	5.47	13.04
Lead-k	29.25	20.92	26.51	14.93	4.12	11.31	31.35	22.86	28.58	15.08	4.17	11.45
LexRank-ST	18.62	6.46	14.26	15.78	3.36	11.52	18.83	6.45	14.36	16.04	3.30	11.55
mrm8488	**42.77**	31.89	**38.93**	21.63	6.64	16.32	**44.05**	33.44	**40.36**	21.31	6.36	16.09
ml6team	28.17	18.81	26.05	17.08	5.03	14.18	28.51	19.52	26.53	16.56	4.80	13.78
T-Systems	23.74	11.08	20.34	19.87	6.49	16.40	23.67	11.21	20.36	19.20	6.11	15.84
Shahm	42.59	**31.96**	38.70	21.50	6.87	16.15	43.92	**33.62**	40.09	21.20	6.62	15.79
t5-base	27.54	11.31	20.88	**23.31**	**7.19**	**16.99**	27.99	11.65	21.20	**23.40**	**7.20**	**16.91**

Table 3.10: ROUGE F1 scores on the MLSUM validation and test splits, comparing results with and without data filtering. Across all tested models, a stark drop in performance can be observed. We highlight the highest score for each split in bold.

by users *mrm8488* and *Shahm*. These models all achieve unreasonably high ROUGE-2 scores before filtering and see a reduction to about one fifth of the original scores after filtering. Upon inspection, we found that these models were ultimately simply re-generating the first tokens from the input article. These findings are concerning, as they ultimately question the current state-of-the-art on the MLSUM dataset. It further validates the necessity of filtering, given that we can ultimately change the course of evaluation and interpretation of models. For MLSUM, per our results, the *t5-base* model, trained on a related news dataset and utilizing the largest underlying neural model, seems to perform best on filtered datasets while originally lagging behind even a simple lead-3 baseline. This is particularly interesting, because the underlying model checkpoint used is primarily trained on English texts.

We further analyze the impact of filtering on the length distributions of the two heavily affected datasets, MLSUM and MassiveSumm. We observe a more strictly enforced minimum length for both references and summaries in the MLSUM dataset even before filtering. In stark contrast, MassiveSumm is shrunk considerably by the minimum length filter, which in turn shifts the samples towards generally longer reference texts.

Changes in the length distribution, however, do not explain any of the deterioration in raw ROUGE scores; a further indicator that several different evaluation methods need to be combined in order to paint a more complete picture for the realistic performance of models.

Qualitative Analysis of Generated Summaries

Key Finding 5: With the exception of one work (Aksenov et al., 2020), no publicly available system performs experiments beyond simple ROUGE score computation.

Key Finding 6: Despite high reported scores, catastrophic failures can be observed in some systems.
Key Finding 7: All utilized architectures only work with a relatively limited context, proving to be incapable of dealing with long-form summarization.

The first criterion we were looking at when checking for existing systems is the evaluation setting that was used in the respective work. The findings, reported in Table 3.1, point towards a more rigorous evaluation setting for models backed by a scientific publication, which comes as no surprise. However, we also note that these systems are also more likely to withhold their respective models from public access. This ultimately means that those models can only be judged based on the reported evaluation and no further checks can be performed on those models. To aggregate the insights gained across these works, most frequently mentioned is the issue of factual consistency (Venzin et al., 2019; Fecht et al., 2019), which does not bode well for the practical usability of such systems beyond simple settings. Secondly, several works also investigate system outputs' fluency (Fecht et al., 2019; Aksenov et al., 2020), where abstractive models could provide sensible improvements over extractive systems. However, especially for earlier works, consistent generations from language models still prove to be difficult.

To follow our own advice, we manually investigated instances of generated outputs from systems in Table 3.10. In addition to samples from the MLSUM dataset, we further tested with instances from the Klexikon and WikiLingua datasets to check for domain generalization. As others have noted, the factual consistency of abstractive systems is questionable at best, but understated just how badly summaries can deviate from the original. Several times a reversed order of aggressors and victims (respectively, winners or losers in sports game) was generated, and in one particular instance the context was altered from "live-saving" to "drowning (someone)" by the summarization system. This happened on "in-domain samples" from the MLSUM test set.

A similar observation can be made for the syntactic quality of generations, where overfitting of systems becomes particularly apparent during the zero-shot evaluation on other datasets. While it can be expected that the quality of a generated summary may lack in content accuracy or truthfulness, oftentimes no coherent sentence was provided. Less tragic, but difficult for system comparison, is the multitude of parameters for generation functions. While self-reported scores of public models generally rely on greedily decoded summaries, one model frequently started repeating short sequences of about three words indefinitely until the maximum generation length was reached. Importantly, such repetitions are not obvious from looking at a ROUGE-based evaluation of model outputs alone, but could be easily suppressed by enabling n-gram-based filtering during the generation.

We were also able to verify that the highly-scoring models by users *mrm8488* and *Shahm* indeed only copy the leading tokens from the input samples, likely due to training on unfiltered MLSUM splits. This spells further trouble for "state-of-the-art" models, as it requires a deeper examination for determining which summaries are actually better than simple string selection approaches, such as lead-3. We hypothesize that the same concept used in our *extractiveness* filter can also be applied to generated outputs; with a slightly altered similarity scoring mechanism, e.g., the longest common subsequence algorithm, even near duplicates could be detected and flagged for manual review. Most prominently though, due to architectural constraints of the underlying neural models, none of the currently public systems is able to capture an input context beyond 512 subword tokens.

3.4.4 Concluding Thoughts

Studying the current landscape of German abstractive summarization initially paints a grim picture: While the general willingness and ease of sharing systems has greatly increased over the past years, around half of the currently known German summarization systems still remain inaccessible to the public. Of those that *are* available for public scrutiny, a prominent focus on news summarization (with singular documents) is still persisting, similar to related works focusing on English, preventing broader application scenarios of summarization systems. Even worse, the most widely used dataset contains severe flaws in the sample quality, leading to models whose generalization capabilities, even in-domain, are severely hampered by the unfiltered data. This also hints at the general level of care practitioners take with respect to exploratory data analysis, given that several issues can be spotted by simply inspecting just a few samples.

However, there are some silver linings at the horizon. Several of the major data-centric issues can be easily fixed with the introduced quality checks, which can be applied cost-effectively across multiple datasets, as we have demonstrated in this section. Within just two years, we have also seen an unbelievable influx of available summarization datasets for German, importantly extending past the narrow domains into application-specific fields, such as law and medicine, and totaling more than 700,000 samples across publicly available resources. This hopefully paves the way towards a more consistent and generalized approach in German abstractive summarization research; should the efforts of the community keep at the current rate, we will likely see meaningful progress within the next years. The latest trends in the English summarization community also indicate a shift towards greater awareness of long-form summarization (Phang et al., 2023); first long-context multilingual models with open weights are now accessible.[36] Since the time of our original peer-reviewed manuscript, we have also seen the release of commercial models targeting explicit support for other languages, including German.

But, finally, even trained models that apply data filtering steps in their pre-processing pipleine have persisting issues, as qualitative analyses of generations can still reveal catastrophic problems that

[36] `https://huggingface.co/CohereForAI/c4ai-command-r-plus`, last accessed: 2024-04-04

prevent an ethically responsible deployment of the solution in practice. This requirement for a *semantically grounded* analysis of generated summaries finally brings us to the third part of this chapter, which deals with techniques for factual evaluation of generated texts.

3.5 Evaluating the Factuality of Summaries with Semantic Role Labeling

As we mention in Section 2.3, one of the remaining issues that prevents productive deployments of neural text summarization systems is the low correlation of system outputs with human preferences. Among those, *factuality*, i.e., the agreement of facts in the generated summaries with those present in the input text, is not part of the general training objectives of models, which frequently leads to hallucinated facts that are detrimental to perceived system performance (Fabbri et al., 2021; Ter Hoeve et al., 2022). To accommodate a more linguistically grounded representation of summary content units (cf. the work by Nenkova and Passonneau (2004)), we introduce a representation of so-called *fact tuples*. Each fact tuple corresponds to a semantic relation in either the original input text or the summary, and aligning pairs of fact tuples between the two text segments allows for the computation of a factual consistency score. We begin by motivating the need for such a metric, and compare our presented method, called **SRLScore**, to a range of existing factuality evaluation metrics. Empirically, we are able to demonstrate a performance on-par with state-of-the-art metrics.

3.5.1 Motivation

Prior work has introduced metrics for automated testing of factuality in generated text (Goodrich et al., 2019; Kryscinski et al., 2020; Yuan et al., 2021), which allows for a more nuanced verification of model capabilities without further human interaction required. In particular, one of the first relevant works by Goodrich et al. (2019) introduces the idea of representing text as a series of "fact tuples", in their case as `(subject, predicate, object)` triplets. Their method exhibits some assumptions about the underlying data which hampers correlation with human ratings. For example, subject or object may vary for the same sentence meaning expressed using different syntactic structures, e.g., active and passive forms. Semantic Role Labeling (SRL), however, allows for a syntactically independent meaning representation. Our metric, **SRLScore**, improves factuality evaluation, building on fact tuples similar to Goodrich et al. It distinguishes itself in several ways from existing approaches:

1. To account for a more nuanced representation, we employ SRL to produce abstract representations of sentences that are *independent of their syntactic formulations*.

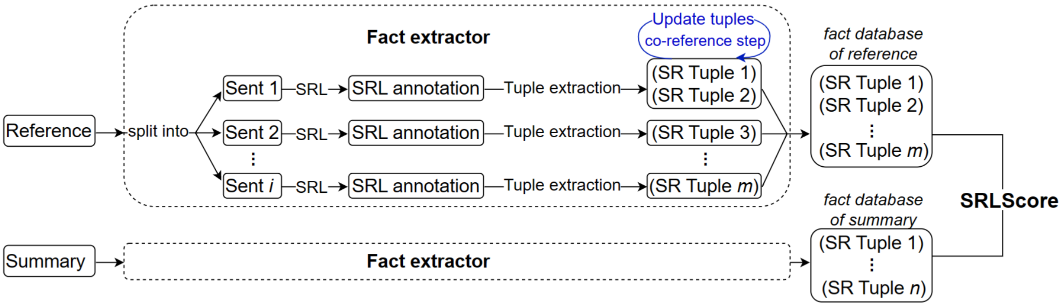


Figure 3.6: Visual explanation of **SRLScore**. An input text and its associated summary are transformed into a series of fact tuples (*SR Tuple*) through extraction from SRL (and optional co-reference) annotations. The final factuality score is computed based on the similarity of the summary facts with fact tuples generated from the input text.

2. Fact tuples in **SRLScore** are generated on the *input text* instead of gold summaries; as a consequence, our method is reference-free, and may be applied for evaluation irrespective of the availability of labeled datasets.
3. We introduce a novel weighting scheme for fact tuple comparison, where adjustable weights allow for user optimization.
4. Finally, we experiment with extensions along different parts of the pipeline, including an optional co-reference resolution step and alternative similarity scoring functions.

Notably, **SRLScore** entirely relies on publicly available software components and may be used without any further domain adaption required. While our experiments are performed on English, we argue that the transfer of our approach to other languages is possible given only the existence of a language-specific tokenizer and a sufficiently good SRL tagger. Furthermore, **SRLScore** offers the additional benefit of being an *interpretable* metric, due to its composition on top of fact tuples. In comparison, metrics used for factuality evaluation that are based on the intermediate presentations of language models, e.g., generation perplexity (Zhang et al., 2020b; Thompson and Post, 2020; Yuan et al., 2021), cannot present insightful reasons *why* a particular score was achieved. There is also growing evidence that generation-centric evaluation strategies have shortcomings: Fabbri et al. (2021) and Liu et al. (2023b) show that generative models (used as discriminators) prefer outputs from systems trained on similar architectures. Kamoi et al. (2023) demonstrate similar problematic issues for QA-based metrics. We empirically show that the correlation of **SRLScore** with human ratings is on par with existing methods, and perform several ablations to study the impact of various algorithmic choices within our pipeline.

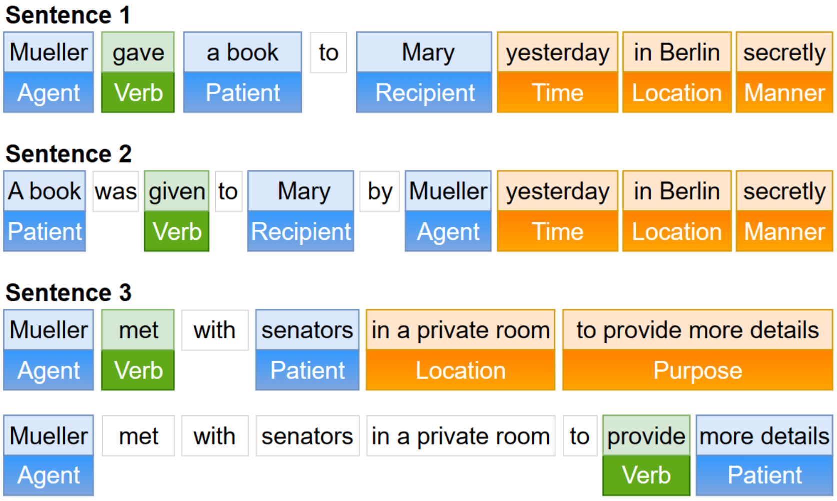


Figure 3.7: Examples of semantic role label annotations. Labels may remain consistent across different syntactic forms (Sentence 1 & 2). A single sentence can also include several relations at the same time (Sentence 3).

3.5.2 SRLScore

Our factual consistency metric, called **SRLScore**, is implemented as a two-stage process: first, extracting fact tuples using Semantic Role Labeling (SRL) on both the source texts and the summary texts, and then determining a factuality score based on tuple comparison. The measure outputs human-interpretable scores between 0 and 1, where a higher score indicates greater factual consistency of a summary text. In this section, we detail the algorithmic choices and present an adaptive weighting scheme for computing the final factuality scores.

Generating Fact Tuples with Semantic Role Labeling

As Figure 3.6 shows, we operate on the sentence level, primarily because existing SRL tools work well on this level of granularity (Shi and Lin, 2019; Xu et al., 2021b). The goal of our fact extractor is to produce *a fact database* comprised of semantic role tuples for each input text.

The primary task of SRL is to find all role-bearing constituents in a sentence and label them with their respective roles (Màrquez et al., 2008). Typical semantic roles include *agent*, *patient/theme*, *recipient*, *goal*, *instrument*, *manner*, *time*, *location* and so on. From the many semantic labels available, we include seven roles based on availability in tagging schemes to construct a fact tuple: *agent*, *negation*, *relation*, *patient*, *recipient*, *time*, and *location*. We further note that not every sentence needs to contain *all* of these roles; absent labels are represented by *None* in this work. Importantly, roles reveal the semantic relations between a predicate (verb) and its arguments, which implies that one can generate several fact tuples from a single sentence, depending on the number of verbs in it. To illustrate an exemplary fact tuple, the extracted semantic tuple from sentence 1 in Figure 3.7 is `(Mueller, None, gave, a book, Mary, yesterday, in Berlin)`.

Scoring Texts by Comparing Fact Tuples

Once fact tuples for both the reference and summary texts are generated, the second step in our pipeline is to compute a factual accuracy score. We implement a dynamic weighting system, which crucially improves over a naive comparison, as we empirically show in Section 3.5.3. Furthermore, we describe the drop-in replacements for exact matching during similarity computation.

Scoring Algorithm Given a reference text D and summary text s, let F_D and F_s be *fact databases*, representing the semantic information contained in D and s, respectively. Individual fact tuples are represented as an ordered list of fact arguments, e.g.,

$$f = (agent, negation, relation, patient, recipient, time, location) \in F \tag{3.1}$$

Particular arguments in a fact tuple are referred to by their index position, meaning $agent := f^0$, $relations := f^1$, and so on. We further assume that there exists a scoring function that expresses the *factual support of two fact tuples f_s, given a reference tuple f_D*, denoted as $S(f_s|f_D)$. To obtain a factuality score, we attempt to extract a best match $\hat{f_D} \in F_D$ for each summary fact $f_s \in F_s$ where $\hat{f_D}$ maximizes the support score $S(f_s|\hat{f_D})$. Importantly, we differ from, e.g., Goodrich et al. (2019), by considering the entirety of F_D, instead of reduced subsets that match both the agent and relation of the fact tuple f_s.[37] The factual accuracy is then the average across all maximized tuple scores in F_s. With that, **SRLScore** is defined as:

$$\textbf{SRLScore}(D, s) := \frac{1}{|F_s|} \sum_{f_s \in F_s} \max_{f_D \in F_D} Support(f_s|f_D) \tag{3.2}$$

The final part of this scoring system is the computation of factual support $Support(f_s|f_D)$. Tuples are scored by comparing the corresponding attributes of each tuple, formally:

$$Support(f_s|f_D) := \sum_i \mathbb{1}_{f_s^i \neq None} \cdot sim(f_s^i, f_D^i) \cdot w_i, \tag{3.3}$$

where the summation over i addresses all attributes of the fact tuples, $\mathbb{1}_{f_s^i \neq None}$ represents an indicator function considering only non-empty arguments f_s^i (zero otherwise), and w_i assigns static weights to arguments in position i. Generally, it should be assumed that the weights allow for a

[37] The original reasoning for this restriction is likely a greedy approximation of an "optimal" match to reduce computational load, as well as reducing the number of false positives, which are more likely to occur in the original fact triplets due to fewer arguments.

maximum factuality score of 1, i.e., $\sum_i w_i = 1$. Finally, $sim(f_s^i, f_D^i)$ is the pairwise argument similarity of f_s^i and f_D^i. We consider different similarity metrics, as described in the following sections.

Dynamic Weighting System The generic weighting in Equation (3.3) does not necessarily apply to the particular case of evaluating factual consistency in summarization, since a summary is still factually correct even if it leaves out particular aspects (e.g., dropping the date of an event), which were present in the reference text. With static weights, however, absent arguments are still contributing to the scoring of the tuple f_s, which means that leaving arguments out might potentially be considered as a penalization of factuality. To address this issue, we introduce a weight re-normalization factor, W_{norm}, that distributes the static weights w_i across only those attributes that are present in the current summary fact. In particular, this also increases penalties for actual mistakes over simple fact omission. The weight normalization is defined as follows:

$$W_{norm} := \frac{1}{\sum\limits_i \mathbb{1}_{f_s^i \neq None} \cdot w_i} \tag{3.4}$$

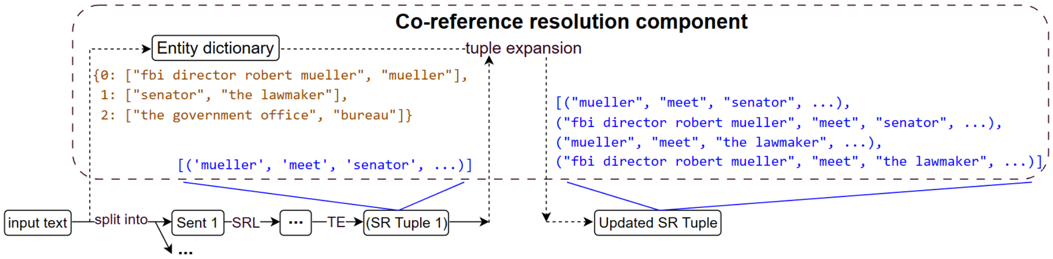
With re-normalization enabled, we replace the existing computation of $Support(f_s|f_D)$ by the product $W_{norm} \cdot Support(f_s|f_D)$.

String Similarity Methods We experiment with different methods to calculate the pairwise similarity $sim(f_s^i, f_D^i)$: exact matching (in line with prior work), but also approximate matching functions, such as word vector similarity[38] and ROUGE-1 precision (Lin, 2004). Computation of similarity with vectors and ROUGE each have their own respective strengths. Word vectors offer the highest flexibility in terms of recognizing partial argument similarity, enabling semantic comparison instead of purely syntactic equivalence. ROUGE-1 similarity does not offer the same level of flexibility in terms of matching, but faster computation, while still recognizing partial matches.

Improved Surface Form Invariance with Co-reference Resolution

In light of the fact that sentence-level SRL extraction misses co-references of the same entity across sentences, we integrate an optional component that takes co-reference resolution into account during the tuple generation. Concretely, we employ an off-the-shelf co-reference resolution tool (Lee et al., 2017) to identify and store all reference clusters in an external *entity dictionary*.

[38] We use spaCy's vector similarity, see `https://SpaCy.io/usage/linguistic-features#vectors-similarity`, last accessed: 2023-03-06



```
{0: ["fbi director robert mueller", "mueller"],
 1: ["senator", "the lawmaker"],
 2: ["the government office", "bureau"]}
```

```
[('mueller', 'meet', 'senator', ...)]
```

```
[("mueller", "meet", "senator", ...),
("fbi director robert mueller", "meet", "senator", ...),
("mueller", "meet", "the lawmaker", ...),
("fbi director robert mueller", "meet", "the lawmaker", ...)]
```

Figure 3.8: Example of tuple expansion through co-reference resolution. In addition to the original SR tuple, we add tuples with all possible permutations of the surface forms of mentioned entities.

There, all linguistic expressions that refer to the same entity will be grouped together, which allows for later disambiguation. As shown in Figure 3.8, if an extracted semantic role tuple contains co-references, a single fact tuple will be *expanded* into multiple tuples, representing the Cartesian product over all synonymous entity surface forms.
The key idea here is to enable a better matching of potential facts across references and summaries, effectively increasing the recall of matches. The disadvantage is that this directly affects the runtime of our method by a strong factor, since the additional tuples in F_s and F_D will undoubtedly increase the number of comparisons.

3.5.3 Experimental Results

We empirically demonstrate the performance of our method through a number of experiments on two popular datasets for factual consistency evaluation, which are covered in this section. We further share implementation details and the choices for extracting SRL tuples and extracting co-reference clusters. In addition to the experimental analysis, we also study the behavior of **SRLScore** through a number of ablation experiments, and a brief error analysis.

Evaluation Datasets

QAGS (Wang et al., 2020) The dataset comprises of 235 instances collected from the test split of CNN/DailyMail (Nallapati et al., 2016), where each instance contains a source article and a model-generated summary using the bottom-up approach by Gehrmann et al. (2018). A secondary set contains 239 further instances from the test split of XSUM (Narayan et al., 2018), with generated summaries sampled from BART (Lewis et al., 2020a).

SummEval (Fabbri et al., 2021) As an alternative study on the CNN/DailyMail dataset, SummEval includes synthetic summaries from 16 abstractive and extractive models of 100 randomly selected articles from the test split of CNN/DailyMail. Unlike QAGS, which collected

Metrics	**QAGS-CNNDM**		**QAGS-XSUM**		**SummEval**		**Avg.**
	ρ	r_s	ρ	r_s	ρ	r_s	ρ
ROUGE-1 (F1)	0.34	0.32	−0.01	−0.05	0.13	0.14	0.15
BLEU	0.13	0.33	0.08	0.03	0.09	0.14	0.10
METEOR	0.33	0.36	0.06	0.01	0.12	0.14	0.17
Cohere command-xl	0.19	0.17	–	–	–	–	0.19
BARTScore	0.65	0.57	0.00	0.02	0.27	0.26	0.31
BARTScore$_{\text{cnn}}$	**0.73**	**0.68**	0.19	0.18	0.35	0.32	0.42
BARTScore$_{\text{cnn+para}}$	0.69	0.62	0.07	0.07	0.42	**0.37**	0.39
CoCo$_{\text{span}}$	0.64	0.55	0.22	0.20	0.40	0.35	0.42
CoCo$_{\text{sent}}$	0.68	0.59	0.16	0.14	0.39	0.35	0.41
ClozE-R$_{\text{core\_web\_trf}}$*	0.66	–	0.32	–	0.47	–	**0.48**
ClozE-R$_{\text{confidence}}$*	0.65	–	0.29	–	**0.48**	–	0.47
SRLScore$_{\text{base}}$	0.67	0.59	0.20	0.18	0.43	0.33	0.43
SRLScore$_{\text{coref}}$	0.65	0.58	0.27	0.26	0.43	0.32	0.45
SRLScore$_{\text{coref-optimized}}$	-	–	**0.33**	**0.33**	–	–	–

Table 3.11: Pearson (ρ) and Spearman (r_s) correlation of metrics with human ratings on the evaluated datasets. Bold scores indicate highest absolute values. For **SRLScore** variants, we report highest scores across all similarity functions. No significant differences were found between the correlation scores of factuality-specific metrics. *: results were taken from the respective paper, as there is no existing code to reproduce their results as of now.

annotations from MTurk,[39] each SummEval sample was evaluated by 5 crowd-sourced annotators and 3 experts. For each summary, the judges were asked to evaluate the coherence, consistency, fluency and relevance. For our evaluation, we use the expert ratings with regard to factual consistency as the gold score, per the recommendations of the original SummEval authors.

Evaluation Metrics and Significance

In line with prior work, we evaluate metrics by computing Pearson correlation (denoted as ρ) and Spearman correlation (denoted as r_s). Given the limited size of evaluation datasets, we further test results for significance using permutation tests (Riezler and Maxwell, 2005; Deutsch et al., 2021b). In all tables, † denotes a significance level of 0.05 ($p < 0.05$) and ‡ represents a significance level of 0.01 ($p < 0.01$). When testing significance against several systems, we further apply Bonferroni correction of significance levels (Dunn, 1961).

Implementation

We use AllenNLP (Gardner et al., 2018), specifically version 2.1.0, to extract semantic role labels. AllenNLP implements a BERT-based SRL tagger (Shi and Lin, 2019), with some modifications. The output of AllenNLP uses PropBank convention (Bonial et al., 2012), which lists for each

[39] https://www.mturk.com/, last accessed: 2023-03-06

verb its permitted role labels using numbered arguments (*ARG0, ARG1, ...*) instead of names, due to the difficulty of providing a small, predefined list of semantic roles that is sufficient for all verbs. However, numbered arguments are meant to have a verb-specific meaning (Yi et al., 2007). In other words, the mapping between numbered arguments and semantic roles is not always consistent. In our implementation, we extract sentence spans with label ARG0 as *agent* and spans with label ARG1 as *patient*. The extraction of *time* and *location* also does not pose any difficulties, because ARGM-TMP and ARGM-LOC are both given as modifiers that remain relatively stable across predicates (Jurafsky and Martin, 2009). However, as shown in Table 3.12, there is no one-to-one relationship between numbered arguments and the *recipient* role. For the sake of simplicity, we extracted elements with label ARG2 as *recipient*, because the probability that ARG2 correlates to *recipient* is the highest among all other possible roles (Yi et al., 2007). For co-reference, we simply use the model provided by AllenNLP (Lee et al., 2017), which matches the output format of the SRL tagger.

All experiments were carried out on a system with an Intel Xeon Silver 4210 CPU, two TITAN RTX GPUs (24 GB GPU VRAM each) and 64 GB of main memory. We run inference for the SRL model and co-reference component on separate GPUs. We use the official scripts provided by the authors of BARTScore[40] and CoCo[41]. Unfortunately, no public implementation exists at the time of writing for the work of Li et al. (2022), which prevents significance testing against ClozE models. For the work by (Goodrich et al., 2019), we similarly found no publicly available implementation; however, we note their wikipedia-based training data for generating fact extractors is available online[42]. When attempting to reproduce the scores of Xie et al. (2021), based on their own implementation, we encountered wildly differing result scores. On two subsets, we obtain drastically better results compared to their reported Pearson correlation of 0.58 (our own results indicating a correlation score of 0.68), while other values dropped even further (e.g., on QAGS-XSUM, we see a reduction of scores from 0.24 to 0.16 in terms of Pearson correlation). For the sake of reproducibility, we have included the exact commands that were used to run the CoCo models in our repository,[43] and report the scores obtained during our reproducibility experiments. On the other hand, all of our reproduced scores for BARTScore (Yuan et al., 2021) match the available self-reported results by the authors. For significance testing, we use our own implementation of a permutation-based significance test. We fix the initial `NumPy` random seed to 256, and compute results over 10,000 iterations for each test.

[40] `https://github.com/neulab/BARTScore`, last accessed: 2023-02-01
[41] `https://github.com/xieyxclack/factual_coco`, last accessed: 2023-03-16
[42] `https://github.com/google-research-datasets/wikifact`, last accessed: 2023-05-17
[43] `https://github.com/heyjing/SRLScore`, last accessed: 2024-04-06

ARG0	agent	**ARG1**	patient
ARG2	instrument, recipient, attribute	**ARG3**	starting point, recipient, attribute
ARG4	ending point	**ARGM**	modifier

Table 3.12: Mappings between numbered arguments in PropBank and relevant semantic roles (Bonial et al., 2012). Particularly the mapping of argument 2 makes simplifying assumptions about different verb forms.

We further briefly experimented with the idea of Large Language Models as evaluators. Given the computational cost of inference, we only ran one model by Cohere[44] on the CNN/DailyMail subset of the QAGS evaluation, which comprises relatively few instances, but of longer summaries. For the explicit LLM evaluation setup, we prompt a model to individually rate whether a summary sentence is factually consistent with the reference text, forcing a binary output label, where 0/False means no factual consistency, and 1/True indicates factually consistent sentences. This procedure is the same as the original human labeling process for the QAGS dataset, and we finalize instance-level predictions by averaging over the ratings for all sentences within a summary. Our initial findings were disappointingly low correlation scores (0.19 ρ and 0.17 r_s), letting us to believe that few-shot attempts with LLMs are not (yet) comparable to hand-crafted metrics. It is very likely that more recent models would perform better.

System Variants

We compare with a number of generic summarization evaluation metrics, including BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), and METEOR (Banerjee and Lavie, 2005). Besides, we also consider several metrics specifically developed for factuality estimation, which have reported prior state-of-the-art correlation. Wherever possible, we reproduce scores with the official scripts provided by authors. Comparison is done with three variants of BARTScore (Yuan et al., 2021), two variants of CoCo (Xie et al., 2021), and two variants of ClozE (Li et al., 2022). We chose each variant such that the highest self-reported scores on all evaluated datasets are considered.

For our own method, SRLScore$_{\text{base}}$ represents a default setting, assigning equal weights $w_i = \frac{1}{7}$ to all attributes (*agent, negation, relation, patient, recipient, time, location*); the respective similarity function (exact match, spaCy vector, or ROUGE similarity) is chosen to maximize dataset-specific performance (see results of Table 3.13). SRLScore$_{\text{coref}}$ uses the same weights, with coreference enabled. We further provide model ablations of our method: SRLScore$_{\text{openie}}$ and SRLScore$_{\text{goodrich}}$ are approximations of the tuple variant introduced by Goodrich et al. (2019), where fact tuples are reduced to `(agent, relation, patient)` (with equal weights $w_i = \frac{1}{3}$). We note that this is not a true equivalent although "[i]n most English sentences the subject is

[44] To be precise, we used the `command-xlarge-nightly` model, accessed on the 23rd March, 2023.

the agent" (Bates and Macwhinney, 1982); in reality, a broader variety of roles in the subject position may be encountered. The same applies for our mapping between *object* and the *patient* role. However, this way, we can compare scoring methods independent of upstream labeling tools (in our case, the SRL tagger). The difference of SRLScore$_{\text{openie}}$ and SRLScore$_{\text{goodrich}}$ lies in the implemented scoring function, where the OpenIE variant employs our own scoring algorithm and SRLScore$_{\text{goodrich}}$ uses the preliminary filtering step defined in (Goodrich et al., 2019). We do not apply a co-reference system in either one of the two ablation settings. Finally, SRLScore$_{\text{coref-optimized}}$ illustrates the possibility of adapting our method to a particular dataset. For this variant, we optimize available hyperparameters (weights, scoring function, co-reference) in order to obtain the highest possible scores.

Main Results

The central evaluation results with recommended default settings are shown in Table 3.11. In almost all cases, specialized factuality metrics show higher correlation than generic summarization evaluation metrics (ROUGE-1, BLEU and METEOR). Notably, despite the high increase in absolute scores, we do not always detect a significant level of improvement between factuality-specific metrics and generic metrics, particularly on QAGS-XSUM; we will discuss further implications of this in more detail later. When testing our own method, SRLScore$_{\text{base}}$, against generic metrics, we find strongly significant improvements only for Pearson correlation of QAGS-CNN/DM and SummEval, as well as Spearman correlation on SummEval ($p < 0.01$, with Bonferroni correction).

It should be further noted that BARTScore$_{\text{cnn}}$ and CoCo results use BART models (Lewis et al., 2020a) that were fine-tuned on the CNN/DailyMail corpus (respectively a variant fine-tuned on XSUM for CoCo on QAGS-XSUM); this may shift the results in favor of these methods for the particular dataset. In comparison, **SRLScore** does not make such assumptions, which may indicate a potentially stronger generalization to unseen datasets.

Results in Table 3.11 also show that there are no significant differences between any of the factuality-specific metrics (**SRLScore**, BARTScore, and CoCo), particularly after applying Bonferroni correction for the comparison against several methods. These insights open up discussions about the current claims of "state-of-the-art" performance, which may not be easily distinguished on the current evaluation datasets. We admit that there is likely no trivial solution to this (besides further annotations), as the main problem seems to stems from the high variance on small sample sizes. We also note that Bonferroni corrections are particularly "strict" in their approach to significance, and possibly discards actually significant results. However, a deeper statistical analysis is beyond the scope of this current chapter.

Metrics		**QAGS-CNNDM**		**QAGS-XSUM**		**SummEval**	
		ρ	r_s	ρ	r_s	ρ	r_s
SRLScore$_{\text{openie}}$	Exact	0.59	0.51	0.09	0.09	0.34	0.28
	ROUGE	0.62	0.56	0.07	0.07	0.41	0.32
	SpaCy	0.59	0.53	0.13	0.10	0.37	0.32
SRLScore$_{\text{base}}$	Exact	0.61	0.54	0.14	0.15	0.37†	0.31‡
	ROUGE	**0.67**	**0.59**	0.15†	0.13	**0.43**†	0.33
	SpaCy	0.63	0.55	**0.20**	**0.18**	0.40†	**0.34**†

Table 3.13: Comparison of **SRLScore** with a simplified triplet representation (SRLScore$_{\text{openie}}$). Extending the fact tuples improves correlation with human ratings across all similarity functions. Significance markers indicate improvements over the same similarity function of the $_{\text{openie}}$ variant.

Ablation Study

Given the limited (statistical) expressiveness of the general evaluation setting, we perform a series of ablation studies on **SRLScore**specifically, to support the individual algorithmic choices made in our method. We study how the inclusion of further attributes affects performance, which empirical impact our newly proposed weighting scheme has, how the choice of similarity computation affects performance (especially on abstractive summarization data), and finally briefly evaluate the wall-clock performance of our current implementation against BARTScore.

Extending Tuple Attributes We investigate the assumption that semantic representations of sentences are usually far more complicated than the simplistic view of (*agent*, *relation*, *patient*) triplets, and the fact that errors may involve further roles. To this end, we compared SRLScore$_{\text{openie}}$, using a triplet representation, against SRLScore$_{\text{base}}$ which includes the full seven roles present in our fact tuples. Results in Table 3.13 confirm that extending tuples to cover more semantic roles is effective across datasets and metrics; SRLScore$_{\text{base}}$ scores strictly better than SRLScore$_{\text{openie}}$, with significant improvements primarily on SummEval (the largest considered dataset).

Performance of Similarity Functions Also seen in Table 3.13 is the difference in scores across various similarity functions. **SRLScore** achieves generally higher correlation when using vector (spaCy) or ROUGE similarity over exact matching, although not to a significant degree. These observations can be attributed to the hypothesis that abstractive entity references will not be detected by exact matching. Also note that results on QAGS-XSUM are particularly affected by this, which shows higher levels of abstraction than CNN/DM-derived resources (Wang et al., 2020; Pagnoni et al., 2021). This is also visible for the SRLScore$_{\text{coref}}$ variant, as seen in Table 3.11, which can further improve the matching of re-formulations.

Weight Setting	**QAGS-CNNDM**		**QAGS-XSUM**		**SummEval**	
	ρ	r_s	ρ	r_s	ρ	r_s
Static weights	0.59	0.49	0.09	0.09	0.38	0.28
Dynamic weights	**0.67**	**0.59**	**0.20**	**0.18**	**0.43**	**0.33**

Table 3.14: Correlation scores of $\text{SRLScore}_{\text{base}}$ with and without weight re-normalization enabled.

Scoring Method	**QAGS-CNNDM**		**QAGS-XSUM**		**SummEval**	
	ρ	r_s	ρ	r_s	ρ	r_s
$\text{SRLScore}_{\text{goodrich}}$	0.45	0.38	0.05	0.07	0.29	0.24
$\text{SRLScore}_{\text{openie}}$	**0.62**†	**0.56**†	**0.13**	**0.10**	**0.41**‡	**0.32**†

Table 3.15: Results of the ablation experiment comparing the scoring method by Goodrich et al. (2019) with our proposed scheme, based on triplet representations.

Dynamic Weight Re-Normalization We next analyze the contribution of our dynamic weighting scheme through removing the weight re-normalization W_{norm} and instead defaulting to a static weighting on $\text{SRLScore}_{\text{base}}$. Results in Table 3.14 again indicate that our choice of re-distributing weights dynamically to only the present roles is very effective. However, we still do not reach a statistically significant level of improvement in this scenario.

Ablation of Goodrich Scoring Method We finally examine the performance of our scoring system against the partial matching approach of Goodrich et al. (2019). For fairness, we compare results on the reduced triplet sets. $\text{SRLScore}_{\text{openie}}$ uses the presented weighting function, $\text{SRLScore}_{\text{goodrich}}$ implements an equivalent scoring to Goodrich et al. Results in Table 3.15 show that the presented scoring algorithm performs better than the scores determined by Goodrich's approach on different datasets, in most instances to a significant degree.

Performance of Co-reference Resolution System Results in Table 3.11 reveal that the co-reference system is not always improving scores, particularly on the CNN/DailyMail-derived datasets. However, the use of co-reference resolution will significantly increase the processing time, as demonstrated in Table 3.17. This is expected given that there are now more fact tuples due to the *tuple expansion*; since the presented scoring method requires the comparison of each fact tuple in the summary against *all* reference tuples. We further compare the runtime against BARTScore, which only requires a single forward-pass through a neural net and can be batched easily, resulting in a 10x speed-up. In comparison **SRLScore** requires building and comparison of fact tuples, which is the main contributor for slower inference times.

	Sample Text	Extracted Fact Tuples	Human	SRLScore
Reference	Former England fast bowler Chris Tremlett has announced his retirement ...	(Former England fast bowler chris tremlett, announce, his retirement, ...)	0	0.87
Summary	Former England seamer James Tremlett has announced his retirement ...	(Former England seamer james tremlett, announce, his retirement, ...)		
Reference	The head of Japanese advertising group Dentsu is to step down following the suicide of an employee ...	(The head of japanese advertising group dentsu, step, ..., following the suicide of an employee, ...)	1	0.10
Summary	The chief executive of Japanese advertising firm Dentsu will resign after a worker killed herself ...	(The chief executive of japanese advertising firm dentsu, resign, ..., after a worker killed herself, ...), (a worker, killed, herself, ...)		

Table 3.16: Examples from the QAGS-XSUM dataset where the majority vote of human factuality ratings differs drastically from **SRLScore**'s predicted score. Text segments highlighted in green mark the position of relevant facts, whereas red text indicates a factual discrepancy between reference and summary segments.

SRLScore		BARTScore		
base	coref	base	cnn	cnn+para
2.35	19.32	0.22	0.23	0.23

Table 3.17: Average processing time (in seconds) per instance in QAGS-CNN/DM. **SRLScore** uses ROUGE similarity. BARTScore is run with a batch size of 4. Note that the CNN/DailyMail dataset is geared towards shorter summaries of less than 5 sentences in most cases.

Error Analysis

To better understand the limitations of our presented methods, we examine a number of instances manually, particularly those where there are large differences between model-generated scores and human annotations on QAGS-XSUM. Table 3.16 shows two instances, where **SRLScore** respectively predicts a much higher and lower factuality score than human annotators. Notably, human raters tend to drastically reduce factuality scores in the presence of even a single mistake (what we refer to as *"strike-out scoring"*). In comparison, **SRLScore** and other factuality metrics tend to be more heavily influenced by the correctness of the *majority* of attributes, which can be seen as a *"bottom-up scoring"* (scores are built up from an initial factuality of zero instead of deducing from an initial score of one). On the other hand, highly abstractive samples, which retain factuality according to human raters, may pose a challenge for tuple-based **SRLScore**. In the second example of Table 3.16, synonymous expressions like *step down* instead of *resign* cause low predicted similarity. This implies that our method may struggle more in settings with highly abstractive summaries, although we argue that the entities central to a factual analysis will stay close in their referenced surface form.

Concluding Thoughts

During our work on factuality detection, we have found further worrying trends in the current state of research: while there is a plethora of existing work dealing with the analysis (and estimation) of factuality, most related works (including our own presented method) rely on the correlation with human ratings on English-only and comparatively small gold datasets. While we are able to demonstrate that **SRLScore** performs on par with existing approaches, we find that, due to the small sample sizes of evaluation datasets, there are no significant differences between any of the considered "state-of-the-art" factuality estimation metrics. Despite this nuisance, we highlight that our approach strikes with its relative simplicity and interpretability due to the intermediate representation of "fact tuples", which makes it possible for human annotators to review how or why system decisions were made. Furthermore, we have demonstrated the suitability of our approach over more naive tuple-based scoring methods through a series of ablation experiments, which also show the adaptability of our method to particular unseen settings by simply adjusting a series of parameters. Given these ablations, we still want to argue that extended tuple representations may offer a more insightful way of also annotating future reference datasets; the abstract representation, independent of syntactic choices in the reference (or summary) offer an improved flexibility that has to be otherwise off-loaded onto the annotator themselves.

Aside from this, there are also challenges concerning the effective deployment of factuality metrics, including **SRLScore**. The current implementation still suffers from impractically long run times for longer input texts. Notably, however, both the tuple generation and comparison stages can be parallelized to improve the future compute efficiency. Secondly, we have seen a general trend that factuality estimation metrics are scoring differently from human annotators, who are putting heavy emphasis on a *completely* factual summary instead. We suspect that adopting a similar *strike-out scoring* for estimation may better correlate with human ratings, although it will require sufficiently accurate taggers to ensure correct recognition of all entities, but have not made any further progress towards formulations that are robust enough with the currently available tagging tools. We also want to point out that the underlying summarization datasets that were used to compare human ratings on are known for their own set of limitations, particularly being fairly extractive in nature (Zhang et al., 2018a). This plays well with **SRLScore**'s estimation of matching between individual tuples extracted from single sentences; on the other hand, if summary texts contain facts derived from multiple source sentences (or undergo otherwise complex structural changes), fact tuples may be insufficient in their current form.

3.6 Conclusion

This chapter dealt with several problems of modern text summarization systems and provided at least partial remediation of some of these aspects. We particularly put a focus on three different areas, which neatly map onto the complex process of designing, implementing, and evaluating a summarization system.

First, we proposed a new dataset for complex summarization tasks, heavily focusing on a long-form resource with differing domain focus from most existing datasets; it further is available in 24 languages, expanding possible research beyond just English. The creation also showcases some of the particular challenges surrounding the dataset curation process: even with existing resources on the web, it has to be first analyzed (and fully understood) how these resources are originally created, whether they are suitable for the training of later summarization systems, and what potential biases can be found in the original source data. Analyzing the existing methods on our dataset, we further find that available abstractive summarization systems are unable to handle the full context of certain samples, leading to a strictly limited applicability of existing approaches.

This problem is only exacerbated for non-English languages, which leads us to the subsequent second point of analyzing existing work for other languages. At the example of German, we examine both existing resources and models with respect to their overall quality and usability. In order to ease the analysis process, we proposed several heuristics to filter out low-quality samples from existing datasets, and find that the most prominent resource, almost exclusively used to train public German summarization models, is heavily flawed in its quality. We further study the impact of dataset filtering on downstream evaluation with automated metrics, and find that these can be easily skewed by low-quality samples, too.

However, even with automated filtering steps in place, we cannot ascertain that all generated outputs of abstractive models are correct. In order to do so, we presented an automated metric for the evaluation of generated text with respect to factuality in the final part of this chapter. Our metric, unlike existing work, does *not* fully rely on neural architectures for score computation, which has been shown to introduce several biases (Fabbri et al., 2021; Liu et al., 2023b). Instead, it utilizes interpretable intermediate representations of "fact tuples" and shows promising correlations with human evaluation scores.

With this deeper analysis of limitations in existing systems, we now also feel equipped to tackle the task of proposing a more flexible and generically useful summarization system that implicitly addresses some of the mentioned shortcomings in the subsequent chapter.

4 A Formal Framework for Aspect-based Summarization

"The belief that one's own view of reality is the only reality is the most dangerous of all delusions."

Paul Watzlawick

The previous chapter has dealt with a deeper and more collected analysis of *existing summarization systems* and discussed a number of consistently re-surfacing problematic issues. Particularly with respect to the user needs, a lack of consideration for subjectivity of summaries has been apparent, and current systems primarily produce texts that are intended to do well on evaluation data rather than addressing particular user needs. To this end, we continue by proposing a new conceptual view of summarization as a series of what we refer to as *aspect-based* transformations of the input data, ultimately leading to a more flexible system allowing for user-specific customization. To be precise, we introduce the distinction of two general notions in these aspects: *ex-ante* and *ex-post*, which will be individually explained in greater detail as well. We argue that previously presented work on controllable text generation systems can be re-stated as either one of the classes that we present in this work (i.e., adjustments towards a particular *need/aspect*). Finally, we point out that these aspects also neatly map onto the practically used architecture of existing hybrid summarization systems, introduced in Section 2.2.3. The modular architecture of hybrid summarizers further improves the extensibility of our proposed framework, as later additions or modifications of an existing pipeline are easy to realize, and require little interference with other components or additional re-training.

The remainder of this chapter is structured as follows: we begin with the introduction of necessary background on controllable summarization systems in Section 4.1. From the various approaches to modeling user preferences, it becomes apparent that many of the existing solutions share a general *structural* similarity, but fail to define a common view of aspects. We attribute a particular aspect's influence on either the filtering (*ex-ante*) or generation (*ex-post*) stage of a hybrid model, and discuss these two (mostly disjoint) aspect categories in Section 4.2 and Section 4.3. However,

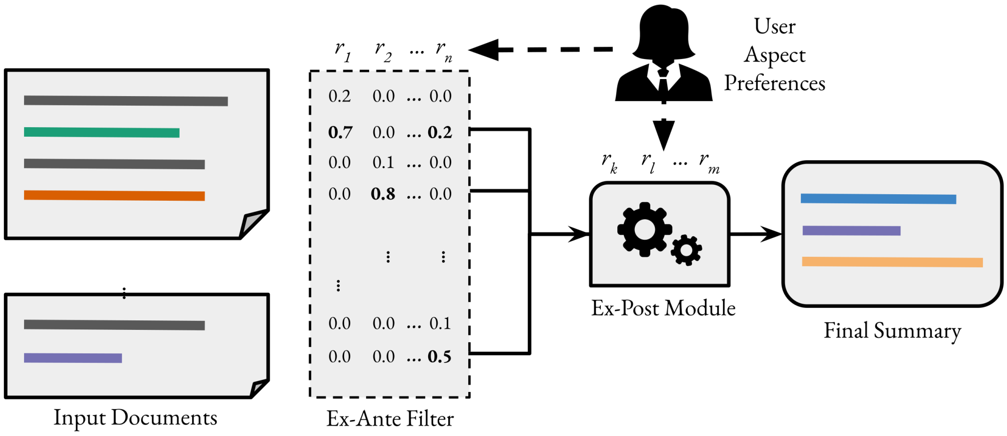


Figure 4.1: Schematic overview of our proposed two-stage aspect-based architecture. Individual segments from input documents are first judged on their relevance with respect to several *ex-ante* filters. A second stage, called the *ex-post* module, acts as a processor to re-write filtered segments into a coherent output, with further aspects imposed on the generation. Users can steer the summary by expressing explicit aspect preferences.

we demonstrate that this theoretical notion can be practically combined into a singular system, and elaborate on the challenges for a generalized application in Section 4.4.

The final part of this chapter, Section 4.5, details a first implementation and considerations for its application, although a more thorough refinement of modeling aspects can be found in subsequent chapters as well. Here, we illustrate that aspects can be realized by both neural and classical algorithms, which is especially useful to address the need of providing solutions that work well on non-English inputs. We briefly conclude with a synopsis of our insights in Section 4.6.

4.1 Background and Related Work

The reason of why existing models are particularly challenged by the incorporation of user preferences is quite easy to demonstrate. Existing work largely builds on a similar formalization of the "summarization task" as we introduced it earlier in Section 2.1. Notably, the mention of *users* is completely absent from this section, and the task is entirely defined as a pre-conditioned text generation problem. Therefore, in order to explicitly incorporate user needs into our systems, we partially re-arrange the formalization to introduce another input parameter, the *user*, in Section 4.1.1. While no two users may be the same, we can further break down the existing approaches into several *aspects*, which serves as a proxy for the implicit user preference model.

As the second part, in Section 4.1.2, we discuss prior work in the area of aspect-based summarization, and how they deal with the varying preferences (or subjectivity) in summary generation.

Particularly, we find existing architectures that introduce weak forms of what we consider as aspects. However, the existing works mostly focus on the inclusion of parameters in end-to-end settings, and consider narrow datasets for evaluation scenarios. Similar limitations exist specifically for multi-document settings, where we find that previous systems consider only active user inputs as aspects. In fact, a central contribution in later sections will be the unification of several distinct aspect-based framework in one central model. We demonstrate that existing work is only a particular variant of our more generalized notion.

4.1.1 A Formal Notion of Aspects

One of the key limitations of prior works in the area of aspect-based summarization is that they generally assume a limited and rather concrete notion of what constitutes an "aspect". For example, Fan et al. (2018) introduce four different aspects for their controllable text generation model: variable length, entity-centricity, source styling, and what they call "remainder summarization", i.e., conducting a summary from only parts of a document. Another example, He et al. (2022) similarly consider length and entity-focused summaries, but further introduce domain-specific notions, such as distinction of contributions in scientific papers, as well as a Question Answering frame for summarization. As a complete opposite to these relatively constrained aspects, Kulkarni et al. (2021) only consider user-specified queries as guidance signals. Even when comparing only these works, it becomes clear that there is no fixed understanding of "aspects", and it oftentimes becomes impossible to tell whether the notion of aspects with the same name (e.g., "length") are understood in the same way across different systems and approaches.

As mentioned, we will use the notation introduced in Section 2.1 to build a more precise theoretical framework for aspect-based summarization, in an attempt to bridge the gap between various different schools of thought. The central aim is to establish a holistic and precise language when talking about "aspects", and allow the representation of any arbitrary (potentially not previously considered) user need within this framework.

Consider, in general terms, a user U, within the possible space of all users $\mathcal{U}$. Without further specifications, a user can have one or more preferences for a summary, which alter the previously given definition of a constrained text-generation problem, to now instead consider the explicit (or implicit) user needs as an optimization objective. We can then rewrite the definition of the constrained summarization problem stated in Equation (2.12), incorporating further inputs of the "user" U, such as

$$summ(\mathcal{D}, U) := \underset{s \in \mathcal{D}}{\operatorname{argmax}} \operatorname{pref}(s, U). \tag{4.1}$$

Here, $\text{pref}(s, U)$ is a function that scores a particular summary s with respect to a user's preference patterns. Compared to the generalized setting in Equation (2.12), we note two key insights: Firstly, we can argue that existing (generic) summarization systems use a proxy over the space of all users $\mathcal{U}$, as obtaining explicit feedback on summarization systems is relatively expensive.[1] Training a supervised generic summarization system can thus be re-written as

$$summ(\mathcal{D}, U) := \underset{s \in \mathcal{D}}{\text{argmax}} \frac{1}{|\mathcal{U}|} \cdot \sum_{U \in \mathcal{U}} \text{pref}(s, U), \tag{4.2}$$

i.e., maximizing the expected return over the aggregate of all users' preferences. Given the unavailability of explicit user feedback, instead simple assumptions about these signals are inferred from, e.g., gold summaries, and a soft alignment is learned based on these outputs.

Secondly, the originally present optimization towards shorter outputs (the constraint) is a simple heuristic to gauge user preferences, assuming that a majority of users would only consider a summarization system preferable over the original input text, if there is some level of compression.

With this in mind, we are able to formulate a theorem aggregating notions of user preferences into a formal view of *aspects*:

Corollary 1 (Aspects in the context of summarization)**.** *Given a user-constrained view of summarization, as defined in Equation (4.1), we refer to an aspect* $\mathbf{a}$ *as any property that directly affects a user's perceived preference rating of the final summary* $\mathbf{s}$ *with respect to a particular dimension.*

Instead of simply optimizing towards the *expected* proxy reward of an average over the population $\mathcal{U}$, we are now able to distinguish between different preference dimensions (the aspects). Thus, an aspect-based scenario no longer necessitates the joint optimization of *all* aspects in a singular end-to-end framework, but rather allows for a modular decomposition of the objectives into different sub-problems. Particularly for the later formalization of disjoint aspect dimensions (e.g., "entity-centricity" and "query focus"), it can be theorized that a decomposed creation of a summary for the "entity-centricity", combined with a summary for the "query focus", can sufficiently cover the user preference along both dimensions. Notably, not all aspects may be considered orthogonal and thus, additive in the nature of optimizing towards them. We will reserve this as a problem fpr further discussion later on.

Such a broad distinction allows us to further categorize and describe two distinct notions of aspects: *ex-ante* (Section 4.2) and *ex-post* (Section 4.3). The former can be considered any modification on the input text that is only concerned with modifications of the input text, plus some

[1] We note that there exist approaches that consider such explicit preferences to train systems, and achieve state-of-the-art performance. See, for example, the early work of Christiano et al. (2017) within the field of RLHF (reinforcement learning from human feedback), as well as summarization-specific applications of Stiennon et al. (2020).

	Summarization Guidance Signals			
Related Work	**Guidance Token**	**Rel. Triples**	**Highlights**	**Summaries**
Kikuchi et al. (2016)	✓(length)	✗	✗	✗
Cao et al. (2018)	✗	✗	✗	✓(retrieval)
Li et al. (2018)	✓(keywords)	✗	✗	✗
Liu et al. (2018a)	✗	✗	✓	✗
Liu et al. (2018b)	✓(length)	✗	✗	✗
Fan et al. (2018)	✓(length, entity, style)	✗	✗	✗
Jin et al. (2020)	✗	✓	✗	✗
Saito et al. (2020)	✓(keywords)	✗	✓	✗
Dou et al. (2021)	✓(keywords)	✓	✓	✓(retrieval)
Zhu et al. (2021a)	✗	✓	✗	✗

Table 4.1: A taxonomy of supported guidance signals in *neural* guided summarization models, taken from Dou et al. (2021). It distinguishes between different forms of guidance signals. *Individual tokens* may be used to steer signal on specific aspects, such as length. For incorporation of specific relationships, *knowledge triplets* from graph databases are used in some instances. *Highlight sentences* may be obtained in a first-stage retrieval system and may guide summary generation. Passing *entire summaries* from a series of retrieved documents, or even using them for templating purposes, is also possible.

optional user parameter. One example would be the entity-centricity as an *ex-ante* aspect. In contrast, *ex-post* aspects require "world knowledge" and generally only affect the *generation section* of a summarization system. Here, we could consider the source styling attribute by Fan et al. (2018), which depends on *knowledge* of particular text styles, and goes beyond the current input text.

4.1.2 Related Work

As illustrated before, we are not the first to consider a user-centric view of summarization, but rather attempt to unify and categorize the previously given – and wildly varying – aspects mentioned. Particularly in this section, we start by addressing the three major themes present in previous chapter as well: data, models, and evaluation. As a central basis for learning user preferences, we highlight the various datasets that may be used for learning user-centric generation patterns, as well as the underlying resources from which they were created. Regarding the modeling of aspects, we can distinguish between a number of high-level approaches. Some works attempt to introduce controllability as a parameter in neural models, whereas others resort to a framing as a conditional selection problem (e.g., controllable extractive summarization). Finally, regarding the evaluation setting, we still find a large similarity in the approach to generic single-document summarization settings.

Resources for Aspect-focused Generation

We find only a few dedicated resources for user-centric summarization purposes. The first prominent pillars of user-centric corpora are the DUC 2003 and DUC 2004 tasks. DUC 2003 introduced focused summarization tasks, distinguished by event, viewpoint, or question guidance signals (Over et al., 2007). DUC 2004, Task 5 similarly dealt with "short summaries focused by questions". For this task, annotators answered questions across a total of 50 distinct document clusters, such as "Who is X?", with an average of 10 documents per cluster.[2] Notably, the summaries are generally expected to be rather uniform in length (around 100 words), with low variance across samples.

Later editions of the Text Analysis Conference (TAC) continued similar tracks between 2008 and 2012.[3], There are several renditions of an "opinion summarization track", as well as the update summarization task, vaguely reminiscent of the remainder summary (re-)introduced by Fan et al. (2018). The former task setting is particularly relevant for various e-commerce settings, such as filtering reviews and providing users with short, aspect-focused texts summarizing the opinion of a larger set of previous customers (Dang and Owczarzak, 2008). The data is similar in structure to the previous DUC resources, and is largely centered around NIST's Text Retrieval Conference (TREC) annotations, which also sponsors the TAC shared tasks. In 2010, the organizers of TAC introduced a guided summarization task,[4] which is probably closest to an unconstrained aspect setting, such as it is discussed in our work. The provided samples include a list of key questions or aspects (e.g., keywords), which should be present in the generated summary. Despite the more general setting, the constraints on generation length (≈100 words) and the size of document collections remained the same. Subsequent analysis found that aspect-focused systems rarely outperformed the much simpler baseline (Steinberger et al., 2010).

New datasets have only been proposed more recently. Krishna and Srinivasan (2018) use a topically guided system to generate synthetic training data, which other systems have previously used for model training. They bootstrap the generation of multi-aspect summaries by fixing topics based on an initial list and learning word-frequency-based counts for topics in existing news article documents (Hermann et al., 2015). Similar synthetic approaches to data generation have later been generalized to an unsupervised setting by Coavoux et al. (2019), who use continuous sentence representations as a means to cluster documents into topically consistent aspect-clusters.

As an attempt to re-distribute the focus in resource creation on semi-automated resources (instead of fully relying on automated alignments), Kulkarni et al. (2021) present a query-focused summarization dataset, which matches natural questions from Google's real-world query corpus

[2] `https://duc.nist.gov/duc2004/`, last accessed: 2023-05-03.

[3] See, e.g., `https://tac.nist.gov/2008/summarization/index.html`, last accessed: 2023-05-03.

[4] `https://tac.nist.gov/2010/Summarization/`, last accessed: 2023-05-03.

with web content from the C4 dataset (Raffel et al., 2020). The resource covers around 8,100 examples, spanning a total of 52,700 reference documents from C4.

The largest aspect-focused resource is presented by Hayashi et al. (2021), who use section information within Wikipedia as a weak guidance signal for focused summaries. Spanning a total of 20 distinct domains, they select "aspects" as the most commonly present section titles within subcategory, with a total of 10 aspects per individual domain. The total number of documents exceeds 320,000 training instances, making this by far the largest created resource. However, due to the redundancy in using relevant paragraphs as silver labels, the authors substitute a concatenation of cited content as the input to a summary setting; this drastically limits the achievable performance, due to the unrelated nature of other articles. Their approach is in large parts based on a similar strategy employed by Liu et al. (2018a), who generate a generic summarization corpus on Wikipedia.

Notably, all the previously mentioned resources are again solely focusing on the English language; as we have discussed this problem previously in Section 3.1.1, this severely limits the applicability of models. To our knowledge, besides the work by P.V.S and Meyer (2017), no resource exists in a non-English language. The mentioned work introduces a German resource to study multi-document summarization settings with a secondary guidance focus, which contains topic clusters similar to the DUC/TAC-style datasets.

Aspect-based Summarization Models

Work on modeling user focus in summarization systems is not new. Already before the 2000s, López et al. (1999) introduce the idea in a user-focused retrieval model. They use summaries, consisting of lists of keywords, to get better precision-focused retrieval items, and allow the modification of these lists by users. Aside from the submissions to relevant DUC and TAC tasks, Vanderwende et al. (2007) introduce a topically weighted focus on top of SumBasic Nenkova and Vanderwende (2005), which allows for further modification of content selection. They also already discuss the inclusion of other aspects, similar to our division into ex-ante (selection) and ex-post (modification) approaches. In particular, they argue that a coupled sentence simplification step may help with intra-sentence summarization. A further nice overview of early methods is given by Díaz and Gervás (2007). They refer to the problem as "user-focused summaries", and include keywords, section or category information, as well as "feedback terms", specified by users. First approaches to aspect-guided summarization with neural models can be found in Kikuchi et al. (2016), who introduce singular constraints (in this case, length of an output summary) on the generation process. Approaches focusing on the stylistic guidance of outputs have also been investigated around the same time, however, with a fairly unusual choice of architecture for language generation. Hu et al. (2017) steer the generation of particular sentiments in output sentences by building a model in the style of variational autoencoders (VAEs) and generative adversarial networks (GANs).

Fan et al. (2018) introduce the first neural approach focusing on a variety of styles, and are probably the most similar of the earlier works to the holistic representation of arbitrary aspects as our work. As previously mentioned, they encode length, remainder summarization, stylistic transfer based on the source, as well as entity-centric generation for styling output summaries. However, their evaluation still focuses primarily on generic summarization datasets.

Concurrently, the previously mentioned work by Krishna and Srinivasan (2018) also introduce a similar multi-focused aspect model, which they dub a "topic-aware pointer-generator network". While their model outperforms other methods on the evaluated datasets, the similarities in which the dataset was constructed to the ultimate evaluation benchmark likely gives them an unfair advantage in this setting.

Frermann and Klementiev (2019) are more forward in their discussion of appropriate evaluation setups. They address the lack of suitable training data by synthesizing datasets based on the work by Krishna and Srinivasan (2018), which further modifies the generation of individual aspect-summary pairings. In particular, they manually evaluate the classification accuracy of individual aspect mappings for a subset of their test collection in addition to evaluating existing summarization qualities. Both Frermann and Klementiev (2019) and Krishna and Srinivasan (2018) use similar architectures internally, building on the (at the time) common architecture of LSTM encoder-decoder models with attention (Bahdanau et al., 2015).

Coavoux et al. (2019) present a first unsupervised neural approach, similar in mind to the previous one. While it focuses on product reviews, which we do not consider explicitly in our work, it presents interesting sampling strategies to describe aspect clusters (based on sentence representations). This is particularly relevant for multi-document settings, where topical overlap is to be expected, or otherwise particularly long input sequences.

One of the first transformer-based approaches to guided text generation is probably the CTRL model family (Keskar et al., 2019), which uses simple one-word (or few-word) prompts to induce stylistic properties on the output text. The inclusion of more complex prompts, e.g., the URL/-domain of a referenced article, exhibits similar behavior to what can be observed by recent large-scale attempts, like ChatGPT.[5] They also investigate the generation behavior under prompt mixing, i.e., the combination of multiple different prompts. As a relevant subsequent work, CTRL-Sum (He et al., 2022) applies this paradigm to the particular task family of summarization, and includes prompts in various forms for output styling. CTRLSum supports keywords-based control tokens, for which the authors present an elegant formalization. Modeling user preferences as a keyword in the conditional generation problem $p(y|x, z)$, where x, y is the pair of input/out-

[5] https://openai.com/blog/chatgpt, last accessed: 2023-05-15.

put sequences, and z the user keywords. This is fairly similar to our own formalization, which in theory allows for a generic aspect-controllability, as well as a dedicated focus on user needs in the learning process. The extraction of keywords at training time is automated, and based on the popular ROUGE-2 maximization algorithm to greedily select matching sentences. They then again sub-select max-spans with exact matches in the reference text to align keyword prompts with "expected summaries". At test time, a BERT-based sequence tagger is trained for the purpose of extracting relevant keywords, if none are specified by the user.

Our main criticism of their work is the fact that they seem to model aspects well that are relevant for the styling of output text (i.e., what we introduce as ex-post aspects). In comparison, modeling length requirements or other selection-focused aspects relating to the *input* is seemingly done in a very naive fashion, given their unwillingness to extend the end-to-end neural approach. This especially limits the applicability to longer input sequences, as well as the addition of further aspects without complete re-training. Furthermore, their evaluation is again limited to generic summarization datasets.[6]

Similar to the previous model, Dou et al. (2021) introduce generalized aspect-modeling architecture in the form of a parallel (weight-shared) encoder stack, which encodes arbitrary sequences into a guidance signal. These can include entire sentences (highlights), or user-provided/automatically extracted keywords (primarily for the inference stage).

The only work considering an explicit relevance representation with respect to multiple aspects are Wang et al. (2023), who learn an attention-like vector representation over the sentence-wise relevance during training. Instead of using it as a pre-selector, however, they limit themselves to using relevance vectors as a weak guidance singla during the generation step. They use the WikiAsp dataset by Hayashi et al. (2021), but extend it by a small evaluation study with graduate students on three different analysis dimensions. Departing from a direct training of neural models, Chan et al. (2021) utilize constrained Markov Decision Processes (CMDP) as a means of controlling the output generation for text summarization. Interestingly, this theoretical modeling allows them to enable training on policy reward functions, such as optimizing for BERTScore (Zhang et al., 2020b) instead of token-level losses. For a meta-analysis of current trends in the field, we further refer the reader to Urlana et al. (2023b).

Evaluation Strategies for Aspect-based Summarization

Yang et al. (2023) are evaluating ChatGPT for aspect-based summarization. They sample from QMSum, SQuALITY, and two other datasets, compute ROUGE scores, as well as the metrics by

[6] We want to point out to the reader that He et al. (2022) also evaluate their model on the *arXiv dataset*. While one could assume that this would constitute significantly longer inputs than other summarization datasets, the authors perform a number of manual filtering steps, effectively reducing the "reference text" to only include parts of a paper's introduction section.

Grusky et al. (2018). They do compare to task-specific fine-tuning models, which are generally on par with the ChatGPT solution, and significantly better in some instances. Hayashi et al. (2021) consider aspect-discoverability as a separate evaluation axis, thereby reducing the complexity of the analysis to a particular sub-problem. Otherwise, we are not aware of setups that allow for the simultaneous evaluation of multiple aspect dimensions in parallel, much less without meaningful insights for users on a larger test bench.

4.2 Ex-Ante Aspects

Existing approaches to aspect-focused summarization share a number of common pitfalls: They predominantly focus in English as their language of choice, with no support for multilingual tasks, and further struggle with long inputs that may be more realistic to generic application scenarios. However, they key limitation of a fair number of models is their cap on the input length at some arbitrary value, generally assumed to be 512 subword tokens. We already previously discussed this limit in Section 3.1.2 and want to reiterate that this is insufficient for a wide range of domain-specific applications. To address both of these problems, we suggest the "separation of concerns" between the *pre-selection stage* and the *actual controlled text generation* in an aspect-based summarization setting. In fact, this neatly maps onto the existing architecture of hybrid summarization systems (Liu et al., 2018a). As mentioned, with respect to recent advancements for versatile summarization systems, the generic two-stage (hybrid) architecture is becoming the de-facto choice for most related works on long-form text summarization. Its particular design is highly similar to architectures commonly found in other sub-fields of the Natural Language Processing community, such as Question Answering (Chen et al., 2017) or more Information Retrieval-related work on two-stage ranking (Nogueira and Cho, 2019).

Broadly speaking, these systems combine a cheap and relatively effective first-stage module as a preliminary filtering approach and relay the smaller intermediate result set to a more expensive (but also more accurate) system that re-checks or refines the content further, according to the expected output task. Frequently, the first module is an established heuristic such as term frequency – inverse document frequency (TF-IDF) (Jones, 2004) or BM25 (Robertson and Jones, 1976), followed by a larger neural model for the second stage. While the general pipelines might seem similar, the exact task during the second stage varies across different sub-fields: in the case of Question Answering, the target is the selection of a (multi-)word span within a pre-filtered sentence/paragraph, a re-shuffling of top-k results in Ranking, or, as is the case in summarization systems, the re-writing of filtered content into a coherent segment.

We briefly discuss a formal view of the ex-ante stage in Section 4.2.1, before manifesting a number of exemplary dimensions in Section 4.2.2. Importantly, in scenarios with multiple aspects, it may be necessary to combine the representations of several different ex-ante dimensions, which we briefly touch upon in Section 4.2.3.

4.2.1 Formalization

With our more general definition of aspects in mind, we can begin to differentiate between nuanced interpretations, specifically in the context of hybrid summarization architectures. This section deals with aspects that require the incorporation of *ex-ante* information for an intermediate summary representation. One may consider multiple simultaneous preferences modeled via separate aspects. Instead of jointly modeling aspects in an end-to-end neural model (Dou et al., 2021; He et al., 2022), we separate aspect-specific relevance by different ranking models. This allows the later addition of further aspects without the explicit need for re-training systems, as well as the recycling of components across models. The combination of multiple parallel aspects is discussed in Section 4.2.3 Generally speaking, ex-ante aspects are modifications on the underlying reference text, such that a *re-weighting of importance* occurs at the level of individual segments. They can be formalized as components reducing the content within a document collection $\mathcal{D}$. In a hybrid summarization setting, we may refer to the ex-ante modules as a *selector* or *filter*, based on their primary function.

Given this focus on an extractive setting, a formal definition is pretty straightforward from the representation of extractive summarizers in Section 2.1. As a formal view, ex-ante filters can be expressed as a function over the input document collection $\mathcal{D}$ and the user preferences of users in $\mathcal{U}$, mapping to a relevance score, limited to the range of $[0, 1]$. This leads to the formalization as

$$f^{\text{ante}} : \mathcal{D} \times \mathcal{U} \rightarrow [0, 1]. \tag{4.3}$$

Unlike previously given definitions of generic summaries as in Equation (2.5), this instead considers an ex-ante filter as a *ranking approach*, assigning relevance scores $r_i^{\text{asp}} \in \mathbb{R}$ to each existing segment $d_i \in \mathcal{S}$.

Given that no explicit "filtering" has occurred at this point, we may naively consider the intermediate representation $\mathcal{S}^{\text{asp}}$, based on a singular aspect, as a ranked list based on the descending relevance scores, or formally

$$\mathcal{S}^{\text{asp}} := [t_1, ..., t_n], \;\; r_i^{\text{asp}} \geq r_{i+1}^{\text{asp}}, \;\; \forall 1 \leq i \leq n, \tag{4.4}$$

where $n = |\mathcal{S}|$ constitutes the number of all available segments. We may further consider a strictly limited reduction of the intermediate segments, e.g., by simply cutting off the list past a certain number n or threshold value $t \in [0, 1]$. However it may be implemented, from here on we assume that a "subjectively relevant" summary can be generated on the filtered set, as defined in the hybrid model of Section 2.2.3.

The formulation of ex-ante stages has several inherent benefits that may not be immediately obvious to the reader: ex-ante filters differ from existing end-to-end approaches in that they do not have to consider the entire context of the document collection $\mathcal{D}$ at once. While the maximum length usually drastically limits the applicability of systems to long-form user scenarios, having a re-ranking function operating on individual segments is flexible in its usage and can easily incorporate much longer contexts, as long as individual segments are relatively self-contained and are consistent in their maximum individual lengths. In fact, it also makes them naively compatible with existing end-to-end aspect-based (generative) solutions. Given the purely extractive nature of the ex-ante stage, we can consider the resulting intermediate representation as a pre-processed input segment to a secondary stage, which then re-writes text into a coherent output summary.

As a secondary advantage, we argue that *learning* the explicit modeling of simple ex-ante filtering steps, such as the query relevance, adds additional complexity into the training process, which can now be separated out into a dedicated relevance module; this offers benefits as it can (a) re-use existing components trained for this particular tasks from other projects and (b) quickly be exchanged for newly trained systems without replacing other components in the pipeline. These final considerations are particularly interesting for the development of multilingual systems (or generally, systems working on non-English data). While there is certainly little relevant data available for training an end-to-end aspect-based summarizer in non-English languages, one can most certainly find query relevance modules that work perfectly fine in those languages. Having said that, we will now move on to exemplify some of the particular ex-ante aspects that were either previously discussed in the literature, or are immediate variants that come to mind.

4.2.2 Ex-Ante Aspect Dimensions

To understand in more detail what particular aspect dimensions may be represented during the ex-ante stage already, we enumerate a number of different example dimensions. Particularly relevant is the consideration of *independence* between different aspects, which is assumed at this stage. Straightforward translations of filtering steps may include: generic query relevance, entity-centric focus, temporal filters, or simple deduplication filtering. We further discuss how individual filters may be realized in practice, which can range from simple heuristics to complex retrieval models.

Generic Query Relevance

One of the most prevalent ex-ante realizations across existing literature is the incorporation of query relevance filters. In practice, they can take many different forms and shapes, depending on the way in which the query signal is created. This includes explicitly provided user queries (Conrad et al., 2009; Kulkarni et al., 2021)), but also generic keyword-based "search terms" (Steinberger et al., 2010; Dou et al., 2021; Vig et al., 2022).

To rank passages with respect to arbitrary query terms, we may consider existing unsupervised solutions from the IR community, such as the aforementioned TF-IDF or BM25 relevance functions. We detail the respective implementations when discussing our prototype system in Section 4.5. Both variants enjoy wide popularity given their relative simplicity (especially compared to neural models), language transferability (they only require a tokenization module in the target language) and ability to generalize somewhat well to different domains (given their long track record of application in the IR domain). It is not entirely decided whether one metric is preferable over the other.[7]

Of course, more complex query relevance models may be utilized, too. Depending on the size of the document collection, however, the performance considerations for a "real-time" setting may be too restrictive to employ more complicated methods. As such, nearest-neighbor search over embedded segment representations are one possibility (Reimers and Gurevych, 2020), but given the asymmetric nature of query strings and segments (i.e., queries being generally much shorter than individual segments), specific architectures, such as Dense Passage Retrieval (DPR) (Karpukhin et al., 2020) may be a preferable choice.[8]

Furthermore, we may consider topically-aware aspects as a variant of query-relevance aspects (Krishna and Srinivasan, 2018). By replacing the generic search setting with a similarity computation of topic vectors of individual segments (and a query relating to individual topics), it is possible to abuse a generic implementation for a slightly more specific setting.

Entities

A prominent filtering variant in existing work is also the focused generation of particularly entity-centric summaries (Dang, 2006; Over et al., 2007; Fan et al., 2018; Maddela et al., 2022). In such instances, a generic query relevance model may not be sufficient, as it cannot distinguish between the more stringent definition of an entity, usually limiting the consideration to any noun phrase describing a (physically present) *thing* or *person*. As such, most of the content within a section can be easily excluded by not referencing the entity in question, which is different from a soft similarity function described in the previous section.

[7] Chen et al. (2017) report that BM25 performs (slightly?) worse compared to TF-IDF, but do not quantify the difference. IR literature generally values BM25 over TF-IDF as a baseline for ranking problems, but acknowledges the additional complexity of choosing appropriate hyperparameters. We leave it up to the reader to decide on the preferred ranker choice.

[8] Practically speaking, neural representation also offer key advantages over classical ranking methods in terms of normalization; a fact we will detail further in Section 4.5.

In fact, the simplest solution for entity-centric ex-ante filters is a boolean mask indicating exactly this presence (or absence) of the entity being mentioned in a segment d_i; realizations of such a filtering step may utilize existing domain-independent tagging methods, such as they are available in libraries like spaCy (Montani et al., 2023). The required existence of such (usually language-specific) taggers slightly limits the transferability of entity-centric filters. Again, these are still more likely to be available in a particular language compared to an entity-centric end-to-end aspect summarization datasets, which reinforces the preference of such independent filtering approaches.

More interesting is the discussion to what extent entities *need* to be pre-specified by a user. Theoretically, given a user-specified entity to focus on, it may be entirely possible to achieve a simple entity-filtering module by checking for exact matches of the entity in each segment d_i.[9]

On the other hand, a user may simply state their preference as “summarize each entity mentioned in the text”; in such cases, the extent of present entities is not known a priory, and in fact requires a more complex self-filtering. In this process, it has to be first determined what entities are occurring across the document collection, and then separate filters have to be instantiated for each of these entities.

Maddela et al. (2022) discuss considerations for the curation of entity salience, but also restrict themselves to the simpler sub-task of generating summaries for pre-specified entities. In fact, their evaluation serves as a great example of supporting simpler methods for ex-ante aspect-summarization, as their simple baseline of simply returning the top three sentences per entity (what they refer to as lead-3_{ent}) outperforms *all other approaches*.

Temporality

Due to the fact that prior work primarily focused on narrow document collections, often to the point of simply considering singular input documents, this particular aspect has been neglected so far. While we will spend more time considering the modeling of temporal aspects in Chapter 6, we briefly introduce a generic temporality filtering for this context as well. Generally, it should be assumed that there is temporal (meta-)information available for documents in the collection. Most commonly, this is the document creation time (DCT), or otherwise a publication date of first appearance. For the purpose of the current section, we will not further discuss *temporal mentions* (i.e., explicit references to points in time *within* the document’s content).

In particular, we can imagine different scenarios in which this is particularly appealing: primarily for social media-like summarization settings, or similar areas with a large number of documents in $\mathcal{D}$, it may be helpful to extract only those texts which fall within a particular time range, e.g., only articles from within the last two weeks should be considered. On the other hand, we may

[9] Note that this does not work for co-reference mentions of entities. However, even with dedicated entity taggers, this is usually not available, or otherwise requires additional computational overhead, see Section 3.5.

also consider a related topic, commonly referred to as timeline summarization (Campos et al., 2014; Steen and Markert, 2019; Hausner et al., 2020a,b). In such instances, temporal relevance can simply be used to create a recency-biased ordering, or otherwise serves as a guidance signal for later (temporally ordered) summarization generation.[10]

Temporal aspects also offer the benefits of being cheap to compare, structured (temporal markers can be compared without explicit training being required), and language-independent. Of course, they can only be considered in the described scenarios with enough documents in a collection, as otherwise the filtering on DCT stamps becomes fairly redundant as a re-ranking step.

Further Filters

We previously mentioned the problem of *content duplication* in multi-document settings. To address this issue, it is also possible to design a deduplication filter as an ex-ante aspect. As such, "relevance" (or the re-ranked order) may first include only the respective choices of deduplicated text segments, and only later on repeat the duplicate content (with much lower relevance scores being assigned to duplicate content, respectively). While we are not consciously aware of any previous system optimizing towards the explicit filtering of content repetition, the argument can be made that it is usually implied by evaluating systems based on their summary-level coherence. As discussed in Section 2.3.1, this generally also includes the explicit repetition of content. Simultaneously, the filtration of duplicates can be impactful on the ex-post stage, as the overall length of considered content may be drastically reduced in such a fashion.

Finally, we also briefly mention the "remainder summarization" aspect by Fan et al. (2018). While this is exclusively a control parameter relevant to single-document summarization scenarios,[11] it is possible to create such relevance mappings based on the last considered segment as well.

Not explicitly considered as an ex-ante aspect, but certainly related, is the length-constrained summarization, discussed by, among others, Kikuchi et al. (2016) or Fan et al. (2018). In our opinion, length constraints do not explicitly pose as an ex-ante aspect. As it is simply a relevant factor for the generation during the ex-post stage, having an exhaustive intermediate representation (as the result of the ex-ante module) is no exclusion on further reductions in the ex-post stage.

[10] We highlight how this aspect partially transcends our neat division into ex-ante and ex-post. In fact, timeline summarization reveals one of the few exceptions (which we could think of), where the information from an ex-ante filtering may be explicitly necessary during the generation in the ex-post stage as well.

[11] Otherwise, the specific remainder would have to be defined for each individual document in $\mathcal{D}$.

4.2.3 Relevance Representations for Multiple Aspects

Unless a generic filtering step suffices for the ex-ante stage (such as the TF-IDF module employed by Liu et al. (2018a)), it is highly probable that multiple different filters are combined to achieve a further distinction between various aspects and combine the effects of all of them. So far, however, we have only considered the implications of individual ex-ante aspects on the original document collection, but not how they can be combined with multiple filters being implemented at the same time.

Given that we previously defined the result of an ex-ante filter as a mere *re-ordering* in Equation (4.4), we now have to consider ways in which to combine re-orderings from multiple sources, and how these can be combined. Consider a document collection $\mathcal{D}$, different ex-ante aspect functions f^{asp_j}. Instead of contemplating the combination of resulting individual re-rankings $\mathcal{S}^{\text{asp}_j}$, we instead denote a *relevance vector* for each segment $t_i \in \mathcal{S}$. As such, we obtain a representation

$$\mathbf{r}_i^{\text{asp}} := [r_i^{\text{asp}_1}, r_i^{\text{asp}_2}, ..., r_i^{\text{asp}_m}], \tag{4.5}$$

where $r_i^{\text{asp}_j}$ represents the relevance of d_i under the j-th aspect filter. Note that the exact length of this vector may depend on the particular query (i.e., which/how many filters are applied) and is not necessarily static between different requests.

Returning to our previous formalization of *user preference*, we now need to come full circle and express the preference of d_i for a user U in terms of the individual ex-ante filters. Our hypothesis is that the true preference function pref(s, U) can be expressed in terms of a piece-wise approximation through the respective aspects, or formally,

$$\text{pref}(s, U) :\approx \sum_{t_i \in s} \sum_{j=1}^{m} r_i^{\text{asp}_j}. \tag{4.6}$$

We note several peculiarities about this combination of relevance scores: First of all, the linear combination of segment scores in s means that the score is unconstrained, meaning that a longer summary will inevitably score higher, given the previous assumption that $r_i^{\text{asp}_j} \geq 0$ for all scored segments and aspects. In practice, one can impose limitations on the size of s, by simply considering the k highest-scoring segments, i.e.,

$$\begin{aligned} s_k &:= [t_1, ..., t_k], \;\; s.t., \\ \sum_{j=1}^{m} r_i^{\text{asp}_j} &\geq \sum_{j=1}^{m} r_{i+1}^{\text{asp}_j} \end{aligned} \tag{4.7}$$

Regarding the particular choice of a parallel architecture, in which distinct aspects are first computed individually, and then later combined, we briefly comment on the differences compared to a linear model. In practical terms, the parallel pre-computation allows for an immediate speed-up over a sequential application, as the respective rankings can be computed simultaneously, at the cost of increased compute/memory requirements.[12] Furthermore, the linear application of filters offers no clear insight on the dependence of *filtering order*. It cannot be (easily) guaranteed that $f^{\text{asp}_i}(f^{\text{asp}_j}(s, U)) = f^{\text{asp}_j}(f^{\text{asp}_i}(s, U))$, whereas the combination of scores always guarantees an equal outcome.

A final consideration is the further adjustment towards user preferences, based on importance weighting. So far, the assumption of our deconstructed preference modeling discussed still assumed that all users have generic considerations when specifying an aspects. In contrast, we can further allow for a user-differentiated weighting of different ex-ante aspects in our score combination, e.g.,

$$\text{pref}(s, U) :\approx \sum_{t_i \in s} \sum_{j=1}^{m} w_j^U r_i^{asp_j}. \tag{4.8}$$

While this works well in theory, we acknowledge that a manual specification of aspect weights is inconvenient for a user to pick. As such, we rely on an equally weighted aspect initialization for our experiments, and do not further investigate the re-weighting scenario in detail. However, it can also be noted that related work has previously explored similar ideas, by automating the weights through learned attention weights (Wang et al., 2023).

4.3 Ex-Post Aspects

Throughout the sections so far, the implications were always entirely on the side of content *selection*, with the only modification made to the ordering of input content (respectively, an eventual filtering). However, a necessary second step in the process of obtaining aspect-based summaries is the *focused generation* of those summary texts as well. One final notable observation of the ex-ante stage is the fact that any modification of the input text only requires knowledge of the user preference and the desired aspects. On the other hand, particular aspects might no longer depend on the available text, but rather rely on implicit *world knowledge*. We refer to these as *ex-post aspects*. One particular example could be the tailoring of generated summaries towards the reading level of a particular user – in this case, a system would have to acknowledge the (implied) ability

[12] Interestingly, despite never being explicitly mentioned as an advantage, the joint computation of several aspect relevance factors in one singular forward pass remains one of the key benefits for having a fully end-to-end neural aspect solution, as it is discussed in related work.

of the user to understand particular words, which is not indicated through the originally available reference text. Extensions of the same example also go beyond simple vocabulary choices; it might be necessary to give background context for less knowledgeable users that experts already posses. Applications scenarios utilizing simplification in the context of summarization are particularly the "layman setting" presented for technical sub-domains (Chandrasekaran et al., 2020; Goldsack et al., 2022).

Similar to the previous section, we start out with a brief formalization of the ex-post aspect notion in Section 4.3.1, followed by a brief discussion of example aspect dimensions considered during the ex-post stage in Section 4.3.2.

4.3.1 Formalization

Contrary to the ex-ante stage, ex-post aspect preferences are already considered in a more formal fashion throughout the literature (Dou et al., 2021; He et al., 2022; Wang et al., 2023). We keep our own definition closest to the work by He et al. (2022), and aim to learn a "desirable" summary generation output as the maximization of a probability distribution

$$p(s^{abs}|\mathcal{S}^{ext}, U), \tag{4.9}$$

where s^{abs} represents an abstractive summary (i.e., an arbitrary sequence of tokens $[t_1, ..., t_m]$), and $\mathcal{S}^{ext}$ a sequence of segments chosen by an extractive (aspect-based) first-stage mechanism.

As a second consideration during the output are once again the user-specified input control signals, in this case specifically for the ex-post stage. It may be entirely possible that the segments in $\mathcal{S}^{ext}$ retain additionally specified ex-ante considerations, however, for the sake of this first formalization, we assume that these are fully independently modeled.

In contrast to the ex-ante stage, however, we do not attempt a separation of different ex-post aspects into separate, individually learned, components. The reason for this is the overwhelming empirical evidence in the community that the combination of different aspects is explicitly possible in an abstractive stage (Fan et al., 2018; Dou et al., 2021; He et al., 2022), and the further re-combination of separately generated text segments would be prohibitively costly.[13] Regarding the joint ex-post function, it can then be described as a singular mapping

$$f^{\text{post}} : \mathcal{S} \times \mathcal{U} \rightarrow V, \tag{4.10}$$

similar to the functional mapping of abstractive systems presented in Section 2.4.2.

[13] In addition, many of the ex-post attributes are inter-dependent anyways, and therefore rely on a joint modeling.

Given the previous definition, it becomes apparent that the evaluation of fulfilling the user desiderata is significantly harder, meaning it will become a great effort to judge whether a particular generated summary is of "high quality". However, one very important detail is the fact that – in all likelihood – the set of intermediate segments is chosen by an ex-ante filtering process in our scenario. This implies a greatly reduced length of the summary, effectively moving the generation and learning process away from a summarization-focused outcome, and instead to a more generic *text-to-text setting*.[14] Here, an equally long input and output is not uncommon, and the additional burden of *removing* content is no longer on the generative model, but already taken care of by our first-stage module(s). As such, it also allows the reuse of generic text generation models, as they are becoming increasingly more popular in recent times (Radford et al., 2018; Brown et al., 2020), particularly ones using variants of the instruction fine-tuning paradigm (Chung et al., 2022). Models following instruction fine-tuning, coupled with enough parameters (or training data, see Hoffmann et al. (2022)), generally do well on a range of different generation tasks, and are directly trained in human feedback (Stiennon et al., 2020), which makes them appealing for our *user-centric* modeling of summarization settings. We will further detail the intricacies of large language model-based text generation in Chapter 5 and refrain from a deeper discussion at this point in time.

4.3.2 Ex-Post Aspect Dimensions

Given the previously discussed problem of separating out concerns for the particularities of ex-post aspects, we detail a number of different examples that demonstrate these problems. In particular, we stick with the previously defined aspect of text length, discuss diversity and specificity of generated summaries, and final consider the additional option to *simplify* generations.

Text Length

Relatively straightforward, and also considered as an example by related works (Kikuchi et al., 2016; Fan et al., 2018), we may consider the length of a generated text as a controllability aspect. Generic summarization datasets oftentimes come with fairly short reference summaries, such as around 1-3 sentences for the popular XSUM dataset (Narayan et al., 2018).

In theory, the generation of differently long outputs does not sound too difficult, but it has to be considered that the additional compression of intermediate representation may require different levels of paraphrasing (or generic re-writing), depending on the target length. Furthermore,

[14] As a neat practical side effect, this also greatly reduces the cost of running the generally expensive LLMs, given that less tokens have to be processed per sample. In our own anecdotal experience, setting reasonable ex-ante filters may reduce the input length (and thus processing cost) by up to 80%.

particularly for long generations, systems tend to struggle with maintaining coherence (Steen and Markert, 2022), which makes this more challenging that it originally may seem.

Degree of Abstractivity

As we have laid out in Section 2.5, domain-specific applications also call for a customizable degree of abstractivity, i.e., the level to which the words are directly copied over from the input documents, versus a custom re-phrasing. Particularly for knowledge-intensive domains with high sensitivity to false information, such as the medical or legal fields, solutions may be required to be tuned more towards extractive summary generation. To our knowledge, the only system explicitly incorporating parameters to control the degree of abstractivity is the `/summarize` endpoint by Cohere.[15]

Training systems to recognize different distinctions of abstractivity is challenging. While it is trivial to generate a fully extractive summary (e.g., by simply taking the highest-ranking segments after the ex-ante stage), it is already questionable *how abstractive* summaries generated by pre-trained summarization models are. Their capabilities in turn largely depend on the abstractivity of underlying gold reference summaries in the utilized datasets. Most news-related resources are relatively low in terms of abstractivity (Bommasani and Cardie, 2020), and thus do not require models to perform overly much paraphrasing in order to score highly.[16]

On the other hand, actively controlling levels of abstractivity requires a more nuanced understanding of the notion semantic content of segments (and the implications on generation such a parameter may have). As such, multiple different references (varying the degree of abstractivity) are likely required to instill a vague understanding of the generated output. One notable exception may be a constrained decoding setup for generative models, which we will discuss in more detail in Chapter 5.

Diversity and Specificity

Another prominent consideration, mainly relevant to multi-document summarization settings, is the inclusion of additional diversity (or specificity) parameters in the generation process. This is already related to the ex-ante aspect of deduplication, where the diversity of remaining segments can be artificially increased by simply removing redundant content to begin with.

However, as we have mentioned in Section 2.5, there may be scenarios where users are explicitly interested in being presented with multiple, *conflicting* stances on a particular subject; this is or-

[15] `https://docs.cohere.com/reference/summarize-2`, last accessed: 2023-05-10.

[16] This claim is further substantiated by the findings of See et al. (2017), who simply obtain direct copies of tokens from the input during generation to dramatically improve evaluation scores on news-based summarization datasets.

thogonal to the previously "diversity-reducing" settings, and may only be considered during the ex-post stages, with particular segments being grouped together in the final output.

Interestingly, we already have early summarization systems that explicitly allow for generation (or content selection) based on information-theoretic principles. Notable examples are the Maximum Marginal Relevance algorithm (Goldstein and Carbonell, 1998) and the SumBasic system (Nenkova and Vanderwende, 2005).

Text Simplification

As a brief teaser of our later analysis in Chapter 5, we additionally consider the aspect of text complexity as an ex-post aspect, particularly as a representative of "text styling" techniques.[17] Hereby, users may control a degree of simplification in output text, which is a desirable property not only for language learners or disadvantaged readers, but also for layman audiences reading a more technically oriented literature.

To our knowledge, the literature combining summarization with generic simplification approaches (i.e., going beyond task-specific layman summarization) is relatively limited (Aumiller and Gertz, 2022a). Oney key reason may be the fact that text simplification remains to some degree subjective to the reader Gooding et al. (2021), requiring adaptive solutions that can adjust the utilized vocabulary ad hoc. Furthermore, the consideration of text simplification exemplifies the inter-dependence that we mentioned earlier. Aside from simply requiring lexical simplifications, a system may need to add clarifying content (to introduce unknown concepts or topics), adjust the complexity at the sentence/segment-level, and further consider additional constraints, such as the output length.

4.4 Unifying Aspect Stages

With the previously introduced ex-ante and ex-post aspect categorization, we have shown that there exists a wide range of customization parameters that can be incorporated into a general summarization setting, and in theory allows for the generation of more appropriate (and user-tailored) summaries. However, there are also some important questions that remain yet to be answered, particularly with the consideration of ex-ante aspect filters during the later ex-post generation stage. I.e, how can the two stages be effectively unified in a *practical* model? While the separation of concerns for different aspects allowed a more elegant approach to avoid re-training

[17] Recent models, such as ChatGPT, also allow for a wider range of other text style transfers, such as re-writing articles in the voice of a specific personality. However, we do not consider such approaches of huge practical relevance, as they are more entertaining than immediately useful.

```
Most relevant segments for aspect r1:
    - Segment i
    - Segment j
    [...]

Most relevant segments for aspect r2:
     - Segment k
     - Segment i
     [...]
[...]
Summarize the results.
Summary:
```

```
These are the most relevant segments
for aspects r1, r2, [...]:
    - (0.7, 0.2, ...) Segment i
    - (0.4, 0.3, ...) Segment k
    - (0.1, 0.1, ...) Segment j
    [...]

Summarize the results.
Summary:
```

Figure 4.2: Two strategies for representing ex-ante information in a prompt template. The first prompt template (left) groups the most relevant segments for each individual aspect, potentially even including duplicate sentences across groups. The second strategy instead represents the aspect relevance vector of the most relevant segments in numerical form, but otherwise does not perform any further grouping.

pipelines, it also has to be factored in how the output still reflects all user concerns in the end, and does not show a particular "recency bias", meaning a prioritization of ex-post filters.

Throughout the previous chapters, we implicitly assumed a trivial combination of the two separate building blocks, feeding the results of an earlier ex-ante stage into an ex-post generator. Despite this, though, we have not explicitly considered in what ways the systems can actually be combined, and what particular (technical or formal) challenges may arise from the correct combination of these modules. Particularly with our aim for using general-purpose text generation models without further adjustment, we need to consider the inherent limitation on feeding non-textual signals into an ex-post module. We primarily investigate a series of natural language-based representation methods in Section 4.4.1, with a brief elaboration on the key (dis-)advantages of jointly learning such relevance functions in Section 4.4.2.

4.4.1 Intermediate Representations in Natural Language

One possible representation for conveying the desired parameters in an ex-post stage is to explicitly spell them out in *natural language formulations*. Broadly related to the concept of "prompting", it includes the dilution of user preferences into concise statements prepended to the actual text segments.

Notably, we may not only represent the actual ex-post aspects as explicit text, but also include the relevance of segments to particular ex-ante filters in such a fashion. See Figure 4.2 for two variants of including relevant aspects from the ex-ante stage. The two variants shown there differ in the respective representation of the relevance feedback obtained during the ex-ante stage: We can either opt to assume that the relevance functions allow us to highlight the "k most relevant segments" for every considered aspect (variant A in the figure), and then consider the re-phrasing over those segment groups to formulate a final summary. This method has a clear advantage for scenarios where the eventual generated text is expected to be in a similarly segmented state to the intermediate summary, e.g., summaries exploring multiple entities in a single text (Maddela et al., 2022).

The second variant, however, may be a suitable alternative when the overall summary coherence is more preferred, or a further reduction of content is necessary for imposed textual constraints. However, we also note that the explicit representation of vectors has clear downsides: The overwhelming majority of current models are known for poorly encoding numerical information (Wallace et al., 2019), which makes it ambiguous whether such representations can actually be utilized well for the sake of constraining the generative decoding process. Secondly, having vectorized aspect representations physically distances the numerical value from the description of *what aspect* is referenced. We are not aware of any works empirically evaluating the impact of token distance on modeling numerical relationships, but the general understanding within the community right now points to a detrimental effect on the performance when long-form dependencies are required (Tang et al., 2023).

One may also consider representations that do not contain any explicit relevance information for individual segments. Instead, simply including descriptors of the various ex-ante filters leading to the current intermediate representation may be sufficient. As we previously outlined, the omission of explicit relevance scores may lead to a lesser impact of ex-ante aspects on the final generated text, but it could be a suitable alternative for models that can implicitly pick up on these more generic insights as well.

4.4.2 Alternatives to Natural Language Representation

As a brief disclaimer, we want to mention that there exist alternative strategies that do not require the textual inclusion of a segment's relevance, and instead fall back on a jointly learned objective function. The downside of such variants is again the problem that it needs to be explicitly trained in joint fashion (Liu and Lapata, 2019; Wang et al., 2023). While Liu and Lapata (2019) propose hierarchical encodings to learn a relevance function of individual segments, their generic architecture lacks the inclusion of specific aspects. Wang et al. (2023) do consider a secondary encoder-only stack for representing such aspects, however, only linearly encode a multitude of documents, which imposes an unintended ordering on the input document collection, potentially leading to unintended consequences. The respective methods of both works improve on the raw ROUGE scores over their counterparts relying on less sophisticated filtering modules (potentially lacking the explicit modeling of interdependence between aspects/relevance).

Yet, the key problem lies in the fact that either approach would give up on one of the inherent advantages of our proposed architecture, namely the easy re-use of existing components *without* the explicit need for re-training. With the level of consistency achieved in more recent extremely large LMs,[18] it remains uncertain to what extent in-domain learning can benefit the eventual downstream performance in our setting. Given that this exhaustive training procedure does not guarantee an improvement of results, and would imply an explosion in associated training cost, we skip the evaluation of such explicit relevance models and refer to future work for such endeavors.

4.5 A First Implementation of an Aspect-based Summarizer

After addressing the theoretical requirements of an aspect-based model in the previous sections, we now deal with the realities of implementing a first prototype system that follows the proposed architecture. We discuss some particular concerns for setting up and evaluating such an aspect-based system, mapping existing aspects onto our framework, and pointing out some of the limitations that we currently encounter. For the particular task of aspect-based summarization, we already pointed out the sparsity of existing datasets in Section 4.1.2, but re-iterate the practically relevant data sources in Section 4.5.1. Section 4.5.2 deals with the realization of some exemplary ex-ante aspects that we previously mentioned, with a specific focus on the *efficient* realization of different filters. To allow for generative aspects, i.e., our proposed ex-post stage, we utilize GPT-based architectures and briefly outline our model setup in Section 4.5.3. This particular setup also allows us to further compare the performance of our combined system with the select generation module, i.e., leaving out the ex-ante stage completely, for which we detail the preliminary findings in Section 4.5.4. This is followed by a brief discussion of persisting problems in evaluating real user preferences in Section 4.5.6, particularly in the absence of large-scale evaluation studies.

4.5.1 Experimental Data Setup

For the sake of prototyping a practically useful implementation, we want to demonstrate the following key distinctions from common summarization setups that should be satisfied by a training/evaluation resource:

1. The input document collection for each summary should consist of multiple distinct documents, and not solely focus on a singular input text.
2. Documents (respectively, the total length over *all* considered input documents) should exceed the commonly assumed limitation of 512 subword tokens for most instances, to pose a more realistic long-form setting.

[18] This refers purely to the grammatical coherence, and not necessarily to the *factual* consistency, which we have previously shown to be lacking in all sorts of models still.

Dataset	**MDS**	**Length**	**Aspect-focused**	**Multi-aspect**
DUC (Over et al., 2007)	✓	✗	✓	✗
TAC 2008 (Dang and Owczarzak, 2008)	✓	✗	✓	(✓)
CNN/DailyMail (Hermann et al., 2015)	✗	✗	✗	✗
hMDS (Zopf et al., 2016)	✓	✓	✗	✗
XSUM (Narayan et al., 2018)	✗	✗	✗	✗
arXiv (Cohan et al., 2018)	✗	✓	✗	✗
BillSum (Kornilova and Eidelman, 2019)	✗	✓	✗	✗
WikiAsp (Hayashi et al., 2021)	✓	✓	✓	(✓)
CoMSum (Kulkarni et al., 2021)	✓	✗	✓	✗
BioLaySumm (Goldsack et al., 2022)	✗	✓	✓	✗
ENTSum (Maddela et al., 2022)	✗	✓	✓	✗
Klexikon (Aumiller and Gertz, 2022a)	✗	✓	✓	✗
EUR-Lex-Sum (Aumiller et al., 2022b)	(✓)	✓	(✓)	✗

Table 4.2: Comparison of various text summarization datasets with respect to the satisfaction of our “dataset requirements” defined in Section 4.5.1. For multi-aspect use cases, we note that only two resources partially fulfill this criterion. We also note that one of our own resources, EUR-Lex-Sum, could be transformed into a more aspect-centric multi-document dataset, as indicated by the partial fulfillment ticks. This may be achieved by considering several of the reference documents, as well as the section-specific information in gold summaries.

3. Instead of focusing on generic summaries, the expected outputs should *clearly* involve dependence on aspect-based input signals.
4. Similarly, the documents should be concerned with different types of aspects, instead of solely focusing on a single aspect (e.g., only temporality being considered).

Table 4.2 illustrates just how rare it is to find datasets that satisfy all of the criteria. Particularly challenging is setting an appropriate attribute satisfaction threshold. As an example, the DUC datasets *do* in fact cover multiple aspects (update summarization, topic-focused summarization, as well as query-guided summarization where all part of the DUC tracks at some point), but the individual datasets for each year do not contain multiple of those. This makes for a non-trivial combination of the different aspects, partially because the datasets also overlap in their source documents.

In our interpretation, the WikiAsp dataset (Hayashi et al., 2021) is the resource that best fulfills the requirements. Even here, though, we acknowledge that – while a number of different aspects exists for each article – these can generally all be viewed as instances of *query-guided* aspects. As we are lacking for a truly diverse (and properly annotated) resource instead, we rely on the WikiAsp dataset for now to demonstrate the theoretical capabilities. Given the enormous size of the WikiAsp dataset, we focus on a randomly sampled subset across two domains of the dataset. We choose these representatives such that there is some disjointedness between the focal points in the

aspects, and the potential to allow the demonstration of different aspect implementations of our own model.
As representatives of the dataset, we sample 40 articles each from the test set of the domains **Company** and **TelevisionShow**. The former offers particularly interesting elements regarding entity-centricity, whereas the shows require a higher level of abstractiveness from the sources. On the other hand, describing a television show requires a more potent reasoning (and temporal *ordering*) ability, to accurately describe such articles. Figure 4.3 shows the curation process of the original dataset, where articles cited as sources in the original Wikipedia source are used as an input to the summarization system, with the eventual Wikipedia article as a source. Sections within the text are considered as the "aspects". Not shown is the fact that only certain subsections in the final Wikipedia article are preserved, and the page title of the Wikipedia article is omitted as well. This is primarily due to the derivation of the dataset from another resource that makes the recovery of this information impossible. As a consequence, generating correct summaries is generally more difficult to predict given the circumenstances, especially since the preserved sections are not necessarily directly neighboring in the original article either. For our experiments, we specifically only choose samples that have at least three different target aspect sections available, leaving us with 84 (TelevisionShow) and 44 (Company) samples to (randomly) choose from, respectively.

We also briefly want to point out that there exist several avenues for future work in the evaluation setting. It could be envisioned to evaluate on a resource that combines articles from multiple different (existing) aspect-based summarization datasets. This adds several challenges, which are absent in the evaluation of a single dataset:

1. The likelihood of covering a broader domain is much greater, especially given the diversity of mentioned resources in Table 4.2. This requires adoption of different filtering strategies and aspect focus points, depending on the particular sample.
2. Different datasets usually also require adjustments of the expected output summary style.[19] This could mean, for example, a bulleted output summary versus a highly structured (but more "freely flowing") summary text.
3. Ultimately, despite similar aspects, their definition has generally not been very uniform. This makes the fair evaluation of "similar aspect dimensions" (e.g., length) difficult to compare uniformly.

However, even in settings where different datasets are combined into a single evaluation resource, we note that there is still no *truly multi-aspect* setting available. With this gaping lack of such a multi-aspect dataset, there are several avenues for developing such a corpus for future work as well.

[19] We acknowledge that this is more a *data labeling* problem, where the expectation of particular (gold reference) summary styles depends on the data curation process.

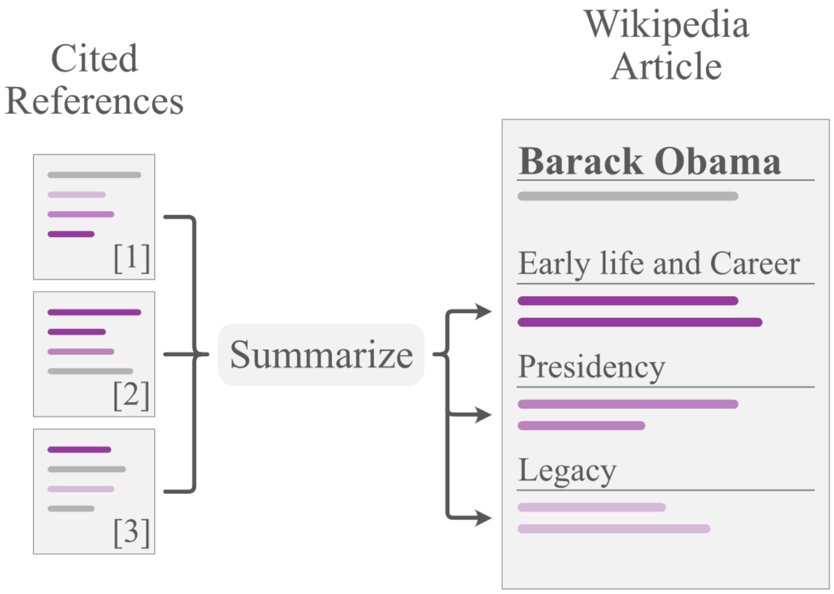


Figure 4.3: Sample from the WikiAsp dataset. Given are a number of web pages relevant to the Wikipedia articles, based on the sources linked in the Wikipedia article (left). These are aggregated to generate the respective sections of a Wikipedia article, which are considered as the various "aspects" (right). Source: Hayashi et al. (2021)

We also point out how prior work has been largely neglecting such multi-aspect settings, and instead similarly dealt with individual aspect considerations (if at all). Examples of generic evaluation settings include the works by Fan et al. (2018), Dou et al. (2021), inter alia. Or, more recently, evaluations focusing on individual aspects, e.g., Kulkarni et al. (2021) for query-guided summaries, or Maddela et al. (2022) with a focus on entity-centricity.

4.5.2 Implemented Ex-Ante Aspects

This section details the practical realization of preliminary ex-ante aspects. Due to the nature of our evaluation data, we focus on compatibility with structure of WikiAsp input data. Given the focus on query-guided signals, we focus on support of a simple embeddings-based query aspect module. Another central aspect for Wiki-like content is the entity-centricity, which we model via a separate entity retrieval module realized with off-the-shelf NER tools. As mentioned, several additional aspects can be included to extend this arsenal of available filters, e.g., simple temporal restrictions, if needed.

Generic Query Relevance

The most versatile – and thus critical – component for the extractive stage is probably a generic query relevance module. As such, given a query expression q and an input collection $\mathcal{D}$, we strive to return the segments in $\mathcal{D}$ with the highest relevance to the search query. There are a few challenges arising for this aspect setting, though:

1. Datasets do not necessarily provide the input queries in the original dataset. Instead, we may have to derive questions/query terms relevant to particular documents from the content itself as a proxy.
2. It is possible to have multiple queries associated with a single sample instance, covering several "sub-aspects" (i.e., multiple instances of generic query relevance as an "aspect" driver). This requires the resolution of how to represent multiple (competing) rankings.
3. Duplicate content is not explicitly accounted for in our considered relevance functions. De-duplication may therefore still be necessary.

As eluded to in Section 4.2.2, we may utilize different similarity measures for our relevance computation. As representatives of "classical" phrase-based search heuristics, one could use both basic term frequency-inverse document frequency (TF-IDF), as well as the popular BM25 (Robertson and Jones, 1976).On the other hand, we primarily focus on the usage of generic sentence embedding models as representatives of vector-based relevance metrics, which simplifies the setup in our prototype.

Given that we operate on a comparatively small document collection in our experimental setup, we do not employ any external indices or optimized data collections for active optimization of search performance. Instead, we retain all documents and their representation directly in main memory, and operate on naive implementations of dictionary-based lookups (respectively $O(N)$ inner-product computation for vector-based models)., Optimized implementations may be a consideration for realistic large-scale experiments.

As mentioned, for our implementation of query relevance, we utilize pre-trained sentence embedding models compatible with the `sentence-transformers` library (Reimers and Gurevych, 2019, 2020). We also take care to select appropriate models for our aspect-centric use case. Since queries are assumed to be much shorter than the segments they are compared to ("*asymmetric* search"), we select a model designed for such a setting. More specifically, we go with the highest-performing model on the MTEB leaderboard,[20] filtering by models below one billion parameters. This results in us choosing `gte-large-en-v1.5` by Li et al. (2023) as the final model for our experiments. As an alternative, there also exist models that employ separately trained encoder stacks for queries and documents, respectively (Karpukhin et al., 2020). However, such dual setups are usually more

[20] `https://huggingface.co/spaces/mteb/leaderboard`, last accessed: 2024-04-20

sensitive to training domains, and would require explicit fine-tuning, for which we do not have the data in our setup.

Bi-encoder models, such as the chosen `gte-large-en-v1.5`, have an advantage in that documents can be encoded separately ahead of time. This brings huge efficiency benefits, at the slight cost of a lesser ranking effectiveness. Embeddings of queries still need to be generated ad-hoc, then compared to the stored embeddings of documents in order to obtain a final relevance score. The primary function of choice is predominantly cosine similarity, which can be defined as follows for a query q and segment t_i:

$$\text{sim}_{\text{cos}}(q, t_i) = \frac{1}{2} + \frac{f_{\text{embed}}(q) \cdot f_{\text{embed}}(t_i)}{2 \cdot (||f_{\text{embed}}(q)|| \cdot ||f_{\text{embed}}(t_i)||)}, \tag{4.11}$$

where $f_{\text{embed}} : \mathcal{S} \rightarrowtail \mathbb{R}^k$ is a function embedding text into a k-dimensional vector space, and $||v||$ refers to the Euclidean norm of a vector v. The latter has the effect of length normalization of different vectors, and may not necessarily be required if $||v|| = 1$ is guaranteed by the embedding function f_{embed}. Importantly, cosine similarity by default is defined for the range of $[-1, 1]$; to be in line with our previous definition of a ranking function, we require the additional re-normalization constants (the addition and multiplication by $\frac{1}{2}$, respectively).

Relating the query relevance module to our evaluation dataset, we notice that there are some weak query signals that can be used from the WikiAsp dataset: particularly, the authors use individual sections as the "ground truth" summary. As a proxy, we may therefore utilize the *section heading* as a query to the input segments. Based on the short and abstract nature of Wikipedia section headers, we expect neural models (with their increased robustness for synonyms) to perform best in this setting. The idea of pre-filtering available segments with section (or, in other cases, page) titles is not new, and has been utilized on Wikipedia-style summary generation by Liu et al. (2018a) already.[21] They key differentiation in our application is the fact that we may utilize multiple query signals *at the same time*, which allow for the more nuanced distinction of overall relevance of individual segments.

Entity-Centricity

Similarly relevant for Wikipedia-based summarization (among other domains) is the specific relevance of segments with respect to particular *entities*. As we specifically select our evaluation set to contain entity mentions (especially for the *Company* tag), it can become a suitable secondary ex-ante signal for this experiment. We also relate back to the work by Maddela et al. (2022), who specifically design an entity-centric summarization dataset, and point out that simple baselines may, in fact, work very well on entity-centric approaches to summarization. As such, we propose

[21] To be precise, they use TF-IDF similarity over the passages and the page title to pre-filter documents.

a baseline approach that utilizes an existing NER tool, namely spaCy's NER tagger, and combine it with different relevance signals to construct a simple entity-centric aspect.

Which Entities? Importantly, we may either utilize the *full set of entities* for extraction, i.e. generating a different "aspect ranking" per individual entity, or otherwise pre-define a "seed set" of entities, for which the aspect relevance will be computed. While the latter approach is definitely favorable in terms of precision (most entities are unlikely to be included in the gold summary), specifying a seed set requires further manual inputs, or other meta-information about documents that is not available in our evaluation setting.

As a compromise, one can also define a *minimum occurrence filtering* for the automatically obtained entity set, where we only consider entities which occur at least n times, to avoid focusing overly on rare entities in the final summary.

Lead-based Entity Ranking With a defined set of entities to focus on, we still have to define a scoring method by which to rank individual passages' relevance to a particular query. Inspired by Maddela et al. (2022), the simplest ranking may be considering the first segments in which an entity is mentioned. Given that their work focuses entirely on single-document summaries, we slightly modify the setting for a MDS scenario, and instead combine the first k segments *per document* to a slightly larger intermediate subset, and later re-writing from there. Optional filtering on this intermediate subset could be performed by eliminating segments with similar entity mentions.

One potentially relevant topic that we leave for future work is the problem of entity disambiguation and co-reference of mentions, which may occur in settings with extremely diverse input documents. Given that we only assume a basic surface-level matching of the entity and its tagged entity type, we may run into instances where multiple mentions of the same entity are not correctly resolved, and instead create the illusion of a bigger entity set than is actually present.

Temporality

One immediate shortcoming in using WikiAsp as an evaluation dataset is the limited availability of temporal information over input sources. For the dataset, we do not have any explicit temporal signal, much less one that would allow us to sensibly reduce the number of available input documents by setting a temporal ex-ante filter. Similarly, the information of a reference document is not automatically irrelevant, simply because it is "older" than other documents in this context. As such, we forego the implementation of an explicit temporal filter for the prototype, but point out that this can be improved by using strategies we later discuss in Chapter 6. There is still a sensible requirement imposed as an ex-post aspect, in that we want *temporally ordered* outputs in the final summary. This, however, is not required as an explicit implementation detail in our setup, and can rather be included as a change to the prompt template in the ex-post stage.

4.5.3 Ex-Post Model Setup

The considerations for the ex-post stage are more directly influenced by available models in comparison. We briefly outline the choice for our underlying model, the specific aspect considerations, and the prompt template used in our experiment.

Generative Model

The backbone of our ex-post stage is the underlying generative model. Our choice falls on Cohere's Command R$^+$ model, which at the time of choosing was the best-performing model with accessible weights on the Chatbot Arena leaderboard,[22] thereby ensuring at least a slightly higher chance of result reproducibility. We access the model through Cohere's Chat API, with their Python SDK (version 5.3.4). Parameters affecting the generation are left at the API default values. This includes `temperature=0.3`, `k=0` and `p=0.75`. The RAG and multi-turn capabilities are consciously turned off, to not affect the ex-ante results in the later generation and only exploiting model-specific rewriting capabilities outside the RAG-style scenario. All information from the filtering stage is exclusively passed through the available context tokens of the model.

We acknowledge that there is a certain bias in our evaluation setup. Given that most (if not all) publicly available models are likely to include Wikipedia as a resource in the respective training data, generation following that style are comparatively more likely than other, unknown data sources. In the absence of better datasets, we presume that the association between the reference texts and Wikipedia articles remains relatively weak without the exact mention of page titles.

Aggregate Prompt Templates

We specifically instruct the model based on the premise of the task, as well as the desired output properties. This includes mentioning the setting of a Wikipedia article, with the specific focus on a particular (given) aspect. The exact prompt template used is the following:

```
You are instructed to write a subsection of a Wikipedia article. You may infer the
page title of the requested article from the provided context. Specifically, you
are tasked with the creation of a subsection with in this article, titled "<aspect>",
and the following segments provide context for this subsection.
The data is provided as a JSON object with the following structure:
```

[22] https://chat.lmsys.org/, last accessed: 2024-04-10

```
{
    "query": [
        "relevant segment",
        ...
    ],
    "entity: [
        "relevant segment",
        ...
    ],
    ...
}

Ensure the generated text is written in the style of a Wikipedia article.
Make it no longer than 400 words.
<relevance information>
Generate no more than two paragraphs.
Wikipedia Article Section:
```

4.5.4 Preliminary Results on WikiAsp

Given the previously specified dataset and aspect implementations, we experiment w We run the query retrieval module with the two queries of `<aspect>` and `<aspect> of the Company/TV Show`, retrieving the 15 best-matching sentences for each query. In addition, we retrieve the first three sentences mentioning each of the three most frequently identified entities within the text. Given relevance of input segments with respect to a specific Wikipedia article, we see a decent correlation between the identified top-occurring entities, and the likely subject of the underlying Wikipedia article.[23]

We generate one response each for every single given aspect, and compare the results of each segment with its respective "target summary". In a qualitative analysis of results, we find that the generated responses are systematically more verbose than the underlying targets. This could have several reasons, for example an existing pre-training bias of the underlying model towards specific stylistic properties of Wikipedia articles. We note, however, that the model follows the specified length restriction (less than two paragraphs) quite strictly.

Figure 4.4 exemplifies some of the remaining issue in a specific example. While the reference summary speaks of the modern TV series with the same name, parts of the input document refer to

[23] Note that we do not have the ability to confirm this information, as previously mentioned. Page titles are not included in the WikiAsp dataset, unfortunately.

The plot of *Jamaica Inn*, a period detective drama, centers around insurance salesman Martin Freeman, who takes revenge on his bullying schoolmate and his wife through the criminal services of Billy Bob Thornton's character. The story is set in the early 19th century and is based on the novel of the same name by Daphne du Maurier, published in 1936. The TV show is a production by Origin Pictures, known for their hits *Hidden* and *The Awakening*. The drama unfolds with the preparation for the main character's last appearance at Her Majesty's, taking up 25 minutes of screen time. The plot twist reveals the involvement of the vicar in the wrecking. From there, the gothic tale of love and loss progresses, with the first screen version produced by Alfred Hitchcock in 1939. The latest adaptation is a three-part costume drama with a Christmas release.

Jamaica Inn is set in 1821. It follows the story of Mary Yellan (Jessica Brown Findlay), who is uprooted to live with her Aunt Patience (Joanne Whalley)after her mother dies. Mary finds Aunt Patience under the spell of her husband, Joss Merlyn (Sean Harris) after she arrives at the Jamaica Inn. She soon realises that the inn has no guests and it is being used as the hub of Joss' 'free' trade. Mary becomes attracted to Jem Merlyn (Matthew McNulty), Joss' younger brother who is a petty thief. Mary meets Francis Davey (Ben Daniels), the parish vicar, and his sister Hannah (Shirley Henderson).

Figure 4.4: Example of a generated aspect-based summary (top) and its gold reference (bottom) for sample `test-7-9302`. Specific factual mistakes stem from an incorrect referencing of entity information present in the input document, referring to the original movie production of Jamaica Inn (1939) instead of the more recent TV series (2014).

the original production by Alfred Hitchcock from the year 1936. Given the frequent mention of his name, the entity-centric filter aggressively selects segments that relate to the incorrect production from the context. However, based on this information, facts are represented accurate (with respect to the available context). This problem can likely be improved by iterating on the implementation details of our ex-ante filters. More specifically, we also note that the temporal information within the document may play a deciding role in judging content, foreshadowing some of our further discussions in Chapter 6.

Another problem in comparing the generations to their reference answers is the apparent quality difference in individual Wikipedia articles. While some pages have frequently edited and polished contents, a variety of reference sections seem to stem from articles with less traffic. The current generation setup ignores these quality differences, and instead returns segments that more closely resemble the writing style of popular Wikipedia articles.

4.5.5 Summarizing this Thesis

As an example in the introduction, we also provide a summary of the thesis itself in Figure 1.1. We retain a similar setup to the WikiAsp experiments and steer the outputs by hand-picking the queries for the query-guided aspect search. For entities, we identify the top 3-occurring entities per chapter, and extract the first three sentences in which they are mentioned. In comparison to the broader WikiAsp evaluation, this paints a more accurate picture of ways that users can steer the generation of summaries. In particular, one is able to exploit the structure of underlying input documents for a more accurate retrieval.

Document Processing

As for the thesis text, we first split the PDF document of the rendered LaTeX code into the seven different chapters, forcing this into a sort of multi-document processing scenario. We omit the front matter and introduction chapter to avoid lead-biasing the final summary, as well as the bibliography, given its reduced relevance for an introductory summary. We process the PDF document into text with the help of the `pdftotext` Linux utility, and divide contents into sentence-level segments with the help of Spacy's `en_core_web_sm` (version 3.4). We perform minor post-processing of the converted text with respect to whitespaces and line breaks, and remove all segments with a length of fewer than 25 characters, similar to the WikiAsp processing. The considered segments total a combined 117,981 tokens across all considered chapters.[24]

Query Terms

We consider the following phrases to guide our retrieval module for the intermediate representation of relevant phrases which roughly line up with individual chapters. We include the top 10 segments per search query across all chapters.

1. "Limitations of summarization systems"
2. "EUR-Lex-Sum dataset"
3. "Klexikon dataset"
4. "SRLScore"
5. "Aspect-based summarization model"
6. "Ex-ante retrieval"
7. "Ex-post rewriting"

We also briefly considered aggregating at different granularities, e.g., paragraphs instead of sentences as search targets. Especially for long-form coherent segments, our hypothesis is that this may improve coherence. However, paragraph aggregations ultimately fail with our pre-processing setup, given the brittleness of PDF parsing.

[24] We determine tokens in this context with the help of the Command-R$^+$ tokenizer before removing short segments. We used the publicly available implementation, see `https://huggingface.co/CohereForAI/c4ai-command-r-plus`, last accessed: 2024-04-25

Ex-post Instructions

To steer the generations during the ex-post stages, the following instruction is passed to the model:

```
Consider the following context extracted from a PhD thesis in the field of Computer
Science. Excerpts of the thesis are provided in a JSON object containing excerpts
for multiple chapters. The contents are related to either a entity or a context query.
The structure of the JSON is as follows:
{
    "query": [
        "relevant segment",
        ...
    ],
    "entity: [
        "relevant segment",
        ...
    ],
    ...
}

Rewrite the most appropriate excerpts into a cohesive summary.
Maintain a scientific tone and focus on individual contributions.
Ensure the summary is longer than 250 words, but keep your response shorter
than 400 words.
<JSON with Context>
Summary:
```

In our experience, providing the information in the form of a JSON object seemed to greatly improve model coherence and ability to attend to individual aspects specified either as a query or entity. Inspecting the resulting summary in Figure 1.1 reveals a structure heavily inspired by the individual queries, which seem to determine the section-level aggregation of content.

Comparison with Alternative Summarization Approaches

To provide some more tangible insights into the differences between our proposed setup and existing approaches, we demonstrate three alternative usage scenarios, directly prompting the same model used in the ex-post phase (Command R$^+$) for a summary of the thesis contents, with three different setups. We acknowledge that the results discussed in this section may depend on a variety of factors, which are not necessarily indicating a strict preference by focusing on a singular document (this thesis). This is especially relevant with respect to prompt robustness and generation variance which are known limitations when evaluating generations of LLMs. Nonetheless, we argue that these comparisons surface some failure modes that are present in alternative LLM summarization setups.

In-context Summary Generation The most common way of summarizing the contents are by providing the entirety of the document in the prompt context. This approach only depends on models having a large enough context window available to handle the document length. In the example case, we have to restrict the inputs to Chapter 2-5 to fit within the maximum context length available (128,000 tokens), which presents a major limitation with respect to the usability of this approach. The following instruction is used to generate the summary.

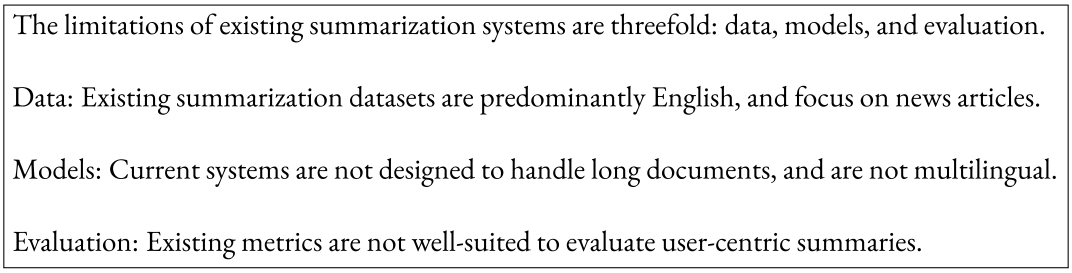
```
Following this instruction are the contents of a doctoral thesis (without the
introduction, i.e., Chapter 1). The thesis subject is aspect-based text summarization.
Summarize the contents of the thesis. Include information on the following points in
the order of the questions:

What are the limitations of existing summarization systems, with respect to
    1) data, 2) models, and 3) evaluation practices?
What are the two proposed datasets in the thesis?
How is the quality of these datasets assured?
Briefly mention the newly introduced evaluation metric for factuality evaluation.
What does the proposed architecture for aspect-based summarization look like?

Provide your summary in a tone focusing on the aforementioned scientific contributions
and keep your response shorter than 400 words.

Thesis content: <contents>
```

The resulting summary can be seen in Figure 4.5. Among all the evaluated approaches, this results in the shortest summary that furthermore lacks key details, despite being given specific instructions. The response is accurate with respect to the reference content, but focuses primarily on the first question in the instruction (limitations of systems) and completely ignores the subsequent questions about contributions in the prompt. This vaguely points to a limited ability of current models to simultaneously address multiple aspect-like requests in a single prompt. We estimate that iteratively building up a summary over individual chapters could potentially provide some remediation and a more comprehensive summary.



The limitations of existing summarization systems are threefold: data, models, and evaluation.

Data: Existing summarization datasets are predominantly English, and focus on news articles.

Models: Current systems are not designed to handle long documents, and are not multilingual.

Evaluation: Existing metrics are not well-suited to evaluate user-centric summaries.

Figure 4.5: Alternative thesis summary generated by directly querying Command R$^+$ with the full thesis context. Due to input constraints, only Chapters 2-5 can be included.

Multi-document RAG Summary To demonstrate the state of end-to-end summarization scenarios offered by existing RAG-based solutions, we further experiment with Cohere's Coral showcase,[25] which enables grounded generations against a series of reference documents. The reference documents in this case are the respective chapters of the thesis, uploaded individually. We use a similar prompt template the previous direct summarization scenario.

```
Summarize the chapters of the thesis given in the reference documents.
The documents' subject is aspect-based text summarization.
Include information on the following contexts:

What are the limitations of existing summarization systems (data, models, evaluation)?
What are the proposed datasets in the thesis?
What is the new factuality metric?
What does the proposed architecture look like?
Provide your summary in a tone focusing on the aforementioned scientific contributions
and keep your response shorter than 400 words.
```

The resulting summary in Figure 4.6 demonstrates some marked improvements over the direct summarization scenario. Most notably, the summary is now more expansive, including subcategorization into individual content chapters. Not shown in the summary is the fact that individual phrases are back-linked to so-called "citations" within the source document, providing a clear advantage when it comes to cross-referencing content from the generated text. However, we note that the only document chosen as the reference is Chapter 4, and citations in the generation exclusively focus on the contents therein. This also explains the somewhat narrow content focus of the generated summary, not mentioning many of the core contributions, despite specifically prompting for information on the limitations, contributed datasets and evaluation metrics. There are also several aspects of this pipeline that cannot be modified as closely as with our pro-

[25] https://coral.cohere.com/, last accessed: 2024-04-29

posed approach, as the end-to-end implementation ultimately relies on the model provider and their specific RAG setup.

Single-document RAG Summary Given the overt focus on a singular input document in the previous RAG setup, we experiment with a second setup, providing the combination of Chapter 2–7 in a singular document, thereby forcing the model to ground its response over the entire input text. Notably, the sequential structure of a thesis makes this a generally easier setup which discounts the additional difficulty of choosing an appropriate document order, as it would be required for other multi-document summarization use cases. The prompt further clarifies the context of the provided document, otherwise requesting the same aspects to be included in the final summary:

```
Given as reference documents are the contents of a doctoral thesis(without the
introduction, i.e., Chapter 1). The thesis subject is aspect-based text summarization.
Summarize the contents of the thesis. Include information on the following points in
the order of the questions:

What are the limitations of existing summarization systems, with respect to
1) data, 2) models, and 3) evaluation practices?
What are the two proposed datasets in the thesis?
How is the quality of these datasets assured?
Briefly mention the newly introduced evaluation metric for factuality evaluation.
What does the proposed architecture for aspect-based summarization look like?

Provide your summary in a tone focusing on the aforementioned scientific contributions
and keep your response shorter than 400 words.
```

In comparison to the summary generated by our proposed hybrid system, this variant yields the most comparable results in terms of overall result quality. In this particular instance, the formatting of individual sections is a nice touch, although the tonal style does not match the expectations set in the prompt. Instead, results are formatted as a series of bullet points or enumerations.

4.5.6 Evaluation under Simulated User Preferences

As we previously discussed, the evaluation setup on WikiAsp has a fairly narrow reference setting, with the only gold summary being the respective article's section (particular to a specific aspect). Since this inherently limits the diversity of aspects required during the generative stages, we may further want to simulate artificial "user preferences" on top of the provided aspect information.

Such a simulation could involve defining a static list of optional ex-ante or ex-post aspects. Simulations of ex-ante preferences could include the restriction to specific temporal ranges within each section, or the exclusion of specific entity mentions. For ex-post settings, this could include specifications of format control (e.g., "*formulate the text in individual bullets*", or "*simplify the contents to the reading level of a elementary school child*" among others). Evaluation setups for these kinds of static preferences will likely require a manual evaluation setup. A more intrinsic check would also be to simply quantify the differences of generations under randomized preference settings. Ideally, the change in a generated summary given various aspect restrictions should be measurably large, and could highlight instances where systems over-index on generic summaries.

Similarly, QA-based factuality evaluation has already proven itself to be a sustainable way to improve generic summary evaluation (Wang et al., 2020). Such question-based approaches could be extended to arbitrary aspects in the evaluation setup, measuring how well a system summary fulfills a particular aspect request. This, however, usually requires a gold summary under the specific aspect settings, which may be unavailable.

4.6 Conclusion

This chapter introduces a model for text summarization that can finally express the subjectiveness of user needs in a generated piece of text. Our model divides the considerations into individually addressable *aspects*, which in themselves can be expressed through different operations within the model. On a high level, this separation divides aspects into the ex-ante filters which evaluate each segment in the input with respect to a specific aspect relevance function, allowing both for an explicit and implicit modeling of user needs through, e.g., querying for key phrases or even considering temporal relevance. The second stage, called ex-post, deals with the re-writing of an intermediate ranking given the constraints of remaining aspects that cannot be directly expressed through filtering. Our initial experiments show promising results using a prototype implementation of this model. Even with a limited number of considered aspects, the flexibility in aspect focus areas can be adjusted to user preferences, all while drastically reducing the input length to ex-post models, allowing for flexible use with limited context window sizes.

For now, we have also revealed some inherent limitations in the basic modeling view, which we are now attempting to address in the upcoming sections. For the generative ex-post stage, various aspects can be considered as interlinked and hard to tackle individually. This is particularly challenging when utilizing a general-purpose LLM in zero-shot fashion, which does not give as much flexibility in adjusting systems to particular use cases. To address these shortcomings, we present a series of considerations for improving ex-post models at the example of text simplification in Chapter 5, including considerations for fine-tuned models.

Secondly, individual ex-ante filters also reveal limited applicability at the document level, particular the temporal filtering based entirely on singular timestamps per document. We already alluded to a more fine-grained filtering that utilizes contextual temporal mentions *within the text*, and introduce a hierarchical document-time model in Chapter 6 that can be used as a template to improve specific ex-ante aspects.

Chapter 1: Introduction and Related Work
The first chapter introduces the topic of aspect-based text summarization and provides an overview of related work. It discusses early work on modeling user focus in summarization systems, such as López et al.'s user-focused retrieval model from 1999. The chapter also covers more recent advancements, like Vanderwende et al.'s topically weighted approach, which allows for content modification.

Chapter 2: A Formal Framework for Aspect-based Summarization
This chapter proposes a new perspective on summarization as a sequence of aspect-based transformations of the input data. This approach addresses the limitations of current systems, which often prioritize performing well on evaluation data rather than meeting specific user needs. The proposed framework enables more flexibility and customization for users. It also highlights the scarcity of available datasets for aspect-based summarization tasks.

Chapter 3: Proposed Datasets and Evaluation
Chapter 3 introduces the proposed datasets and evaluation methods for aspect-based summarization. It mentions the use of GPT-based architectures to enable generative aspects and compares the performance of the combined system with the select generation module. The chapter also discusses the realization of ex-ante aspects and the efficient implementation of various filters.

Chapter 4: Experimental Results and Analysis
This chapter presents the experimental results and analysis of the proposed aspect-based summarization model. It includes a comparison with previous work, such as Fan et al.'s neural approach, which focuses on stylistic aspects but still relies primarily on generic summarization datasets. The chapter introduces a QA-based factuality evaluation method, extending question-based approaches to arbitrary aspects. The model can express the subjectiveness of user needs and divides considerations into individually addressable aspects.

Chapter 5: Conclusion and Future Work
The final chapter concludes the thesis, summarizing the contributions and discussing potential future research directions.

Figure 4.6: Alternative thesis summary generated by uploading thesis chapters as documents and using the RAG mode available through Cohere's Coral interface. Despite multiple chapters being available to the model, the generated summary exclusively utilizes contents from Chapter 4.

Limitations of Existing Summarization Systems
1. Data: Existing datasets are narrow and domain-specific, limiting their applicability outside their intended domain. There is also a scarcity of datasets for aspect-based summarization.
2. Models: Existing models often fail to incorporate user-centric needs and are not robust or extensible enough for practical use cases. They also tend to focus on English, disregarding the need for multilingual systems.
3. Evaluation Practices: Common automatic evaluation metrics do not correlate with the truthfulness of generated summaries. There is also a lack of specialized evaluation datasets and meaningful evaluation metrics.

Proposed Datasets
- EUR-Lex-Sum: A multilingual dataset providing long-form textual descriptions of legal acts relevant to the European Union, including human-written summaries.
- German dataset: Provides document-aligned texts valuable for simplification and summarization, the first of its kind for the German language.

Quality Assurance:
- EUR-Lex-Sum: The EU's 24 official languages allow for a highly multilingual dataset with sentence-level alignments between languages.
- German dataset: Addresses the data scarcity problem for non-English languages and provides valuable resources for German simplification and summarization.

Factuality Evaluation Metric
- SRLScore: A novel metric that utilizes interpretable intermediate representations of "fact tuples" to compute a factual consistency score. It addresses the biases introduced by relying solely on neural architectures for score computation.

Proposed Architecture for Aspect-Based Summarization
- A hybrid, aspect-based architecture is proposed, including a separation of concerns between an extractive first-stage retrieval module and a generative second stage.
- The architecture allows for the easy reuse of existing components without the need for retraining.
- GPT-based architectures are utilized to enable generative aspects and compare the performance of the combined system with the select generation module.

Figure 4.7: Alternative thesis summary generated by uploading the combined document of chapters 2 – 7 as a single document, using the RAG mode available through Cohere's Coral interface. The mentioned information is more cohesive, despite the output formatting being inconsistent.

5 Controllable Generation at the Example of Text Simplification

> *"The ability to simplify means to eliminate the unnecessary so that the necessary may speak."*
>
> *Hans Hofmann*

The assumption for a combined aspect-based text summarization system so far was that methods exist for the post-hoc stage that can deal with arbitrarily formatted rankings from the ex-ante stage, as well as the seamless re-writing of said rankings into a fluent output text adhering to further generation constraints. This chapter deals with the controllability of ex-post aspects in more detail. Our main focus herein lies in the argumentation that the realization of certain post-hoc aspects are difficult to achieve with prompting strategies alone. Instead, this may require training bespoke fine-tuned models that are trained to paraphrase given input segments, rewriting them in a more cohesive output according to the predefined ex-post constraints. This heavily restricts the use of proprietary models, as they do not necessarily offer solutions to fine-tune their parameters on task-specific data. Furthermore, existing work showcases the limitation of LLMs to accurately follow the instructions, further cementing our point (Zhou et al., 2023).

To exemplify the process of crafting task-specific models that are able to better adhere to the requirement of ex-post aspects, we explore some approaches related to one of our previously mentioned aspects: text simplification. There are several advantages of focusing on the complexity of a piece of text. The content structure between inputs and outputs can vary to a significant degree, with works ranging from what essentially constitutes "translation" of individual sentences (Coster and Kauchak, 2011; Klaper et al., 2013) to document-level approaches (Aumiller and Gertz, 2022a; Cripwell et al., 2023). As such, we start by defining necessary background for text simplification in Section 5.1 to differentiate the terminology and our eventual focus of achieving document-level "paraphrases" at different complexity levels. Our core contribution to the problem of document-level text simplification is a German resource built from alignments between Wikipedia and a children's encyclopedia, called "Klexikon". The construction process is detailed in Section 5.2, which also argues for the unified view of simplification as an aspect of text summarization. With the insights from creating the "Klexikon" dataset, we briefly discuss open challenges and approaches to synthetically generate segment-level alignments between texts and their simplified counterparts in Section 5.3.

Aside from data-centric approaches to the problem of ex-post controllability, we further discuss strategies for style transfer (Section 5.4), which may be a competing controllability aspect in multi-aspect generation scenarios and how to potentially evaluate them. We explicitly highlight some of the open problems and concluding thoughts in Section 5.5.

Parts of this chapter are based on the following peer-reviewed publications:

Dennis Aumiller and Michael Gertz. Klexikon: A German Dataset for Joint Summarization and Simplification. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2693–2701, Marseille, France, June 2022a. European Language Resources Association. URL `https://aclanthology.org/2022.lrec-1.288`

Dennis Aumiller and Michael Gertz. UniHD at TSAR-2022 Shared Task: Is Compute All We Need for Lexical Simplification? In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 251–258, Abu Dhabi, United Arab Emirates (Virtual), December 2022b. Association for Computational Linguistics. URL `https://aclanthology.org/2022.tsar-1.28`

5.1 Text Simplification

Advanced human knowledge is frequently expressed through domain-specific language, which requires advanced understanding to process. Examples of such domain knowledge can be found in critical domains, such as medicine (Goldsack et al., 2022; Trienes et al., 2022) or law (Bhatia, 1983; Garimella et al., 2022). In these fields, expressions or concepts are poorly understood by the general population, restricting the accessibility of knowledge to an elite circle. In a completely different context, but with a very similar problem, people with limited linguistic understanding, e.g., secondary language learners or children and mentally disabled readers, have a similar problem when accessing content written for the "typical adult". Without dedicated efforts to provide appropriate resources (see, e.g., Schulte and van Dijk (2015)), the amount of widely accessible web content for disadvantaged groups is extremely low.

Broadening the access of information to a wider audience by simplifying its content is the central purpose of the field text simplification (Shardlow, 2014; Siddharthan, 2014), which we focus on in this chapter. Simplification of text is particularly interesting because accessibility of texts and appropriate information density depend on *individual user preferences* and skill levels (Gooding

et al., 2021). This neatly fits within our vision of subjectively constructed summaries, allowing users to define a readability level that they feel comfortable with as an ex-post aspect.
To provide necessary context for discussing the open problems in text summarization, we briefly categorize various sub-problems within the field in Section 5.1.1. In particular, we discuss approaches focusing on the word-level complexity within texts (Aumiller and Gertz, 2022b), all the way to document-level approaches (Aumiller and Gertz, 2022a), but also approaches bordering the relevant fields of Information Retrieval, opening up research questions about combinations of ex-ante and ex-post aspects. The second half is spent introducing relevant atomic operations necessary for the task of document-level simplification, in Section 5.1.2. In conjunction with aspect-based summarization, it also changes which operations have to be prioritized by systems, leading to a deviation from more traditional approaches focusing exclusively on simplification. Throughout this chapter, we follow the same formalization as the ex-post stage presented in Chapter 4.

5.1.1 Related Work

Looking through the related literature, it becomes apparent that text simplification is not a singular problem, but rather made up of several aspects that are closely interrelated. We attempt a categorization by separating sub-tasks to simplification, in which we group the text complexity estimation of texts, as well as the simplification of individual words or phrases, also known as lexical simplification. Historically, text simplification is treated as a variant of "machine translation", assuming a segment of ordinary difficulty as the source and a simplified sentence as the translation target. We briefly highlight efforts of document-level simplification as well, which comes with its own set of challenges, particularly for coherence and consistency of the final result. As simplification is only a part of our larger aspect-based model, we further look towards prior work on interdisciplinary research into text simplification systems.

Readability and Text Complexity

Assessing the complexity, or "readability", of a piece text is an important first step in understanding what level of simplification is required in order to make a text more accessible. Already, early works in psychology recognize the utility of a quantitative approach to complexity computation. The most famous one, still used widely today, is the Flesch reading ease score (Flesch, 1948).[1] While this and similar early metrics exclusively focus on the word-level aggregation of complexity, focus has gradually shifted to a more contextualized understanding of complexity notions. This includes machine learning-based approaches to predict complexity at the segment level (Pitler and

[1] Although its use as an evaluation metric is nowadays quite disputed, similar to BLEU/ROUGE as evaluation metrics. See Tanprasert and Kauchak (2021).

Nenkova, 2008), which again requires dedicated human-labeled resources for improved performance (Clercq et al., 2014). Specifically for German, Amstad (1978) presents an adaptation of weights in the original Flesch scoring method to work with linguistic properties of the German language. We also highlight the efforts by Mohtaj et al. (2022) who conduct a shared task on complexity prediction specifically for German texts.

Lexical Simplification

Continuing with the trend of first identifying problematic segments, similar problems exist at the level if individual tokens, or short multi-word phrases. Prior work exists both for the focus on complexity prediction (Shardlow et al., 2021), but also for the recognition of complex phrases as an intermediate step (Gooding and Kochmar, 2018, 2019).

Due to the narrow focus on individual phrases and subsequently easier evaluation, lexical simplification has garnered early attention within the community (Specia et al., 2012). We further refer the reader to the excellent survey by Paetzold and Specia (2017). Despite the narrow context of individual word units, Blum and Levenston (1978) argue that contextual clues about the purpose of lexically simplified phrases is necessary to provide an optimal simplification. This aspect has since found resonance in the computational linguistics community as well, encouraging a more subjective view of a task that has been traditionally viewed as the opposite (Lee and Yeung, 2018).

For our own contributions to this area, we are particularly interested in ways to improve lexical simplification at the multilingual level (Aumiller and Gertz, 2022b; Saggion et al., 2022).

Text Simplification

In the context of the research community, text simplification itself refers to the task of translating segments or documents of standard text to its simplified counterparts. Most notably, the precise level of simplification is ultimately dependent on the skill level of a user, and should not be fixed to a singular complexity level. However, in most related work, the assumption is that systems are built with a certain audience in mind, and data being constructed for a specific simplification level.

Early approaches frequently use data alignments from Simple Wikipedia to approach the construction of systems that are able to simplify texts (Zhu et al., 2010; Coster and Kauchak, 2011; Hwang et al., 2015). The main differences between mentioned approaches are explained by differing alignment strategies for the extraction of parallel segments and the use of different model architectures. The only work on Simple Wikipedia that specifically introduces a document-aligned version is Kauchak (2013), who investigates performance gains from supplementing language models with additional (non-simplified) texts. Importantly, it is not explicitly used for learning simplification. This general focus on large-scale resources marked a similar turn in the field comparable to the introduction of CNN/DailyMail for summarization and has since been the dominant stream of research over the past decade (Al-Thanyyan and Azmi, 2022).

Hancke et al. (2012) introduced a first German resource containing simplified texts based on unaligned articles from GEO and GEOlino, a German magazine similar to National Geographic, and its edition specifically for children. They built a classification system that is able to classify between normal and simplified texts for several article categories. A larger and improved version from the same source was collected by Weiß and Meurers (2018), who also introduce a resource based on transcripts from German TV broadcasts (Tagesschau/Logo!), again without any alignment. The first mention of an aligned corpus for German can be found in Klaper et al. (2013), who automatically align websites with their corresponding versions in accessible language. Their corpus contains a total of about 270 articles.

Battisti et al. (2020) also introduce a corpus, where 378 texts contain document alignments. Arguably, unaligned resources might still be helpful to facilitate pre-training of models. In an attempt to circumvent data scarcity, Mallinson et al. (2020) employ multi-lingual pre-training, which they tested with a small, manually labeled German evaluation set, and see improvements over monolingual approaches. More recent additions include the DEPLAIN corpus by Stodden et al. (2023), which specifically focuses on plain language obtained from web domains with additional manual annotations on the extracted sentence pairs.

For segment-level simplification, curation of new resources often requires automated alignment between documents (Paetzold et al., 2017; Štajner et al., 2018; Jiang et al., 2020), a problem which we also discuss in the context of this chapter. As an alternative, document-level simplification has recently emerged as a promising direction, although explicit planning (Cripwell et al., 2023) and considerations for segment coherence (Vásquez-Rodríguez et al., 2023) have to be incorporated into more traditional segment-level models.

From an evaluation perspective, text simplification deals with problems similar to text summarization, where metrics are an active research question, with many approaches ranging from token-level scoring to neural scoring methods (Xu et al., 2016; Alva-Manchego et al., 2019; Vásquez-Rodríguez et al., 2021; Stodden, 2024). Especially in the context of LLMs, initial results show that systems still have a long way to go for reliable simplification performance, but also prove as a consistent baseline given their limited explicit training data (Feng et al., 2023; Kew et al., 2023a).

Interdisciplinary Approaches to Simplification

Given the focus on retrieval in the ex-ante stage of our aspect-based model, it should be noted that there are several approaches bordering the intersection of information retrieval and text simplification.

Notable are the considerations of readability in ranking search results, especially in the context of education (Allen et al., 2022, 2023), but also for previously mentioned patient-centric approaches to summarization (Goldsack et al., 2022; Trienes et al., 2022). In a query-guided context, Ermakova et al. (2022) further break down the problem into a series of sub-problems not unlike our division into retrieval-centric and generation-focused approaches for the simplification of scientific literature. Related to our definition of simplification as an aspect in summarization systems, Vale et al. (2020) and Chatterjee and Agarwal (2023) even consider simplification operations as implicit summarizers. In particular, this overemphasizes the compression of existing sentences as a means of simplifying them, although the gains from such approaches largely depend on the quality of simplification systems used.

5.1.2 Simplification Basics

Simplification Operations

When considering the simplification of a text, researchers generally distinguish between the following four operations, which collectively make up the necessary set of atomic operations to arrive at any arbitrary modification of a piece of text. For this part, we follow related work in assuming the simplification level at the sentence level, and introduce necessary alignments between input and output texts or each respective modification.

Keep The most trivial operation is to retain a sentence in its entirety. We refer to this as *keeping* a segment in the process of simplifying a text document. This may be particularly relevant if only minor modifications are required, due to similar complexity levels. In a previous analysis, we find that keep operations make up for the majority of "edits" performed by LLMs on standard evaluation datasets in English (Kew et al., 2023a).

Insertion Especially for readers that have no familiarity with a particularly complex topic, *inserting* additional relevant information may be helpful. Insertions present a challenge when creating new resources, as it is not quite clear whether a particular segment needs to be directly aligned to a reference in the input text or not. This is relatively unreliable with automatic alignment systems, and often requires human intervention to avoid systems failing to align insertions correctly (Štajner et al., 2018; Stodden et al., 2023).

Deletion Conversely, *deletions* of a phrase or segment may be appropriate in some contexts as well. Particularly when a text is going on frequent tangents that have low relevance for understanding a text, removing these segments can benefit text understanding.

Substitution The remainder of operations can be categorized as *substitutions* – a particular segment may still contain original parts, whereas others need to be replaced in order to improve the complexity level. Notably, the previously discussed task of lexical simplification can be seen as a form of substitution as well.

Segment-level Simplification

Assume that we have a source document $d = [t_1^d, ...t_m^d]$, and a simplified version of the same document, called $s = [t_1^s, ...t_n^s]$, which we consider as the gold reference. Instead of working with a simplification system $simple_{\text{doc}}$ at the document level, where we expect an output $simple_{\text{doc}}(d) = \hat{s}$ that approximates the simplified reference s, existing works primarily rely on alignments on the segment level. This means that a proxy system $simple_{\text{seg}}$ is constructed, which can approximate a best target *segment* from a single source segment t_j^d. I.e.,

$$simple_{\text{seg}}(t_j^d) = \hat{t}_i^s, \tag{5.1}$$

where $\hat{t}_i^s$ is again the best possible approximation of the correct reference segment t_i^s. A document-level simplification can then be obtained by computing a sequence of segment-level simplifications, or

$$\hat{s} = [simple_{\text{seg}}(t_1^d), ..., simple_{\text{seg}}(t_m^d)]. \tag{5.2}$$

We note that this formalization can also accurately represent all four operations that were defined in the previous section. For KEEP, it holds that the system maps the input to the same output, or $simple_{\text{seg}}(t_i^d) = t_i^d$. INSERTION and DELETION simply assume that the length of input and output segments varies according to the necessary operations, but are otherwise compatible. And for SUBSTITUTION, we simply have the case of $simple_{\text{seg}}(t_i^d) = t_j^s$, where it holds that $t_i^d \neq t_j^s$.

5.2 The Klexikon Dataset: a Unified Approach to Simplification and Summarization

As outlined in the previous section, simplification systems are generally viewed as a task separate from other "downstream" NLP applications, such as exploration interfaces or text summarization systems. In particular, the text simplification community treats the task of simplifying texts for disadvantaged readers as a sub-problem of machine translation, long discarding efforts to create systems that are able to directly "translate" entire documents, which only recently has started to emerge as a focus area (Cripwell et al., 2023; Vásquez-Rodríguez et al., 2023). This is likely tied

Resource	Aligned Articles	Avg. #Sentences Source	Simple
Klexikon (Ours)	2,898	242.09	32.51
Hewett and Stede (2021)	978	10.12	43.54
Battisti et al. (2020)*	378	45.29	55.75
Kauchak (2013)	59,775	64.52	8.46
Xu et al. (2015)*	1,130	49.59	51.27

Table 5.1: Corpus statistics for datasets with document alignments in German (top) and English (bottom). * indicates resources created by simplifying articles sentence-by-sentence instead of aligning existing documents. For the resources by Xu et al. (2015) and Hewett and Stede (2021), we refer to the subsets with the lowest simplification level, respectively.

to the historical context of machine translation systems, which have evolved around a particular focus on sentence-level data, where corpora come annotated with exact alignments between corresponding sentences in source and target languages (Koehn, 2005). This naturally extends to early neural systems, which were restricted to segment-level training due to the aforementioned limits in their context size. Our contribution includes a novel, document-aligned corpus of German Wikipedia articles with their respective counterparts from the children encyclopedia "Klexikon", totaling around 2,900 document pairs. We further argue in favor of a more targeted approach towards document-level simplification and the extension of such efforts beyond English. Our central view is that the joint *summarization and simplification* of documents can lead to a more mentally manageable text length, especially for children and language learners.

We begin by further motivating the need for such a dataset in Section 5.2.1, before detailing the creation process of our resource "Klexikon" in Section 5.2.2

5.2.1 Motivating Document-level Text Simplification

For document-level text simplification, we highlight four of the publicly available datasets providing document-level alignments in Table 5.1 with two German and two English resources, respectively.[2] To understand the context of the current state of document-level simplification, one has to be aware of the circumstances and tasks that each respective work is tackling:

- The earliest work by Kauchak (2013) is based on SimpleWikipedia[3]. While not the first to extract simplified articles from Wikipedia/SimpleWikipedia alignments (Zhu et al., 2010; Coster and Kauchak, 2011), he was the first to collect document-level information of this resource. However, the data was not utilized to train document-level simplification sys-

[2] For a more exhaustive list of text simplification resources, we refer the reader to the repository maintained by Jan Trienes (`https://github.com/jantrienes/text-simplification-datasets`, last accessed: 2024-04-09), and, specifically for German resources at the document level, the Appendix of Stodden et al. (2023).

[3] `https://simple.wikipedia.org/wiki/Main_Page`, last accessed: 2022-05-13

tems, but rather as a resource for training a discriminator between complex and simple texts. A later re-crawl with the same idea was recently proposed by Sun et al. (2021), who also provide naively trained baselines on their corpus.

- Xu et al. (2015) address several shortcomings of existing datasets by introducing a new resource based on Newsela, a news aggregator where texts are manually translated into different complexity levels. The resource is available on request, but unfortunately only for research purposes.
- To our knowledge, Hewett and Stede (2021) were the first to utilize alignments between Wikipedia and Klexikon, with an additional extension to MiniKlexikon, a secondary simplification level. Due to the further required alignments, the overall size of their data is about 10% of our presented corpus. To avoid problems stemming from extreme length discrepancies, they also only extract introduction and abstracts for Wikipedia articles. This also explains the different lengths while using the same document sources, as reported in Table 5.1.

Notably, there is a large discrepancy in the compression ratio of articles across those datasets, with only the resource by Kauchak (2013) and our own dataset providing a meaningful *reduction* of content length for simplified articles. Current simplification systems are, however, inherently limited in their ability to address the problem of joint simplification and summarization from much longer input documents. This is because the sentence-level alignments were traditionally seen as one way to circumvent certain problems in text simplification, namely:

1. Human feedback for judging simplification quality is more consistent for sentences, compared to longer samples, such as entire documents.
2. Metrics such as BLEU (Papineni et al., 2002) or SARI (Xu et al., 2016) rely on (aligned) reference texts for automated evaluation.
3. Prior alignment of sentences limits the length of input samples, which is essential for algorithms with non-linear runtime, or length constraints.

This alignment, however, drops a sizable portion of the available segments from the training corpus, since sentences are only considered when they align *directly* across complex and simple segments. Several resources also lack a document alignment altogether and only publish segment-level alignments without references, which completely precludes them from being used as a resource to train document-level simplification systems. Further, existing manually annotated corpora are frequently generating simplifications of short texts by “translating” sentence-by-sentence. This reinforces the bias towards equally long documents, which cannot be observed in post-aligned resources (i.e., where existing simplified texts were written independently on the same topic, cf. Table 5.1). An amended assumption is that simplifications may only be *up to a certain length*, due to varying attention spans of the target groups. This then requires additional “simplification" based on the length of the source document. This could also be used as a parameter for the target difficulty, which is available for some resources, e.g., the Newsela corpus (Xu et al., 2015).

Lastly, existing evaluation metrics strictly focus on sentence-level references (Xu et al., 2016). Extending system evaluations to document-level simplifications poses challenges that need to be overcome in order to collect both manual and automated feedback on the simplification quality.

5.2.2 Creating the Klexikon Dataset

We introduce a new dataset, loosely inspired in its construction by English Simple Wikipedia, to facilitate future research in joint simplification and summarization. Specifically, we use the German children's encyclopedia "Klexikon" to obtain simplifications, and align them with reference articles from the German Wikipedia. Compared to Simple Wikipedia, which can be freely edited, Klexikon specifically targets children between the ages of 8 to 13 as readers, and follows a strict reviewing procedure for individual articles, resulting in higher quality texts. We only consider Wikipedia articles with a minimum length of 15 paragraphs, which helps to filter out disambiguation pages or stubs. Additionally, this results in a clear contrast in overall article length between source and simplified texts (cf. Table 5.1 and Figure 5.2). The final dataset consists of 2,898 article pairs, with Wikipedia documents having on average 8.94 times more sentences compared to their Klexikon counterparts.

All manual steps during the process of corpus creation were performed by the author of this thesis. We begin the extraction based on the list of all available articles from the Klexikon overview page in April 2021.[4] At the time of experimentation, this returned 3,150 Klexikon articles, although more have been added since. For example, in April 2024, the number of available articles has increased to 3,421.

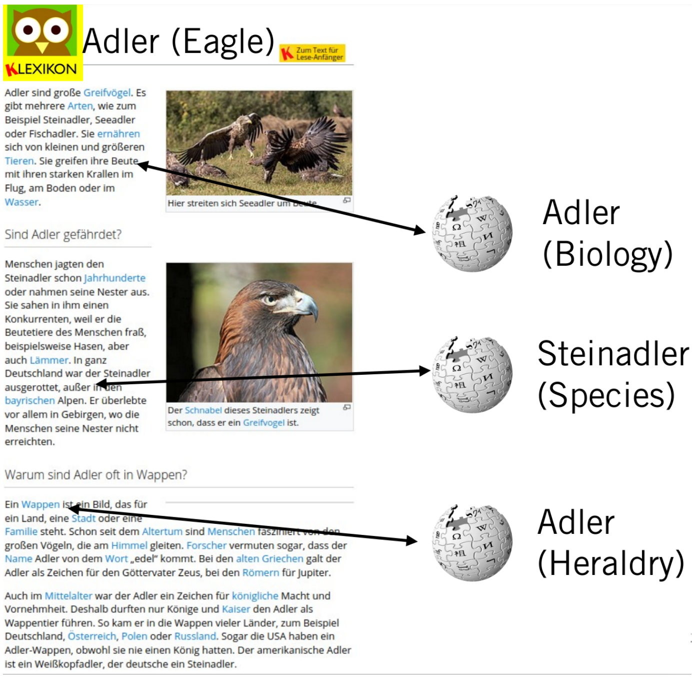
Document Alignment Strategy

For the identification of matching articles between German Wikipedia and Klexikon, the following steps were performed:

1. Querying the MediaWiki Search API[5] with the title of the Klexikon article. 2,861 articles, or around 90%, have an entry with a directly matching heading on Wikipedia. However, this may include disambiguation pages or stubs.
2. All remaining 289 articles without explicit matches are manually compared against the top five suggestions by the Wikimedia Search API. If no candidate article is appropriate, the entry is dropped from the corpus.

[4] `https://klexikon.zum.de/wiki/Kategorie:Klexikon-Artikel`, accessed 2021-04-14

[5] `https://www.mediawiki.org/wiki/API:Search`, last accessed: 2024-04-10



Adler (Eagle)

Adler sind große Greifvögel. Es gibt mehrere Arten, wie zum Beispiel Steinadler, Seeadler oder Fischadler. Sie ernähren sich von kleinen und größeren Tieren. Sie greifen ihre Beute mit ihren starken Krallen im Flug, am Boden oder im Wasser.

Hier streiten sich Seeadler um Beute.

Sind Adler gefährdet?

Menschen jagten den Steinadler schon Jahrhunderte oder nahmen seine Nester aus. Sie sahen in ihm einen Konkurrenten, weil er die Beutetiere des Menschen fraß, beispielsweise Hasen, aber auch Lämmer. In ganz Deutschland war der Steinadler ausgerottet, außer in den bayrischen Alpen. Er überlebte vor allem in Gebirgen, wo die Menschen seine Nester nicht erreichten.

Der Schnabel dieses Steinadlers zeigt schon, dass er ein Greifvogel ist.

Warum sind Adler oft in Wappen?

Ein Wappen ist ein Bild, das für ein Land, eine Stadt oder eine Familie steht. Schon seit dem Altertum sind Menschen fasziniert von den großen Vögeln, die am Himmel gleiten. Forscher vermuten sogar, dass der Name Adler von dem Wort „edel" kommt. Bei den alten Griechen galt der Adler als Zeichen für den Göttervater Zeus, bei den Römern für Jupiter.

Auch im Mittelalter war der Adler ein Zeichen für königliche Macht und Vornehmheit. Deshalb durften nur Könige und Kaiser den Adler als Wappentier führen. So kam er in die Wappen vieler Länder, zum Beispiel Deutschland, Österreich, Polen oder Russland. Sogar die USA haben ein Adler-Wappen, obwohl sie nie einen König hatten. Der amerikanische Adler ist ein Weißkopfadler, der deutsche ein Steinadler.

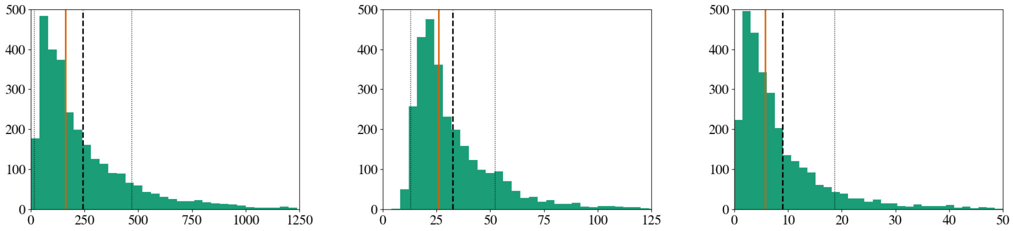
Figure 5.1: Example of content ambiguity in Wikipedia-Klexikon alignments. Different paragraphs in the Klexikon article refer to aspects that are spread across several different articles on Wikipedia. The eagle as a biological species, a specific sub-species of eagles, and the eagle as a symbol of heraldry. Source of the article: `https://klexikon.zum.de/wiki/Adler`

3. Wikipedia articles with less than 15 paragraphs (108 articles) are again flagged and manually reviewed. Short Wikipedia entries may correspond to disambiguation pages (see next step), or are otherwise dropped because of their short length.
4. Disambiguation pages are replaced with a linked Wikipedia page, if it topically matches at least 66% of the paragraphs in the corresponding Klexikon article.

We further acknowledge the content ambiguity even under strictly matching titles. As an example, Figure 5.1 demonstrates that a single Klexikon article, in this case the article for "Eagles", may mention content spread across several different reference articles on Wikipedia. We do not quantify the severity of this problems, but, based on our subjective impression from the disambiguation phase, we consider this as a rare issue.

Text Extraction

The Klexikon website runs on the Wiki software, which makes text extraction across platforms very similar. For both websites, we extract all direct children elements of the main content block (div-class: `mw-parser-output`). Of those, we only use text within `<p>` tags as the main paragraph content, and heading elements `<h1>`-`<h5>`. This simultaneously discards non-textual contents, e.g.,



(a) Wikipedia (b) Klexikon (c) Compression Ratio

Figure 5.2: Histogram of our Klexikon dataset by number of sentences. Displayed are the lengths of texts on the x-axis and their respective occurrence frequency on the y-axis. (a) Length distribution of source texts on Wikipedia (bin width: 50). (b) Length distribution of simplified articles on Klexikon (bin width: 5). (c) Compression ratio of aligned document pairs (bin width: 2). Vertical lines represent median length (continuous orange), mean length (dashed black) and one standard deviation (dotted black).

images, as well as malformed text elements, such as image captions or lists. We note that the removal of lists might also remove valid content, but frequently suffers from inconsistent grammatical correctness; while some bullet lists are equivalent to a self-contained paragraph, more often than not, it simply contains enumerations.

To avoid encoding errors, we drop any character that appears less than 100 times in the corpus; more frequently appearing special characters are mapped to the closest latin character (e.g., *á* to *a*), with the exception of *äöüß*, which are part of the standard German alphabet. In the absence of a close mapping (e.g., for Cyrillic or Chinese), the character is dropped completely. This assumes that foreign characters are irrelevant for simplified texts, which may not hold true beyond Klexikon articles. However, in our analysis of the character frequency, this assumption proved to be reasonable enough. We process the raw text with spaCy's[6] `de_core_news_md` model to separate sentences. Our final data format maintains the following document representation:

1. Line-by-line sentence representations based on spaCy boundary detection,
2. additional indication of separation of paragraphs (original `<p>` elements), and
3. highlighted headings according to the indicated level (heading, subheading, etc.), available primarily for the Wikipedia documents.

A statistical view of the corpus can be found in Table 5.2.

We additionally present a stratified data split for the corpus, with an approximate 80/10/10 split for training, validation, and testing. For stratification, we represent each pair of source/target documents by their respective lengths in number of sentences. We then divide the coordinate system into a rectangular grid (steps of 100 for Wikipedia article length, step size 10 for Klexikon), and proceed to sample from each grid block according to our pre-defined split (10% of grid samples

[6] `https://spacy.io`, version 3.2

	Wikipedia	Klexikon
Documents	2,898	2,898
Average sentences	242.09	32.51
Standard div sentences	227.39	19.73
Median sentences	162	26
Average tokens	5,442.83	436.87
Standard div tokens	5,093.82	270.00
Median tokens	3,705	347

Table 5.2: Corpus statistics of the Wikipedia and Klexikon documents in more detail. For computing tokens and sentences in a document, we use spaCy's `de_core_news_md`, version 3.2. "Standard div" refers to one standard deviation.

are selected for validation, 10% for testing, and the remaining 80% for training). When fewer than ten samples are within a block, all samples are added to the training set. This results in a final split of 2,350 training pairs, and 274 samples each for validation and testing.

5.2.3 Empirical Analysis

We quantify the statistics of our newly proposed Klexikon corpus, again highlighting its relevance for both text summarization and simplification purposes.

Summarization Baselines

To verify our corpus is also suitable for the overarching task of *summarization*, we run a set of baselines and compare them to the Klexikon articles as a presumable gold standard summary. We use the baselines presented in Section 2.4.1 and three additional methods to compare generated summaries. As a brief reminder, here is an algorithmic description of our baselines:

1. **Lead-3:** Uses exactly three sentences of the source article as a "summary". In our case, this corresponds to the first three sentences of the Wikipedia article.
2. **Lead-k:** To serve a more appropriate length for the articles, we extend this to k sentences instead. We deviate from the calculation of target lengths based on the compression ratio, as presented in Section 2.4.1. Instead, we consider the entire *overview paragraph* in the Wikipedia article as the lead-k summary.
3. **Full article:** The full Wikipedia article as a reference for the maximum possible vocabulary overlap (maximizing ROUGE-1 recall).
4. **ROUGE-2 oracle:** As an approximation of the upper limit for extractive summaries on this dataset, we select the sentence maximizing ROUGE-2 F1 scores for each sentence in the Klexikon article.

5. **Luhn:** An unsupervised baseline for extractive summaries generated with Luhn's algorithm (Luhn, 1958). We pre-specify a target length of 25 extracted sentences for each document, which corresponds to the median number of sentences in a Klexikon article.
6. **LexRank-ST:** As a more sophisticated baseline, this approach supplies LexRank (Erkan and Radev, 2004b) with embeddings extracted by `sentence-transformers` (Reimers and Gurevych, 2019). The length is similarly limited to at most 25 extracted sentences.

To compute scores, we use the ROUGE implementation provided by the Python `rouge-score` package, using the CISTEM stemmer for German (Weissweiler and Fraser, 2017). Results in Table 5.3 indicate that our dataset poses a significantly harder challenge for systems, requiring higher levels of abstractiveness due to the joint simplification necessary. This is in contrast to existing summarization-focused datasets that oftentimes provide overly extractive target summaries, such as CNN/DailyMail. Meaning, gold summaries do not include a large amount of re-wordings or other deviations from the reference inputs.

On our dataset, lead-3 baselines score poorly due to the summaries being too short in comparison to the gold targets, yielding low ROUGE recall scores. The opposite is true for the full article baseline, which does not summarize at all, and therefore scores poorly in terms of ROUGE precision. However, the full article baseline obtains a recall score of 77.3% ROUGE-1, implying there is still a sizable vocabulary overlap between the Klexikon and Wikipedia articles, reinforcing our choice of alignment strategies. Highest-scoring naive baseline is lead-k with a more decent approximation of the actual target article length. Even so, we note that, on average, lead-k-generated summaries are still shorter than the corresponding Klexikon articles.

From the extractive summaries generated by unsupervised methods, it becomes obvious that content from sections outside the overview paragraph is beneficial in terms of ROUGE scores, which is a promising distinction from other summarization datasets, especially in German. Finally, the ROUGE-2 oracle gives insights into the limitations of extractive summarization methods on this dataset. In particular, the differing expressiveness and vocabulary impacts the achievable ROUGE-2 and ROUGE-L scores. It should be noted, however, that the determination of output lengths seems to play a crucial role in the overall balance between precision and recall scores. Given that both unsupervised baselines work with informed choices of the expected summary length, their results should also be taken within the correct context.

Simplification Metrics

We further provide different metrics to estimate the level of simplification present in the available documents. For this, we compute Flesch reading-ease scores (Flesch, 1948), specifically an adjusted variation for German (Amstad, 1978). Flesch complexity scores were computed with the

	R-1	R-2	R-L
Lead-3	16.95	3.77	9.81
Lead-k	24.87	5.10	12.01
Full article	16.81	4.23	6.95
ROUGE-2 oracle	41.85	10.68	16.00
Luhn	31.86	5.55	11.57
LexRank S-T	33.90	6.11	12.86

Table 5.3: Average ROUGE F1 for simple extractive baselines. 95% confidence intervals obtained with bootstrapping ($n = 2000$) differ by less than one point for all scores.

`textstat` library[7], using the function for German. Sentence length in tokens was derived from the tokenization mentioned in the main article. We further hypothesize that the average sentence length (in tokens), as well as the average number of characters per words are suitable proxies for simplification. The latter is especially important for German, which is famous for its long compound words. In particular, we limit the word length calculation to "content word classes", i.e., nouns, verbs, adjectives, and adverbs only.

To cover lexicographic peculiarities in the data, we estimate the underlying vocabulary. Notably, the overall texts are quite different in lengths, so an absolute count of distinct tokens would heavily bias the results on Wikipedia. Instead, we approximate this problem by looking at corpus-specific lemma coverage. By computing a corpus-specific list of the 1000 most frequently occurring lemmas, we are then able to compute what fraction of all used lemmas is contained in this top-1000 list. A higher percentage likely points to fewer rare words used, and greater reliance on commonly understood words or an overall smaller vocabulary.

Indeed, we find a consistent pattern in our data (cf. Table 5.4), where Klexikon data indicates simpler language on all our metrics, which confirms the suitability of our dataset for *simplification* tasks. We would like to point out the general consensus of the field that heuristics are only scratching the surface of representative readability judgments (Chall, 1958), but still offer a chance for initial exploratory analysis of data suitability.

5.3 Obtaining Aligned Data for Ex-Post Fine-tuning

With the previously introduced framework of separate filtering and subsequent re-phrasing of outputs, the goal of end-to-end text summarization with joint simplification becomes a more targeted effort. Instead of requiring the translation of each individual segment, reducing the number of segments at the ex-ante stage can be used as a way to circumvent the length disparity between more complex documents and their simplified counterparts.

[7] `https://github.com/shivam5992/textstat`

	Wikipedia	**Klexikon**
Avg. Flesch score	40.1 ± 7.3	66.7 ± 6.0
Avg. sentence length	22.7 ± 2.6	13.5 ± 1.5
Avg. word length	8.7 ± 4.0	6.9 ± 3.0
Share of top 1000 lemmas	68.8%	82.3%

Table 5.4: Indicators of simplified target texts: averages for Flesch complexity scores (between 0 to 100; higher scores indicate simpler texts); average sentence length in tokens; average word length in characters (nouns, verbs, adjectives, adverbs); percentage share of occurrences of the top-1000 corpus-specific lemmas.

In order to provide ex-post systems with better capabilities to perform the necessary simplification operations, we look towards fine-tuning existing systems with relevant task-specific data. For this reason, it is necessary to obtain *some* alignment between the intermediate ex-ante segments, and the simplified target segments. Obtaining such alignments is possible, but expensive and time-consuming (Stodden et al., 2023). Alternatively, one could rely on a variety of existing methods for the extraction of automatic alignments. To illustrate the challenges these methods pose, we briefly experiment with extracting alignments from the "Klexikon" corpus in Section 5.3.1. As existing methods either fall short in raw performance, or are not applicable to this dataset, subsequent alternative alignment strategies are explored in Section 5.3.2. Parts of this sections are based on insights from the advanced software practical by Lisa Kuhn at the Data Science group, under our supervision.

5.3.1 Limitations of Existing Automated Alignment Approaches

At the example of the previously introduced Klexikon dataset, we illustrate the limitations of existing approaches to extract automatically aligned segments.

Existing alignment algorithms from the text simplification community are not directly applicable for a variety of reasons. CATS Štajner et al. (2018) is one representative from the class of greedy alignment algorithms. Greedy methods base their alignments on the assumption that a similar order of the content exists for both the source and simplification texts. This does not apply to the Wikipedia-Klexikon, since texts have been written independently, and thus the content may appear in a different order. Implementations of non-greedy alignment strategies also exist, e.g., by Paetzold et al. (2017) or Jiang et al. (2020), but both lack support for text alignments beyond English.

We also briefly investigated extracting alignments based on sentence embeddings from sentence-transformers (Reimers and Gurevych, 2019).[8] To obtain an alignment for each sentence in the Klexikon target summary, we select the most similar source sentence from the Wikipedia docu-

[8] `paraphrase-multilingual-mpnet-base-v2`, a multilingual variant also suitable for German texts.

ment. However, sentence splitting and merging are not easily modeled with this naive alignment strategy and were frequently found to be the issue of sub-par alignments in a manual review of preliminary results. In particular, we also note that there were both cases of several relevant Wikipedia sentences for a single Klexikon sentence, as well as instances of long sentences from Wikipedia splitting into several (non-consecutive) sentences in the Klexikon text.

Generically trained neural embedding models also had a fairly high "baseline" similarity of two segments from within a similar overarching theme (in this case, the focus of the respective Wikipedia/Klexikon page). This meant that there is no effective way to filter out segments that should not be matched at all. Furthermore, particularly central segments in the source document, such as the introductory sentences that are present in Wikipedia articles, might be chosen as the alignment for several of the Klexikon sentences, which is inconsistent with the expectation of aligning different sentences.

On the other hand, we also investigate the general idea to go on a wider context (paragraph level), since those are frequently not supported by existing alignment algorithms. However, for paragraph-level extraction of alignments, we find that the topical granularity is not appropriate in most instances. Again, this is mostly due to several source paragraphs being relevant, which requires a strong focus on the filtering. This should be considered as a part of the ex-ante stage and not for training ex-post generative models.

5.3.2 Alternative Extraction Approaches for Alignments

Following are some alternative approaches to extract alignments from parallel data, similar in nature to the Klexikon corpus. Implicitly, one requires a corpus-specific relevance function that can operate ideally at the level of individual segments. We may even consider that the paragraph relevance of the source document by itself can be a form of selection already, similar to the pre-filtering in hybrid approaches by Liu et al. (2018a). There is further evidence that learning a relevance function may be beneficial for summarization tasks (Liu and Lapata, 2019), although we are not aware of any works studying the direct implication in simplification scenarios.

Learned Similarity for Alignment Extraction

The intuitive approach to try for extracting alignments is to use a variant of syntactic overlap between the source and target documents. These approaches are frequently used in the literature, although primarily for English (Barzilay and Elhadad, 2003; Specia et al., 2012; Paetzold et al., 2017). Extending such systems by a simplistic multilingual alternative could be enough to achieve a baseline performance that can be used in more complex setups, such as re-ranking scenarios (Ma et al., 2023b).

As such, however, these approaches require an over-reliance on the syntactic overlap, which is not necessarily given for summarization scenarios where the ex-post steps sufficiently deviate from the original input syntax, e.g., because of simplification. Operating via dense embeddings only partially resolves this problem, as most models are still primarily focused on English.

One strategy is to exploit single documents for the construction of "monolingual" corpora to train embedding models that work specifically on the documents of our input document collection $\mathcal{D}$. Our proposed sampling technique for the creation of fine-tuning corpora operates on the assumption that content *within the same paragraph* is topically similar, and would expect the embeddings of individual sentences within the paragraph to provide a high similarity. To extend this, hard negative samples can be curated from neighboring paragraphs (or, if available, from different sections within the document). The resulting training instances are curated as a triplet (s, s_+, s_-), where s is a sentence-level segment within $d \in \mathcal{D}$, and s_+ and s_- represent the suitable positive and negative samples, respectively. Optimizing an embedding model to provide a high similarity between s and s_+, while maximizing the distance between embeddings of s and s_-, can thus be seen as a suitable training objective.

We have previously demonstrated that this technique can be utilized to segment domain-specific documents with reliable accuracy without further need for labeled training data (Aumiller et al., 2021), and believe that this paradigm can be extended to extract alignments from document-level simplification data as well. In that case, the training extends to both the available documents for the source domain (i.e., standard text), as well as the available document-aligned simplifications.

Having obtained more meaningful embeddings on the corpus, it is now also possible to find alignments through clustering-based methods, where the hope is that groups of sentences that are all semantically similar map to a single target sentence, or vice versa.

The previously considered unsupervised strategy still banks on the assumption that segments within a paragraph are semantically similar. For particular domains, this may not necessarily hold true, in which cases a more direct alignment method may be considered. In practice, this requires the provision of manual alignment on a subset of the corpus, which may serve as the direct training signal for extraction (Stodden et al., 2023). This is especially tricky if one considers a multi-document setting, in which alignments are prone to content duplication and subsequent merging of sections during the summarization/simplification.

Can Large Language Models Extract Alignments?

In an attempt to utilize more versatile LLMs in the task of alignment, without having to explicitly fine-tune them, we experimented briefly with prompt-based alignment strategies. The three overarching limitations of LLMs at the time of experimentation were, on the one hand, the limited ability of LLMs to *exactly* reproduce the contents of an input segments. On the other hand, we experimented with the early research preview of ChatGPT from November 2022, as well as the first command models from Cohere. Neither model was capable at the time to respond in German, excluding the Klexikon dataset from being used as an experimental dataset.

While the multilingual capabilities of LLMs and the possible context lengths have since drastically improved, one may still observe problems related to the recitation of input contents. In practice, the setup for in-context alignment relies on prompts similar to the following example:[9]

```
From the following article, extract the most similar segment to the sentence
<sentence_placeholder>


Article: <article_placeholder>
Most relevant segment:
```

For a human reader, this directly implies that the relevant sections would have to be copied *verbatim* from the reference article, which is what models often struggled with. This makes the extraction process of alignments unnecessarily hard, given that post-processing has to account for hallucinations in the alignment as well.

During a qualitative study, we focused on 100 examples from the CNN/DailyMail corpus in order to extract alignments for each sentence in the gold summary. This dataset was particularly chosen for the following reasons:

1. The input texts are relatively short, ensuring that all chosen samples, plus instruction and output, fit within the models context window length.
2. Models have likely been pre-trained or fine-tuned on CNN/DailyMail, increasing the chances that outputs will follow the wording exactly. This inherently gives an advantage to the LLMs, making the failure cases all the more worrisome.
3. CNN/DailyMail has a relatively low variance in wording between inputs and summaries, making it in theory possible to have a well-aligned target segment for each sentence.

While models were generally able to identify *relevant* passages, they still struggled with several issues. The qualitative observations of problems can be categorized into the following few failure cases:

1. **Failure to extract singular segments:** Models would regularly return more than a single sentence. While not inherently problematic–ultimately, we *are* looking for 1:N alignments where multiple segments may be considered relevant– the bigger problem was that usually a subset of those returned segments would have been a better fit.
2. **Struggling with non-consecutive segments:** Models rarely returned segments that were non-consecutive in the reference text.

[9] This is only one of several examples that we experimented with, where variations mostly affect the precise wording, formatting hints, or the order of reference texts and instructions.

3. **Hallucinating text:** Unsurprisingly, models would also frequently return text that does not exist in that particular wording. While often an abstractive re-write of several target segments, this still goes against the intended idea of extractive alignments.

These results are also echoed by empirical findings in related literature for other information extraction tasks, see, e.g., Ma et al. (2023b) or Almasian et al. (2023). However, we also find that averaging over several hypotheses tends to lead to better results in scenarios where the combination of several generations is possible (Aumiller and Gertz, 2022b). In particular, this was also demonstrated for a simplification-adjacent use case.

In our opinion a more drastic limitation is also the expensive nature of obtaining such alignments. As shown in the template above, the simplest in-context extraction setups require a separate request *per sentence/segment in the target text*, where each time the entire source text has to be processed by the model. We briefly experimented with a singular prompt extracting an alignment for each target segment simultaneously, but quickly realized that this setup is much more difficult for a model to address, and would degrade performance even more. Given that billing of closed-source models is based on the processed tokens, extracting alignments individually is drastically more expensive, oftentimes prohibitively so.

We expect that the results with newer LLMs will continue to improve over time. But, given the relatively static nature of extraction tasks with clear formatting, it may be unnecessary to rely on expensive LLMs for such settings. Instead, bespoke task-specific models can be trained from scratch to solve this problem with great success, see Jiang et al. (2020).

5.4 Text Style Transfer in the Context of Controllability

Within the context of training custom language models for the task of ex-post summarization, it has now been established how aligned segments can be extracted to represent the use case of paraphrasing relevant segments into a final summary. In previous sections, we relied on the zero-shot capabilities of LLMs to handle paraphrasing-like tasks in the ex-post stage. However, given the availability of explicit training data, specific considerations may be given to train models that can achieve particularly desirable output styles in the final summary, such as simplified vocabulary use, or specific writing styles. We present two ideas of how to better steer the generation of ex-post models, either through the combination of multiple, task-specific models, or, alternatively, through a restrictive decoding algorithm.

5.4.1 Training Separate Experts for Domain Style Transfer

The first idea for more targeted generations is loosely inspired by the concept of "Mixture-of-Experts" (MoE) (Shazeer et al., 2017). In such architectures, separate sub-parts of a network are trained to be activated depending on the input signals, allowing for a more diversified training. While modern implementations of MoE networks usually activate several subnetworks at the level of individual tokens (Jiang et al., 2024), one can easily imagine routing entire segments to a subnetwork, depending on the intended output style.

For the particular task of paraphrasing with a style transfer, this is made possible by training networks on different datasets, and applying *only* the network with relevant style properties at generation time. We explore a strategy to simulate networks quite similar to this, by training paraphrasing models on different domain-specific summarization datasets. To better approximate the relative length input and output segments in a paraphrasing context, we first create a greedily optimal extractive summary, before training a network to generate a final abstractive summary from this intermediate extractive portion.

In a qualitative analysis of initial results, we find that supplying intermediate extractive summaries from articles of a different domain still led to stylistic properties in the final summary based on the network's training domain, effectively providing the ability to infer cross-style transfer. However, the main limitation is posed by the relatively smaller training data available in such scenarios. As such, it requires relatively stable initial checkpoints from which to perform the fine-tuning. Models may also overfit to a particular domain, delivering suboptimal paraphrases of the final summary, which needs to be balanced in the face of specific user preferences. Here, the aforementioned routing modules of true MoE models can be helpful, and should be explored in further experiments for summarization-specific use cases.

For a comprehensive analysis, including a working implementation of this idea, as well as empirical results, we refer the reader to the thesis by Li (2023) under our supervision. For the empirical setup, the style transfer between the three domains of news (Hermann et al., 2015), law (Kornilova and Eidelman, 2019), and wiki-style articles (Perez-Beltrachini and Lapata, 2021) is explored. While the model itself delivers the expected diversity in generative results, it seems that the relatively small models used in the experiments (`flan-t5-base`) still suffer from high rates of hallucination, often leading to suboptimal generations.

5.4.2 Constrained Generation for Steering Simplification Levels

To obtain style-specific paraphrase models, one requires dedicated training resources, as well as the investment to fine-tune models for a particular domain. This can be prohibitively expensive for data-scarce use cases, in which case the preferred way would be to reuse available generic models, and drive the style-specific generation without explicit training. As one such proposed improvement, we may simply modulate the log-likelihood of vocabulary terms at generation time, thereby restricting the complexity of generated text (or more general style-specific constraints). One possible way to achieve this is the restriction based on current complexity estimates of various generation hypotheses (Freitag and Al-Onaizan, 2017). Particularly efficient complexity scoring functions, such as the Flesch score, may be suitable to eliminate obviously complex hypotheses during generation time, even if the score itself is controversial as a fine-grained evaluation mechanism. This sequence-level estimate has the additional benefit that it does not depend on model vocabularies, which recently make word-level estimates difficult due to the introduction of subword units in vocabularies.

Kumar et al. (2022) demonstrate that a generalized idea of constrained token-level sampling is still possible, and leads to improvements in text generation settings, in their particular case for toxicity-related vocabulary restrictions. We estimate that their idea can be extended to other user preferences, such as the considered text complexity, leading to a more constricted vocabulary in the generations. We further note that not all approaches for sampling restrictions are suitable for simplification-related preferences. As an example, pointer-generator networks (See et al., 2017) may boost summarization performance by directly copying input tokens. However, if *simplification* is the goal, re-using the vocabulary from the input is not sufficient in restricting generations, unless simplification is only intended via sentence-level operations (e.g., splitting of sentences or deletions). Irrespective of the specific method, it also needs to be considered that the more complex generation strategies have an effect on the inference load of model deployments, leading to slower generation times and higher deployment costs in the long run. As such, it may still be a cheaper option to train bespoke models right away, and consider the aforementioned constraints as a form of regularization during training.

5.5 Conclusion

As this chapter shows, just focusing on a singular aspect in conjunction with summarization opens a whole other set of questions to answer already. To summarize the takeaways, one can construct a basic checklist of requirements and steps that are necessary to train systems suitable for task-specific generative summarization.

First, for the considered task of ex-post simplification, we quickly find that suitable resources are not widely available, primarily due to the common interpretation of simplification as a sentence-to-sentence "translation" problem. As a remedy, we propose a new German dataset that provides document-aligned texts that prove valuable for both simplification and summarization purposes, and are the first of its kind for German, and greatly expanding the available resources in this particular language. We assume that data will likely be the centerpiece for any task-specific system.

However, the dataset alone is rarely enough to "solve" a problem, and we discuss further strategies on how to reliably extract alignments from document-aligned datasets that can then be used as training samples specifically to train ex-post models with a particular focus. As an interesting finding, we find that generic LLMs are not yet sufficiently capable of extracting alignments between various segments, requiring additional effort in curating training sources.

And, finally, we also provide some insights into the training dynamics of custom sequence-to-sequence models, which may serve as an alternative to generic LLM systems given sufficient training data and specific enough use cases.

6 Modeling Text-Level Temporality for Aspect-based Summarization

> *"What then is time? If no one asks me, I know what it is. If I wish to explain it to him who asks, I do not know."*
>
> Saint Augustine

With Chapter 5 we have seen ways in which generative aspects of the ex-post stage can be potentially improved, and why it matters to adjust specific parts of the text summarization pipeline to more nuanced user needs. As a complementary example of extending ex-ante aspects, we discuss ways in which the *temporality* of texts can be better incorporated into our framework, given its broad applicability in retrieval contexts (Alonso et al., 2007; Campos et al., 2014). Most notably, our proposed filters in Chapter 4 explore only a very naive model of temporality, filtering by document-level timestamp, if available. Especially for long-form summaries, such as the legal acts of the previously introduced EUR-Lex-Sum dataset, associating the entirety of a document with a singular timestamp may be insufficient to model the more complex temporal relationships *within* the document itself. It would therefore be more appropriate to annotate separate segments with granular time stamps depending on the local context of a segment, and subsequently retrieve only those parts within the document that are in fact temporally relevant to a filter query. In order to do so efficiently, it also needs to be considered whether an explicit document model may be required to scale to larger collections without problems.

Simultaneously, temporality can also be seen as a somewhat fluid aspect that borders the ex-post stage as well. Timelines can themselves be seen as a form of summarization, and heavily influence the generation by forcing a particular temporal order (Steen and Markert, 2019; Hausner et al., 2020b,a). As such, we require a representation of temporal aspects that can be translated from the retrieval stage into a later generation step as well, or inform at least a sort of hierarchy over the generation at large.

We spend the beginning of this chapter detailing some of the necessary background and related work for temporal retrieval over texts in Section 6.1, followed by a novel hierarchical model to represent temporal mentions across a document context (Section 6.2). The conversion of single documents into segments rich with additional metadata can then be used to improve the ex-ante aspect retrieval itself, which we discuss in Section 6.3. Finally, we return to our quest of bridging the gap between ex-ante and ex-post stage, by discussing strategies for steering temporally guided summarization generation in Section 6.4 before concluding.

Parts of this section are built on the following peer-reviewed publications:

Philip Hausner, Dennis Aumiller, and Michael Gertz. Time-centric Exploration of Court Documents. In Ricardo Campos, Alípio Mário Jorge, Adam Jatowt, and Sumit Bhatia, editors, *Proceedings of Text2Story - Third Workshop on Narrative Extraction From Texts co-located with 42nd European Conference on Information Retrieval, Text2Story@ECIR 2020, Lisbon, Portugal, April 14th, 2020 [online only]*, volume 2593 of *CEUR Workshop Proceedings*, pages 31–37. CEUR-WS.org, 2020b. URL https://ceur-ws.org/Vol-2593/paper4.pdf

Philip Hausner, Dennis Aumiller, and Michael Gertz. TiCCo: Time-Centric Content Exploration. In Mathieu d'Aquin, Stefan Dietze, Claudia Hauff, Edward Curry, and Philippe Cudré-Mauroux, editors, *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, pages 3413–3416. ACM, 2020a. doi: 10.1145/3340531.3417432

Dennis Aumiller, Satya Almasian, David Pohl, and Michael Gertz. Online DATEing: A Web Interface for Temporal Annotations. In Enrique Amigó, Pablo Castells, Julio Gonzalo, Ben Carterette, J. Shane Culpepper, and Gabriella Kazai, editors, *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, pages 3289–3294. ACM, 2022a. doi: 10.1145/3477495.3531670

6.1 Background and Related Work

To achieve the goal of representing document contexts more accurately for a temporal retrieval setting, we first establish a formal view of "time" and suitable representations of various temporal mentions. We further briefly discuss approaches to extract temporal expressions from a text with the help of existing methods and their respective advantages and disadvantages. To ground the proposed solution in the frame of existing works, we further discuss approaches to timeline summarization as a subfield of aspect-based summarization.

6.1.1 Formalization of Date Representations

We rely on a formalization (and normalization) of temporal expressions into a unified representation. In particular, the format is in accordance with the ISO 8601 standard of a particular date, i.e. `YYYY-MM-DD` for a date representation. For simplicity, we ignore the extension for intra-day temporal units denoted by `HH:mm:ss`, but acknowledge that this is theoretically compatible with our further definitions.

We refer to any temporal mention in the ISO 8601 format as a *normalized* temporal expression. Schilder and Habel (2001) further divide temporal mentions into one of three categories:

1. **Explicit mentions**: These expressions directly refer to a particular date, e.g., "*October 16, 1998*". Explicit mentions can also refer to a coarser period, e.g. "*October 1998*", or "*1998*".
2. **Implicit mentions**: Expressions that still refer to an explicit date, but through implied meaning. Examples of that are references to public holidays, e.g., "*New Year's Eve 2002*".
3. **Relative mentions**: Not all expressions are relating to a fixed temporal point in time. Instead, relative mentions, such as "*yesterday*" can refer to different dates, depending on context. One temporal anchor could be a document creation date associated with a piece of text, or other, explicit temporal expressions in adjacent segments.

However, it should be noted that not all temporal expressions can be uniquely resolved into a distinct ISO 8601 representation. This can either be due to the lack of context (relative mentions remain ambiguous without a date anchor being available), or simply because the expression does not specify a particular date, but rather a set of possible dates. For example, the expression "*every other week*" refers to multiple points throughout time.

For simplicity, the rest of this chapter assumes that we deal with fully normalized temporal expressions stemming either from explicit mentions, or implicit/relative expressions that can be uniquely disambiguated into a normalized expression. This decision is made to simplify the later translation into query parameters for explicit temporal mentions during aspect retrieval.

To allow reasoning over a multiset of temporal expressions $\mathcal{T} = \{e^1, ..., e^m\}$, the following formalization allows a distinction at different granularity levels with respect to a linear temporal axis, or "timeline". We define each temporal mention $e \in \mathcal{T}$ by the smallest mentioned unit that is inferrable from the normalized date expression. For a temporal expression e, individual granularity levels can be referenced, such as e_Y for the year, e_M for month, and e_D for day associated with e, respectively. If an expression does not have any value for the particular granularity, it is considered undefined and cannot be compared at this level to any other expression. We call e an *incomplete* time stamp if any of the leading granularity levels are undefined. E.g., the phrase "*December 12*" by itself, yielding a normalized temporal expression $e =$"*XXXX-12-12*", would be considered incomplete. We still allow for dates that are only defined *up to* a certain granularity. As an example, "*March of 2024*" and its associated expression $e =$"*2024-03*" is still considered a valid date, even though it is only defined up to the granularity of months. Specifically, $e_Y = 2024$, $e_M = 03$, and $e_D =$*undefined*.[1]

We further define subsets of $\mathcal{T}$ depending on the granularity levels from ISO 8601 as well: $\mathcal{T}_Y$ contains all expressions explicitly mentioning years, $\mathcal{T}_M$ all expressions mentioning a month, and $\mathcal{T}_D$ all specific mentions of a day. Given our prior assumptions about the completeness of individual expressions in $\mathcal{T}$, it holds that $\mathcal{T}_D \subseteq \mathcal{T}_M \subseteq \mathcal{T}_Y = \mathcal{T}$.

For a pair of temporal expressions $e, f \in \mathcal{T}$, it holds that there exists a natural ordering between the expressions if both normalized representations have the same granularity available. We call e a "more recent event" if it holds that

$$e_Y > f_Y, \text{or} \tag{6.1}$$

$$e_Y = f_Y \land e_M > f_M, \text{or} \tag{6.2}$$

$$e_Y = f_Y \land e_M = f_M \land e_D > e_D. \tag{6.3}$$

In other words, e has to contain a larger value in the coarsest granularity that does not equal the value of f. By extension, it only holds that $e = f$ if values are defined to an equal granularity depth and equate all values. Formally, $e = f$, if and only if

$$e_Y = f_Y \land e_M = \text{undefined} \land f_M = \text{undefined, or} \tag{6.4}$$

$$e_Y = f_Y \land e_M = f_M \land e_D = \text{undefined} \land f_D = \text{undefined, or} \tag{6.5}$$

$$e_Y = f_Y \land e_M = f_M \land e_D = e_D. \tag{6.6}$$

While the equality of events is only possible if they both have a similar granularity available, there are still ways to express more relative temporal relations. When comparing a fine-grained date mention with a coarser representation, *containment* of dates refers to one temporal mention being within a larger duration. As an example, it is a truthful statement to say that "*2020-03-15*" happened during "*2020-03*", or during "*2020*". We use the subset predicate $\subset$ to express temporal containment of one event e within another coarser event f. Formally,

$$e \subset f \iff e_Y = f_Y \land f_M = \text{undefined} \land e_M \neq \text{undefined, or} \tag{6.7}$$

$$e \subset f \iff e_Y = f_Y \land e_M = f_M \land f_D = \text{undefined} \land e_D \neq \text{undefined.} \tag{6.8}$$

[1] The implication here is also that any date that has an explicitly defined *day* will always require to have a *month* mentioned. Any temporal expression violating this assumption would have to refer to a *set* of dates, which we previously excluded. As an example, consider "*the 15th of every month in the year of 2024*", which in itself expressly mentions (several) months.

With this model of temporal relations in mind, it is now also possible to explore the conversion of natural language text into a series of temporal expressions and its accurate representation for retrieval contexts.

6.1.2 Temporal Expression Recognition and Normalization

Turning a document into a series of temporally annotated segments further requires the automated extraction of temporal expressions from natural language text and storing said annotations in a suitable data format. Assume an extractor function $\Phi : d \rightarrow \mathcal{T}$ is given that can recognize and normalize mentions of temporal expressions in a document d.[2] In order to retain the association between a normalized temporal expression $e \in \mathcal{T}$ and its mention within d, one can further define an inverse mapping $\Phi' : \mathcal{T} \rightarrow d$.[3] This allows the recovery of every segment $t \in d$ where a temporal mention occurs that is equivalent to the normalized date in e. In practice, most systems implementing a variant of Φ and Φ' will frequently use an XML-like format over the text in d to store annotations directly within the text itself. See, for example, the popular TIMEX3 standard (Ferro et al., 2002; Pustejovsky et al., 2003). We previously presented an interface to unify the data export between different tools, improving the practical usability of systems in Aumiller et al. (2022a).

While our approach ultimately depends on the quality of upstream detection accuracy, we briefly highlight some of the available tools. Strötgen and Gertz (2010) introduce a rule-based extraction system called HeidelTime, which has later been extended to work with a variety of languages (Strötgen and Gertz, 2015). Another tried-and-tested method is SU-Time (Chang and Manning, 2013), which has since been incorporated in the Stanford CoreNLP framework (Manning et al., 2014). Given its better multilingual capabilities and flexible application to specific domains, we recommend HeidelTime as a flexible tagging approach in practical implementations (Aumiller et al., 2022a).

For the sake of completeness, it should be highlighted that recent works have explored the use of supervised neural approaches for temporal tagging and relation extraction (Laparra et al., 2018; Chen et al., 2019; Lange et al., 2020; Almasian et al., 2022a). We find that these approaches tend to scale poorly relative to the previously mentioned rule-based approaches and even more recent generative models seem to struggle with accurate information extraction from texts (Almasian et al., 2023).

[2] In practice, recognition and normalization of temporal expressions can be viewed as two separate problems, see Almasian et al. (2022a). For simplicity, assume systems can in fact perform both steps simultaneously.

[3] While we explicitly define functions in this section at the individual document level, we can similarly extend this notion for an entire document collection $\mathcal{D}$. We stick to the level of individual documents for now as it simplifies the disambiguation between segments from different documents.

6.1.3 Timeline Summarization

Previously there have been several attempts to phrase timeline summarization as an independent task setting with different focal points on the expected outcome. Compared to other summarization tasks, timeline summarization generally focuses on the creation of summaries from multiple documents (Steen and Markert, 2019; Gholipour Ghalandari and Ifrim, 2020; Hausner et al., 2020a,b; Yu et al., 2021; Ziegler et al., 2021; Campos et al., 2022). One can also observe a difference in the target modality, with approaches ranging from traditional textual outputs (Steen and Markert, 2019) to graph-based visualizations (Hausner et al., 2020a; Ziegler et al., 2021). Gholipour Ghalandari and Ifrim (2020) further highlight that existing approaches differ along several dimensions. While some works directly summarize contents from documents, others rely on an intermediate step of date-wise selection or event detection for summary generation. Similar intermediate steps can also be found in the methods used for clustering the input documents, an approach especially relevant when dealing with highly redundant content from web media (Piskorski et al., 2020; Ziegler et al., 2021). On the other hand, we also note approaches focusing on particular domain-specific timeline representations, such as court transcripts (Hausner et al., 2020b).

6.2 Hierarchical Temporal Representations

We already established that temporal expressions in texts can exist at different levels of granularity. However, it still needs to be defined how the mapping of segment temporal information can be transferred to document contexts. Realistically, not every segment may have an explicit temporal expression, but still provide relevant information for a particular temporal context. Consider the following segments at the sentence level and the temporal expressions within them.

The Berlin Wall fell on November 09, 1989. It was a notable event marking the fall of communism in Central Europe. However, it was not until October 03, 1990 that Germany officially reunited.

The first and last sentences in the given example contain an explicit temporal expression, "*1989-11-09*" and "*1990-10-03*", respectively. However, the sentence in between still carries relevant semantic meaning for the context of this document, and may be considered relevant for one or both of the adjacent sentences. Further problems arise if one instead considers all three sentences as a single segment, e.g., at the paragraph level. In such cases, further definitions are required for the priority between differing dates or decide on aggregation strategies. In the following sections, we present different ideas for both the propagation and aggregation of several expressions.

6.2.1 (Bi-)Directional Date Propagation

In a naive solution, it may be sufficient to associate temporal expressions only with the respective segments in which they occur. However, as previously mentioned, propagating the association of dates to adjacent segments of the actual occurrence of the underlying temporal expression can be beneficial to extend contextual knowledge in adjacent segments.

The first propagation scheme is *directional date propagation*, in which we associate the occurrence of expression with neighboring segments. Let $e \in \mathcal{T}$ be a temporal expression, with segment t_e referring to the segment $t_e \in d$, such that $\Phi'(e) = t_e$., We then refer to the directional date propagation $prop$ of e within d as the associated segment t_e, including its k right-sided neighboring segments. Formally,

$$prop(e, k) := [t_e, ..., t_{e+k}]. \tag{6.9}$$

In practice, suitable values for k depend on both granularity and intended propagation level. The intention here is that a left-to-right reading order may imply a one-sided context relevance, and preceding segments do not refer to the same temporal expression within the context of the document. Of course, the directionality may be adjusted for scripts that read right-to-left, such as Arabic.

In other contexts, it may be valid to extend the context in both directions, or what we refer to as *bidirectional date propagation*. Formally, we refer to the bidirectional propagation as $prop_{bi}$, with

$$prop_{bi}(e, k) := [t_{e-k}, ..., t_{e+k}]. \tag{6.10}$$

As an extension, one may consider a tapered relevance of temporal expressions for neighboring segments, depending on their distance. Similar weighting functions have previously been introduced in the context of textual co-occurrence weighting schemes (Spitz and Gertz, 2016; Almasian et al., 2022b), and even within for ranking temporal co-occurrences (Hausner et al., 2020b).

Compared to the previous assumption, where each segment $t_i \in prop(e, k)$ has a uniform relevance to the originating expression e, we now consider the segment distance in the form of an additional relevance weight for later rankings. Formally, let $id(t_i)$ be the index position of segment t_i within the document d. Considering a weighting function ω of a segment t_i relative to a temporal expression e is then defined as

$$\omega(e, t_i, k) := \begin{cases} \frac{1}{|id(t_i) - id(t_e)| + 1}, & \text{if } |id(t_i) - id(t_e)| < k \\ 0 & \text{else} \end{cases} \tag{6.11}$$

Similar weighting functions can also be defined with negative exponential weights, see, e.g., the previously mentioned work by Spitz and Gertz (2016). The specific limitation to the context window of k segments for the weights is primarily intended to optimize implementations of such weighting functions, where explicitly setting specific ranges to zero reduces the overall number of computations necessary.

6.2.2 Segment-level Aggregation of Expressions

While propagation of dates is helpful for instances where we have a *lack* of temporal expressions across segments, and helps to improve the coverage within a document, we often also face the contrary problem: segments containing multiple expressions at the same time. The likelihood to encounter such segments also increases with coarser granularity, e.g., when looking at paragraphs instead of individual sentences as a segment.

To this extent, we propose an aggregation scheme called that we refer to as *inclusive temporal range*. Instead of individually iterating over every expression $e \in \mathcal{T}$, we assume that it is more efficient to operate over a single segment-associated temporal range. Let $t_i \in d$ be a segment, then we consider the inclusive temporal range r of t_i as

$$r(t_i) := [\min_{e \in \mathcal{T}_i} e, \max_{e \in \mathcal{T}_i} e], \text{ with } \mathcal{T}_i := \{e \mid \Phi'(e) = t_i\}. \tag{6.12}$$

In other words, we consider the earliest and latest date mentions associated with the segment t_i as a sort of range delimiter for temporal relevance.[4] To give an example, the inclusive temporal range of the phrase "*He won the championship in 1987, 1988 and 1990.*" would be $r = [1987, 1990]$. Neatly, ranges are also still defined if only a single temporal expression is contained within t_i. For paragraphs without explicit mentions we consider the range to be undefined and the segment not relevant to a temporally sensitive ranking.

While the inclusive temporal range is a lossy approximation of the exact nature of the context, it allows for a more open-ended search relevance over temporal ranges, such as we expect them for aspect-based summaries. On the other hand, we can also briefly mention a lossless representation of mentioned dates if computational efficiency is not a concern. In such a case, the *date collection* over a segment t_i would simply be given by $\mathcal{T}_i$.

[4] Strictly speaking, extending by the propagated temporal mentions is also possible, making $\mathcal{T}_i$ compatible with our ideas from the previous section.

6.3 Intra-document Temporality as Ex-Ante Filters

From the given definition of segment-level annotation of documents it is now the obvious next step to consider the challenges of extending our ideas of a temporally annotated document to a fully fledged text summarizer. To illustrate some of the use cases for a more pronounced ex-ante filter, we further elaborate on the implementation of various temporal query operators, and notions of centrality for ranking competing sections revolving around the same date in Section 6.3.1.

For the purpose of efficient retrieval with the mentioned operators, we introduce a new index structure, called the Date Hierarchy Index (Section 6.3.2). It is a system not unlike more traditional inverted indices for Information Retrieval which would improve retrieval-focused query loads over a corpus with temporal annotations.

6.3.1 Temporal Coverage as Explicit Query Parameters

Inspired by Almasian et al. (2022b), we define a variety of temporal query operators based on argument (in)equality and range operations as explicit parameters for temporal retrieval in the context of an ex-ante filtering step. Additionally, we briefly discuss the use of other data structures, such as graphs, as means to compute content centrality (Hausner et al., 2020a).

Temporal Query Operators

We extend our temporal filtering criteria defined in Chapter 4 to a more granular level. This allows for expressing more distinct temporal constraints over the set of temporal expressions within a document.

Let $temp$ be a query operation, mapping a temporal query to a selection of relevant segments within a document d. With the definitions from Section 6.1.1, we already have a very explicit notion for the exact equality of two temporal expressions. Assuming an explicit query date q, we can now define the equality operation $=$ as

$$temp_{=}(q, d) := \{t_i |\, e = q \wedge \Phi'(e) = t_i\}. \tag{6.13}$$

Simply put, any segment containing an associated temporal expression matching the *exact same date* as the query date q will be relevant. We consider the resulting segments *without duplicates*, even though it could be the case that a date is associated multiple times with the same segment through differing temporal expressions $e, f \in \mathcal{T}.'$

We can similarly define relative query operations with respect to dates occurring prior to (<) or later than (>) the query date:

$$temp_{<}(q, d) := \{t_i |\, e < q \land \Phi'(e) = t_i\}, \text{ and} \tag{6.14}$$

$$temp_{>}(q, d) := \{t_i |\, e > q \land \Phi'(e) = t_i\}. \tag{6.15}$$

In accordance with Almasian et al. (2022b), we also define a temporal *range* operation, denoted by $[\cdot]$. For this operation, we are given two dates delimiting a temporal range $[q_1, q_2]$, and matching segments may simply fall within this range. Formally,

$$temp_{[\cdot]}(q_1, q_2, d) := \{t_i |\, q_1 <= e <= q_2 \land \Phi'(e) = t_i\} \tag{6.16}$$

Range expressions are particularly important to express search queries that match at different granularity levels. Assume, for example, a document d with associated extracted temporal expressions $\mathcal{T} = \{e = \textit{2022-05-10}, f = \textit{2023-10}\}$. As a given query, consider $q =$"*2022-05*", i.e., the entirety of May 2022. While we intuitively see that $e \subset q$, our definition of equality of temporal expression requires *all* granularity levels to be equivalent. As q_D (the day property of our query term) is technically undefined, this equality is not given. As such, $temp_{=}(q, d) = \{\}$.
Instead, we can express a temporal range query with similar meaning that would yield matches, as $temp_{[\cdot]}(\textit{2022-05-01}, \textit{2022-05-31}, d) = \{\Phi'(e)\}$. By explicitly giving the range at the day level, we can now match all expressions that fall within this range. By extension, this formalization would also allow the inclusion of coarser temporal expression, such as $e =$"*2022-05*", with ranges. For example, "*2022-04-30*" < "*2022-05*" < "*2022-06-01*".

Temporality in the Absence of Explicit Queries

Not all use cases would expect users to provide a temporal query. In the absence of such explicit queries, we may still want to incorporate temporal information as a separate aspect and return a ranking over segments within our input documents.
A naive approach is to simply order the segments in a ascending or descending order of temporal occurrence, and use this as a proxy for relevance. This may already be sufficiently helpful for news-related use cases, where recent events are likely more to be relevant (Campos et al., 2022).

We also hypothesize that a centrality-based ranking of "temporal significance" for particular dates could be used as a replacement when considering temporality as an ex-ante aspect. To name a simplistic approach, one could define a frequency-based occurrence centrality, such as a variant of TF-IDF, over only the temporal expressions (or their normalized representations). Even then, it has to be considered *which* of the segments relating to a frequently occurring date would be the most representative sample (a topic that we pick up again in the following section).

To allow implicit reasoning over different granularities, it is also possible to count each possible granularity variation of a single expression in occurrence-based centrality. To give an example, this could imply not just counting an expression in its original form, e.g. "*2024-04-06*", but also as the supergranular dates. In the example, this would be the month "*2024-04*" and year "*2024*". Counting the occurrence towards each of the respective dates would give a more comprehensive picture relative to the sparse nature of specific date occurrences. Finally, alternative centrality notions are also possible by considering co-occurrences graphs of temporal expressions within a text (Hausner et al., 2020a). This also opens up additional advantages of providing summary-like representations of event structures by traversing the graph structure (Spitz et al., 2019).

6.3.2 Date Hierarchy Index

In Information Retrieval, inverted indices are commonly used to map the occurrences of individual words or phrases back to the list of documents in which they occur. While the initial construction of an inverted index can be quite costly and requires constant memory to store, it dramatically reduces the access times for retrieval loads. The primary advantage is the reduction of lookup times from a full scan ($O(N)$, with N being the number of documents), to a $O(1)$ lookup in an inverted index. Similarly, we would like to optimize the representation of our temporal expression set $\mathcal{T}$, with specific consideration for the various query operations that might be expected. We again rely on similarities to Almasian et al. (2022b). However, where Almasian and colleagues assume that searching over quantities is inherently coupled with term retrieval, we make no such assumptions in our system for temporal retrieval. Importantly, leaving out such simplifications prevents us from cutting down on potential candidate segments by jointly filtering with the term relevance ranking.

Instead, we rely on the notion of sparsity in temporal events and argue for a more flexible architecture that utilizes the hierarchical nature of temporal expressions. This also brings us to our proposed indexing structure: a tree-based temporal index, loosely inspired by B-trees used in database management systems (Bayer and McCreight, 1972). Different levels within the tree refer to the temporal granularity of year, month, or day. An example of a date hierarchy index structure can be seen in Figure 6.1. Sorting node elements allows the efficient search over the tree, especially since the month and day layers (second-to-last and last layer, respectively) are of constant width. For months, there are at most 12 elements in each node–the number of months in a year. Whereas a node for the day level can have at most 31 elements, based on the number of days in a month. Any preceding layer in the graph may be utilized to efficiently represent the (theoretically unlimited) number of years. Specifically for leaf nodes, each leaf presents a list of segments relating to

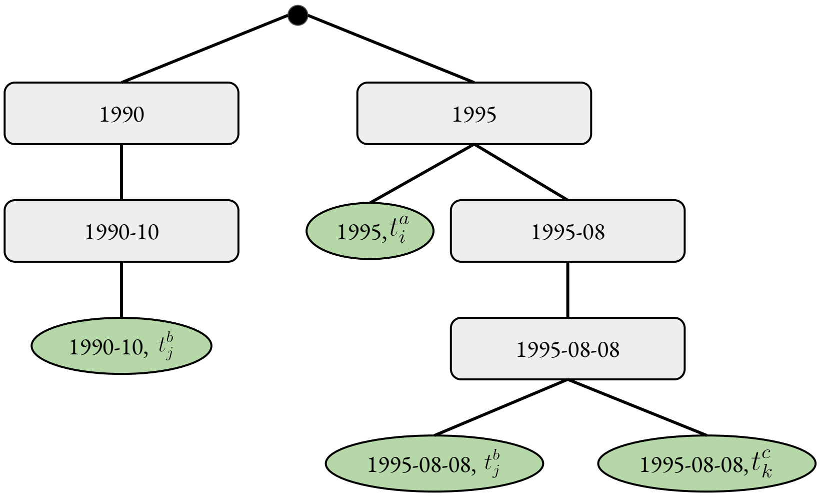


Figure 6.1: Visualization of a possible hierarchical index over a document collection with four distinct temporal mentions in underlying segments (green leaf nodes). The index is structured such that years, months and days are organized in a tree-like hierarchy.

its particular parent note. See, for example, the leaf node attached to the value "*1990*" in the year layer. As such, we can deal with incomplete month or year mentions, as well as specific temporal expressions referring to a single day as well.

We again highlight that in practice the tree structure will likely be relatively sparse, depending on the density of temporal mentions. Furthermore, we can also improve the storage size by not explicitly storing realized directionally propagated dates and traversing the implicit context window *in memory* instead of retrieving it directly from the index.[5] As previously mentioned, we want to highlight the shortcomings of such index structures. Insertion and deletion of documents within the index is relatively speaking more costly, and re-building the index with correctly sorted node elements is taking its time. As such, a hierarchical date index should only be considered for workloads where retrieval operations dominate.

6.4 Temporal Ordering as an Ex-Post Aspect

One of the practical problems we discuss in Hausner et al. (2020b) is the accurate depiction of temporal ordering within court records: the documents of a single proceeding reference various dates in an order different from a historical timeline. Meaning, earlier events are discussed much

[5] We not our previous work investigating similar co-occurrence-based context windows over graph representations in databases, where such an implicit context traversal lead to measurable improvements in memory consumption of the index structure (Spitz et al., 2020).

later in the court proceedings and vice versa, without any apparent ordering available. This is only one of several practical examples where temporality affects the generation of text, even if the proceedings are "manually" generated text.

However, similar nuances of temporality in automatically generated text still exist, and we briefly discuss instances of factors that can be considered almost as ex-post aspects according to our previously introduced framework in Chapter 4. For one, it may be a useful to retain a temporal ordering adhering to a timeline in the generation itself, and only sequentially generate based on segments referencing a particular window of time (Section 6.4.1). Related, we may also encounter dates that are frequently mentioned within a particular document, making many segments "relevant" for said date and competing for attention during the generation step.

6.4.1 Summaries following Timelines

While we previously suggested a timeline-centric ranking order in the summarization-related retrieval stage, we want to briefly elaborate how temporal information can influence the *generation of summaries* during the ex-post stage as well. In related work, Chan et al. (2024) specifically find that ChatGPT is struggling with the extraction of temporal relations within text, even after dedicated prompt engineering efforts by the authors. While the datasets used in their experiments rely on primarily implicit mentions of temporal expressions, it also relies only on the comparison of two events. We hypothesize that explicit knowledge of temporal mentions will improve the reasoning ability of LLMs in more complex contexts, even if their implicit temporal reasoning may not yet be satisfactory.

Xiong et al. (2024) note a similar trend with LLMs in general and propose the combination of a neuro-symbol engine to first disentangle event ordering in order to provide better answers in temporal QA settings. This is not unlike our proposed idea of ranking segments by their temporal order, although it specifically focuses on events and not exclusively on date mentions. They also focus on the secondary task of temporal reasoning, specifically for answering questions relating to event duration and ordering, instead of generating task-specific answers only.

For both works, the authors focus more on the ability to provide temporal ordering, and less so on the influence these rankings have on the generation step. Instead, they rely on models being able to follow generic instructions when reorganizing and aggregating the provided information with context clues. Specifically for graph-encoded knowledge, both teams also rely on subject-relation-object triplets to provide information in the prompt, which we believe to be relatively simplistic. As mentioned in Section 3.5, relation triplets may operate on lossy knowledge representation compared to the full underlying text from a segment. We therefore propose to simply input segments with the associated date (and relevance score) in the order of the ranking in a prompt template. Similar to our proposed "translation" of rankings into model-accessible representations, we suggest the use of structured prompt templates, such as JSON objects. A possible representation of the example at the beginning of Section 6.2 could be visualized as shown in Figure 6.2.

```
Use the information given in the following JSON object to create a coherent summary.
Retain the temporal ordering given by the field "date".
Use the relevance score to determine whether a segment should be included.
{
        {
                "text": "The Berlin Wall fell on November 09, 1989.",
                "date": "1989-11-09",
                "relevance_score": 1.0
        },
        {
                "text": [...],
                "date": [...],
                "relevance_score": [...]
        },
        [...]
}
```

Figure 6.2: Example prompt for representing segments individually annotated with explicit date mentions as defined in Section 6.2.

The main advantage of JSON objects is that it can also be extended by the other ex-ante ranking information, such as the `"relevance_score"` field shown in the example. Furthermore, the date field could be extended by a series of dates that occur within the segment, or have been propagated from neighboring segments. Explicitly instructing models to follow a sequential order with given dates seemed to work relatively well in a preliminary experiment with Cohere's Command R+ model.[6]

6.4.2 Cluster-based Temporal Summaries

As a similar problem to the previously mentioned "translation" of relevant information between ex-ante and ex-post stage, we also encounter issues when particular dates are over-represented in the temporal expressions extracted from d. This is not entirely unlikely to occur, especially when

[6] We experimented with 9 events given in random order, accessing the model through Cohere's web interface on April 09, 2024. With the exception of one date, all information was correctly ordered in the output.

documents address the events of a narrow time frame. In such scenarios, it is important to limit the amount of segments that will be provided for a single date. Otherwise, the sheer amount of data will negate any use of the previous ex-ante filtering, as there is effectively too much content that cannot be distinguished along our specified temporal ranking order. In such cases, it may be of additional help to employ intermediate steps to aggregate the content in relation to a particular date e, by using clustering techniques or some alternative iterative summarization approaches.

For context, works coupling external graph-based knowledge with abstractive summarization modules have been previously discussed in the context of temporal question answering (Gao et al., 2024) and frequently make use of clustering steps within their pipeline. While Martschat and Markert (2018) use k-means clustering to provide representative sentences, whereas Steen and Markert (2019) later improve this into a greedily selective cluster method that adds more sentences as long as similarity to existing selected sentences is not exceeding a threshold value. Other approaches also extend the use of cluster-based content selection to semantic clusters and event knowledge (Barros et al., 2019).

On the other hand, it may also be of interest for the user to investigate specific clusters in more detail, which is why it may be beneficial to explicitly pass all other segments in the ex-post context as well, but instruct models to only answer in their initial summary. For dialogue-oriented systems, follow-up questions can then still retrieve relevant event context from other segments that were not chosen as the cluster representative, without having to recompute an explicit ranking. However, if there are indeed specific (temporal) clusters that a user is interested in, this may be used as an input signal to generate explicit temporal queries that could be used for better filtering in the first place. As such, we leave it up to future work to determine a human-preferred solution of incorporating cluster information in temporal summaries.

6.5 Conclusion

This chapter has dealt with the extension of a very fundamental aspect that we previously only lightly touched: time. As such, we demonstrated that extensions of ex-ante aspects are non-trivial when considering the intricacies of a specific aspect, and sometimes can even influence the ex-post stage of generation.

More specifically, we present a model that allows to flexibly represent temporal expressions within a document, and separate the context into a more fine-grained representation for temporal filtering. It is also possible to extend this notion of temporal representations to a query language that allows the realization of temporal constraints on user preferences, whether through explicit query parameters or implicit centrality among date mentions. We further present a theoretical index structure to allow the efficient search over temporally annotated document collections, in order to allow for a practically feasible realization of our presented query system.

While it is an open problem to provide empirical evidence for the exact query and index model described here, our previous approaches to modeling temporal information with graph-based structures have already shown promising improvement in discovering temporal information within document collections (Hausner et al., 2020b,a). As such, we expect that explicit modeling of temporal information can further improve the quality of summaries, especially over large contexts. Furthermore, while existing temporal taggers already demonstrate reliable performance across many languages, we expect that training on synthetic temporal annotation data will eventually outperform rule-based approaches, even if we are currently not reaching comparable performance in non-English settings (Almasian et al., 2022a).

Of course, of special interest are also similar modeling approaches for other ex-ante aspects, such as spatial information (Sengstock et al., 2012), or quantity-focused retrieval (Almasian et al., 2022b). We imagine that similar aspect-specific document models can further improve the quality of downstream summaries, specifically for multi-aspect application scenarios. Examples may include the separate indexing of other numerical information, such as quantities (Almasian et al., 2022b), or the hierarchical representation of entity-related mentions.

7 Conclusion

"The most reliable way to predict the future is to create it."

Abraham Lincoln

Working towards a productive setting for text summarization systems is a critical need in today's information society, where an ever-increasing stream of data is tugging away at the focus of every human. The reality of summarization research looks inherently different, though: aside from a lack of diversity in domain-specific data collections, one can also observe a dissociation between user-specific needs regarding a summary and the focus points of newly introduced summarization systems. With this work, we have attempted a re-calibration of what truly matters for developing new systems: focusing on a user-centric model of text summarization, addressing each step in a summarization system pipeline, to ensure that outcomes are desirable – and practically useful. In this concluding chapter, we will briefly reiterate our contributions in Section 7.1, before finishing with a discussion of potential future work in Section 7.2.

7.1 Summary

Chapter 1 initially motivated the need for summarization systems as a means to accelerate the acquisition of knowledge and the reduction of mental burden on a variety of social groups: from students learning about new ideas to domain experts or knowledge workers trying to stay up-to-date on the latest news. Summaries can provide fast-tracked access to an exhaustive document collection, and greatly accelerate our interaction with existing information, if done correctly.

This section – coincidentally also named "summary" – is a good time to re-visit the motivating example of this thesis: considering on what information one can remember from reading this work. In all likelihood, re-reading the example summary given in Figure 1.1 may be a helpful reminder for some of the central concepts, which only goes to show the immense help summaries can provide, even after first interacting with a unknown document.

To truly understand the nature of practical requirements of summarization systems and the fair evaluation of them, we have spent considerable time building up the necessary foundations and relevant background work in Chapter 2. Aside from a formal model of text summarization we introduce the different variants of summarization systems, depending on the available input and output data. We further explore the various evaluation modalities for text summarization, which reveal one of the central difficulties in this particular task: it is a highly subjective setting, with ambiguous "ground truths" and a mentally straining annotation task, making it a challenging task to even gather relevant test data. Despite this, a flourishing research community has established itself over the past decades, proposing various training resources, model architectures, and automated evaluation metrics. As a conclusion to the motivating chapters, however, we already outline how different use cases for summarization systems have wildly differing requirements, many of which are not accurately represented in current directions of research.

Limitations of Existing Systems

To better understand what existing hurdles need to be tackled in newly proposed systems, Chapter 3 poses three central foundations on which successful summarization systems need to be developed: high-quality data, robust and extensible models, and reliable evaluation.

To alleviate some of the data-centric concerns, we propose several remedies in this work. As an instant solution to address problems in existing datasets, we demonstrate that some form of low-quality samples can be automatically filtered out, without further interference by humans. Part of our efforts also include proposing two completely new datasets for summarization. The first resource has been extracted from EU web platforms, providing a high-quality dataset for legal summarization, and is in parts made available by human translators in all 24 official EU languages.

In our exploration towards data-specific issues, we find that existing resources, particularly for German summarization, oftentimes fail basic quality checks. This includes, e.g., appropriate length proportions, or containing duplicates in their corpora. Similarly, we find that the majority of resources available focuses on a single domain, namely news-centric summarization. This focus on a single domain has immense consequences on the design of existing summarization models. Through the relatively short length of news articles, researchers are discouraged from developing models that are capable of dealing with longer context lengths. Unless specifically designed for domains that require such extensions, models generally do not support practically relevant document collections.

Finally, we are not the first to observe a lack of correlation between the truthfulness of a generated summary and high system scores on common automatic evaluation metrics. Our proposal is a novel metric to address this lack of consistency, and demonstrate that this metric is able to improve the automated analysis of existing summarization systems with respect to factuality.

An Aspect-based Model for Summarization

As a demonstration of how to incorporate more user-centric needs into summarization systems, we further propose a *hybrid and aspect-based architecture* for text summarization in Chapter 4. This includes a separation of concerns between a extractive first-stage retrieval module, which can be adjusted with a variety of aspects, depending on user preferences (or unsupervised extrapolations). Examples include filtering of entity- or query-restricted segments in an input text, or filtering by setting specific time ranges. As a second stage, we propose to use existing large language models as a guided paraphrasing module, which in turn incorporates further generation-specific user needs into the final summary. We specifically focus on the previously mentioned aspect of readability, as well as more generic styling approaches in generation.

The proposed architecture greatly benefits from the modular design, allowing for later changes to the system without the requirement of retraining from scratch. Simultaneously, we also benefit from the recent advancement in LLM research, which makes more powerful models (both open and closed source) available to the broader public. While such language models can already solve naive summarization tasks "by themselves", without the incorporation into our proposed framework, they remain constricted to a narrower context length and generic summary generation. Our preliminary evaluation results with a prototype implementation already reveals some of the possible extensions enabled by the plug-and-play model we present. We particularly note that there are distinct advantages over existing summarization approaches in how summaries can be controlled with respect to user preferences.

Model Extensions

Chapters 5 and 6 finally present possible extensions to our framework that focus in two very different aspects, respectively. Given the usage of an existing LLM in our prototype of Chapter 4, we lay out possibilities to train an aspect-based paraphraser to replace such a model. Notably, our findings of controllable generation agree with prior work, in that simple textual changes are noticeably more steerable (and correct) than larger, document-level rewriting phrases of documents. We illustrate these insights at the example of simplification-related tasks, where we have both document- and token-level resources available. While LLMs are currently used at an increasing rate to annotate silver-label data, we find that they are (as of now) unsuitable to improve the data availability for alignment-based tasks, such as paraphrasing.

On the other hand, our initial experiments in Chapter 4 also revealed a shortcoming in some of the retrieval-focused aspects. At the example of temporal metadata, we conceptualize a more elaborate annotation scheme. By going beyond simple document-level timestamps, and instead incorporating a hierarchical temporal ordering over temporal references *within the document text*, we are able to enable a more nuanced filtering of results at the sub-document level.

7.2 Future Work

While there are several obvious avenues open for future work, it is hard to predict the immediate next steps with respect to the rapid advancement in the field of Natural Language Processing. Even during the writing phase of this thesis, we have seen several iterations and innovations towards some of the central limitations discussed in this work. Similar to our discussion in Chapter 3, we will broadly address future work in the areas of modeling, data provenance, and evaluation of summarization systems.

End-to-End Modeling

Extensions of LLM context lengths open up more "direct" applications of models, and have become generally available by now with context lengths of more than 100,000 tokens being the norm for new releases. However, it is still open for discussion whether such extensions of context actually retain the same level of performance as a retrieval-augmented pipeline, such as particularly when operating with a diverse set of documents in a single reference collection. Related findings point out that critical aspects, such as factuality, may deteriorate over longer context windows (Tang et al., 2023). If this is the case for modern models, the utility of a modular aspect-focused architecture may still be highly relevant. Studying such nuanced behavior in more detail for summarization tasks would be immensely helpful for the community, and provide concrete evidence for the need of further improvements.

Should the results of such a study indicate the opposite, i.e., the direct integration of long-context models being equal or even superior to a two-stage architectures, then there are still several questions that need to be addressed. Firstly, the performance aspect may become more relevant, similar to current trends in the IR community, see, e.g., Lin et al. (2020) for a more detailed discussion of (dis-)advantages of fully neural approaches to ranking. Secondly, the problem of integrating document metadata into a LLM context remains challenging. Due to the positional bias encoded in the model architecture, it may be next to impossible to clearly indicate a separation between *multiple* inputs. Finally, in the end-to-end case, integration of different aspects from the ex-ante stage remain uncertain. While models may certainly be able to annotate and extract with a certain quality, dedicated taggers, for example for information extraction, remain superior to few-shot settings with LLMs for now (Almasian et al., 2023). Whether (and how) such extraction settings for ex-ante stages can be integrated into a single end-to-end model, remains yet to be seen and poses similarly interesting questions for future work.

Task-specific Multi-Aspect Datasets

Regarding datasets, we have already mentioned the lack of a *truly multi-aspect-focused dataset* in Chapter 4. Instead, we currently rely on the individual evaluation on single-aspect datasets, which decidedly limits the interpretability of results in our opinion. However, creating a (sizable) collection that provides such detailed annotations is currently not feasible for academic purposes, it seems. Regardless of these limitations, it may be possible to combine multiple different annotation sets on similar enough source data to create an artificial resource of what we imagine.

Furthermore, we have discussed a number of differing requirements, depending on the specific application domain. More recently, a number of datasets have been proposed for domain-specific summarization purposes (e.g., Goldsack et al. (2022) for medical use cases, or our own work on legal summarization (Aumiller et al., 2022b)). We believe that the aggregation of a more holistic evaluation suite across *summarization-specific datasets* could provide a helpful basis for future comparison of generic summarization systems, similar to the GLUE benchmark suite for NLU/NLI tasks (Wang et al., 2019). It needs to be ensured that proper evaluation metrics allow for a practical analysis of model performance, though. This could come in the form of specialized evaluation datasets, such as BUMP (Ma et al., 2023a), who propose a narrow dataset of minor modifications that cause large changes in factuality outcomes, or even more targeted datasets to provide a testbed for evaluation metrics themselves (Clark et al., 2023). Alternatively, small but high-quality datasets like SQuALITY (Wang et al., 2022a) with uniform instruction guidelines for annotators can also provide a better evaluation signal for testing systems.

Finally, a major milestone that is required for explainable truthful summaries, is the ability to back up generation snippets with relevant reference points in the original text (Lewis et al., 2020b). These alignments are furthermore useful to create more nuanced document annotations, required in focused applications such as legal or medical contexts. Even more generic "transfer task", such as simplification of snippets, suffer from a lack of funding to create non-English language resources (Ryan et al., 2023). Providing better alignment methods would greatly help with the curation of a wider range of datasets, especially focusing on different segment granularities, such as elementary discourse units or similar sub-sentence segmentations (Hewett and Stede, 2022).

Evaluation Metrics

Regarding evaluation metrics, we have exhaustively talked about the difficulty of proper experimental analysis. Similarly, with our argumentation regarding the more integrated modeling of users, it remains uncertain whether we will be able to *appropriately* model a system performance with automated evaluation metrics only. While LLM-based approaches to evaluation have certainly increased the overall interest in systems, they do not provide a one-size-fits-all remedy to evaluate summaries. We strongly suspect that such metrics similarly struggle to evaluate accurately on long-form or structured summarization tasks. Empirical studies of metrics on such settings would be of particular interest for the community, especially when moving to more generic systems. First recommendations have already been discussed, e.g., by Krishna et al. (2023).

Ultimately, we have spent considerable time attempting to answer the opening question: why do we not regularly interact with automatic text summarization systems? Through a series of limitations we have pointed out that existing approaches still suffer from shortcomings that often limit the flexibility, ultimately making them less productive for end users. Yet, we expect that summarization systems will continually improve and hopefully be more heavily integrated in business-centric applications in the near future. This opens up a whole different set of research challenges that need to be tackled from the point of evaluation. Safety- and ethics-compliant generation is already an emerging topic (Levy et al., 2022), but will also affect summarization-specific efforts. Particularly with respect to the aspect of factuality, we will see a reinforced effort towards compliant systems, particularly for sensitive domains such as medical applications. On the other hand, this may also allow (corporate) researchers to study the interaction patterns of actual users in more detail. Which in turn may lead to a more user-centric study setting in future work, hopefully streamlining evaluation protocols for user studies.

And this is the path forward that we envision as a central paradigm shift in the way automatic summaries are generated: with a focus on user utility, and not for the sake of a new model.

Bibliography

Samir Abdaljalil and Houda Bouamor. An Exploration of Automatic Text Summarization of Financial Reports. In *Proceedings of the Third Workshop on Financial Technology and Natural Language Processing*, pages 1–7, Online, 19 August 2021. URL https://aclanthology.org/2021.finnlp-1.1.

Gavin Abercrombie, Valerio Basile, Sara Tonelli, Verena Rieser, and Alexandra Uma, editors. *Proceedings of the 1st Workshop on Perspectivist Approaches to NLP @LREC2022*, Marseille, France, June 2022. European Language Resources Association. URL https://aclanthology.org/2022.nlperspectives-1.0.

Dmitrii Aksenov, Julián Moreno Schneider, Peter Bourgonje, Robert Schwarzenberg, Leonhard Hennig, and Georg Rehm. Abstractive Text Summarization based on Language Model Conditioning and Locality Modeling. In Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asunción Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020*, pages 6680–6689. European Language Resources Association, 2020. URL https://aclanthology.org/2020.lrec-1.825/.

Suha Al-Thanyyan and Aqil M. Azmi. Automated Text Simplification: A Survey. *ACM Computing Surveys*, 54(2):43:1–43:36, 2022. URL https://doi.org/10.1145/3442695.

Lamia Alam and Shane Mueller. Examining the effect of explanation on satisfaction and trust in AI diagnostic systems. *BMC Medical Informatics and Decision Making*, 21(1):178, 2021.

Garrett Allen, Ashlee Milton, Katherine Landau Wright, Jerry Alan Fails, Casey Kennington, and Maria Soledad Pera. Supercalifragilisticexpialidocious: Why Using the "Right" Readability Formula in Children's Web Search Matters. In Matthias Hagen, Suzan Verberne, Craig Macdonald, Christin Seifert, Krisztian Balog, Kjetil Nørvåg, and Vinay Setty, editors, *Advances in Information Retrieval - 44th European Conference on IR Research, ECIR 2022, Stavanger, Norway, April 10-14, 2022, Proceedings, Part I*, volume 13185 of *Lecture Notes in Computer Science*, pages 3–18. Springer, 2022. URL https://doi.org/10.1007/978-3-030-99736-6_1.

Garrett Allen, Katherine Landau Wright, Jerry Alan Fails, Casey Kennington, and Maria Soledad Pera. Multi-Perspective Learning to Rank to Support Children's Information Seeking in the Classroom. In *IEEE International Conference on Web Intelligence and Intelligent Agent Technology, WI-IAT 2023, Venice, Italy, October 26-29, 2023*, pages 311–317. IEEE, 2023. URL https://doi.org/10.1109/WI-IAT59888.2023.00050.

Satya Almasian, Dennis Aumiller, and Michael Gertz. Time for some German? Pre-Training a Transformer-based Temporal Tagger for German. In Ricardo Campos, Alípio Mário Jorge, Adam Jatowt, Sumit Bhatia, and Marina Litvak, editors, *Proceedings of Text2Story - Fifth Workshop on Narrative Extraction From Texts held in conjunction with the 44th European Conference on Information Retrieval (ECIR 2022), Stavanger, Norway, April 10, 2022*, volume 3117 of *CEUR Workshop Proceedings*, pages 83–90. CEUR-WS.org, 2022a. URL https://ceur-ws.org/Vol-3117/paper9.pdf.

Satya Almasian, Milena Bruseva, and Michael Gertz. QFinder: A Framework for Quantity-centric Ranking. In Enrique Amigó, Pablo Castells, Julio Gonzalo, Ben Carterette, J. Shane Culpepper, and Gabriella Kazai, editors, *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, pages 3272–3277. ACM, 2022b. URL https://doi.org/10.1145/3477495.3531672.

Satya Almasian, Vivian Kazakova, Philipp Göldner, and Michael Gertz. CQE: A Comprehensive Quantity Extractor. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12845–12859, Singapore, December 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.emnlp-main.793.

Omar Alonso, Michael Gertz, and Ricardo A. Baeza-Yates. On the value of temporal information in information retrieval. *SIGIR Forum*, 41(2):35–41, 2007. URL https://doi.org/10.1145/1328964.1328968.

Fernando Alva-Manchego, Louis Martin, Carolina Scarton, and Lucia Specia. EASSE: Easier Automatic Sentence Simplification Evaluation. In Sebastian Padó and Ruihong Huang, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, pages 49–54, Hong Kong, China, November 2019. Association for Computational Linguistics. URL https://aclanthology.org/D19-3009.

Toni Amstad. *Wie verständlich sind unsere Zeitungen?* Studenten-Schreib-Service, 1978.

Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Slav Petrov, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy P. Lillicrap, Angeliki Lazaridou, Orhan Firat, James Molloy, Michael Isard, Paul Ronald Barham, Tom Hennigan, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, Ryan Doherty, Eli Collins, Clemens Meyer, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, George Tucker, Enrique Piqueras, Maxim Krikun, Iain Barr, Nikolay Savinov, Ivo Danihelka, Becca Roelofs, Anaïs White, Anders Andreassen, Tamara von Glehn, Lakshman Yagati, Mehran Kazemi, Lucas Gonzalez, Misha Khalman, Jakub Sygnowski, and et al. Gemini: A Family of Highly Capable Multimodal Models. *CoRR*, abs/2312.11805, 2023. URL https://doi.org/10.48550/arXiv.2312.11805.

Dennis Aumiller and Michael Gertz. Klexikon: A German Dataset for Joint Summarization and Simplification. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2693–2701, Marseille, France, June 2022a. European Language Resources Association. URL https://aclanthology.org/2022.lrec-1.288.

Dennis Aumiller and Michael Gertz. UniHD at TSAR-2022 Shared Task: Is Compute All We Need for Lexical Simplification? In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 251–258, Abu Dhabi, United Arab Emirates (Virtual), December 2022b. Association for Computational Linguistics. URL https://aclanthology.org/2022.tsar-1.28.

Dennis Aumiller, Satya Almasian, Philip Hausner, and Michael Gertz. UniHD@CL-SciSumm 2020: Citation Extraction as Search. In *Proceedings of the First Workshop on Scholarly Document Processing*, pages 261–269, Online, November 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.sdp-1.29.

Dennis Aumiller, Satya Almasian, Sebastian Lackner, and Michael Gertz. Structural text segmentation of legal documents. In Juliano Maranhão and Adam Zachary Wyner, editors, *ICAIL '21: Eighteenth International Conference for Artificial Intelligence and Law, São Paulo Brazil, June 21 - 25, 2021*, pages 2–11. ACM, 2021. doi: 10.1145/3462757.3466085.

Dennis Aumiller, Satya Almasian, David Pohl, and Michael Gertz. Online DATEing: A Web Interface for Temporal Annotations. In Enrique Amigó, Pablo Castells, Julio Gonzalo, Ben Carterette, J. Shane Culpepper, and Gabriella Kazai, editors, *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, pages 3289–3294. ACM, 2022a. doi: 10.1145/3477495.3531670.

Dennis Aumiller, Ashish Chouhan, and Michael Gertz. EUR-Lex-Sum: A Multi- and Cross-lingual Dataset for Long-form Summarization in the Legal Domain. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7626–7639, Abu Dhabi, United Arab Emirates, December 2022b. Association for Computational Linguistics. URL https://aclanthology.org/2022.emnlp-main.519.

Dennis Aumiller, Jing Fan, and Michael Gertz. On the State of German (Abstractive) Text Summarization. In Birgitta König-Ries, Stefanie Scherzinger, Wolfgang Lehner, and Gottfried Vossen, editors, *Datenbanksysteme für Business, Technologie und Web (BTW 2023), 20. Fachtagung des GI-Fachbereichs „Datenbanken und Informationssysteme" (DBIS), 06.-10, März 2023, Dresden, Germany, Proceedings*, volume P-331 of *LNI*, pages 195–220. Gesellschaft für Informatik e.V., 2023. doi: 10.18420/BTW2023-10.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Yu Bai, Yang Gao, and Heyan Huang. Cross-Lingual Abstractive Summarization with Limited Parallel Resources. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*

(Volume 1: Long Papers), pages 6910–6924, Online, August 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.acl-long.538.

Satanjeev Banerjee and Alon Lavie. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. URL https://aclanthology.org/W05-0909.

Elena Baralis, Luca Cagliero, Naeem A. Mahoto, and Alessandro Fiori. GraphSum: Discovering correlations among multiple terms for graph-based summarization. *Information Sciences*, 249: 96–109, 2013. URL https://doi.org/10.1016/j.ins.2013.06.046.

Cristina Barros, Elena Lloret, Estela Saquete, and Borja Navarro-Colorado. NATSUM: narrative abstractive summarization through cross-document timeline generation. *Information Processing & Management*, 56(5):1775–1793, 2019. URL https://doi.org/10.1016/j.ipm.2019.02.010.

Regina Barzilay and Noemie Elhadad. Sentence Alignment for Monolingual Comparable Corpora. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 25–32, 2003. URL https://www.aclweb.org/anthology/W03-1004.

Elizabeth Bates and Brian Macwhinney. Functionalist Approaches to Grammar. *Language Acquisition: The State of the Art*, pages 173–218, 01 1982.

Alessia Battisti, Dominik Pfütze, Andreas Säuberli, Marek Kostrzewa, and Sarah Ebling. A Corpus for Automatic Readability Assessment and Text Simplification of German. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3302–3311, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL https://aclanthology.org/2020.lrec-1.404.

Rudolf Bayer and Edward M. McCreight. Organization and Maintenance of Large Ordered Indices. *Acta Informatica*, 1:173–189, 1972. URL https://doi.org/10.1007/BF00288683.

Iz Beltagy, Kyle Lo, and Arman Cohan. SciBERT: A Pretrained Language Model for Scientific Text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China, November 2019. Association for Computational Linguistics. URL https://aclanthology.org/D19-1371.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The Long-Document Transformer. *CoRR*, abs/2004.05150, 2020. URL https://arxiv.org/abs/2004.05150.

Anya Belz, Craig Thomson, Ehud Reiter, Gavin Abercrombie, Jose M. Alonso-Moral, Mohammad Arvan, Anouck Braggaar, Mark Cieliebak, Elizabeth Clark, Kees van Deemter, Tanvi Dinkar, Ondřej Dušek, Steffen Eger, Qixiang Fang, Mingqi Gao, Albert Gatt, Dimitra Gkatzia, Javier González-Corbelle, Dirk Hovy, Manuela Hürlimann, Takumi Ito, John D. Kelleher, Filip Klubicka, Emiel Krahmer, Huiyuan Lai, Chris van der Lee, Yiru Li, Saad Mahamood, Margot Mieskes, Emiel van Miltenburg, Pablo Mosteiro, Malvina Nissim, Natalie Parde, Ondřej Plátek, Verena Rieser, Jie Ruan, Joel Tetreault, Antonio Toral, Xiaojun Wan, Leo Wanner, Lewis Watson, and Diyi Yang. Missing Information, Unresponsive Authors, Experimental Flaws: The Impossibility of Assessing the Reproducibility of Previous Human Evaluations in NLP. In Shabnam Tafreshi, Arjun Akula, João Sedoc, Aleksandr Drozd, Anna Rogers, and Anna Rumshisky, editors, *Proceedings of the Fourth Workshop on Insights from Negative Results in NLP*, pages 1–10, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.insights-1.1.

Darina Benikova, Margot Mieskes, Christian M. Meyer, and Iryna Gurevych. Bridging the gap between extractive and abstractive summaries: Creation and evaluation of coherent extracts from heterogeneous sources. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1039–1050, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee. URL https://www.aclweb.org/anthology/C16-1099.

Hélène Bernet and Pascale Berteloot. EUR-Lex: A multilingual on-line website for European Union law. *International Review of Law Computers & Technology*, 20(3):337–339, 2006.

Vijay K. Bhatia. Simplification v. Easification – The Case of Legal Texts. *Applied Linguistics*, 4 (1):42–54, 1983.

Paheli Bhattacharya, Kaustubh Hiware, Subham Rajgaria, Nilay Pochhi, Kripabandhu Ghosh, and Saptarshi Ghosh. A Comparative Study of Summarization Algorithms Applied to Legal Case Judgments. In *Advances in Information Retrieval - 41st European Conference on IR Research, ECIR 2019, Cologne, Germany, April 14-18, 2019, Proceedings, Part I*, volume 11437 of *Lecture Notes in Computer Science*, pages 413–428. Springer, 2019.

David Biesner, Eduardo Brito, Lars Patrick Hillebrand, and Rafet Sifa. Hybrid Ensemble Predictor as Quality Metric for German Text Summarization: Fraunhofer IAIS at GermEval 2020 Task 3. In Sarah Ebling, Don Tuggener, Manuela Hürlimann, Mark Cieliebak, and Martin Volk, editors, *Proceedings of the 5th Swiss Text Analytics Conference and the 16th Conference on Natural Language Processing, SwissText/KONVENS 2020, Zurich, Switzerland, June 23-25, 2020*, volume 2624 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2020. URL http://ceur-ws.org/Vol-2624/germeval-task3-paper3.pdf.

Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media, Inc., 2009.

Shoshana Blum and Eddie A. Levenston. Universals of lexical simplification. *Language Learning*, 28(2):399–415, 1978.

Rishi Bommasani and Claire Cardie. Intrinsic Evaluation of Summarization Datasets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8075–8096, Online, November 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.emnlp-main.649.

Phillip Bonacich. Power and Centrality: A Family of Measures. *American Journal of Sociology*, 92(5):1170–1182, 1987.

Claire Bonial, Jena Hwang, Julia Bonn, Kathryn Conger, Olga Babko-Malaya, and Martha Palmer. English PropBank Annotation Guidelines. *Center for Computational Language and Education Research Institute of Cognitive Science University of Colorado at Boulder*, 48, 2012.

Stevo Bozinovski. Reminder of the First Paper on Transfer Learning in Neural Networks, 1976. *Informatica (Slovenia)*, 44(3), 2020. URL https://doi.org/10.31449/inf.v44i3.2828.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html.

Isabel Cachola, Kyle Lo, Arman Cohan, and Daniel Weld. TLDR: Extreme Summarization of Scientific Documents. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4766–4777, Online, November 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.findings-emnlp.428.

Ricardo Campos, Gaël Dias, Alípio Mário Jorge, and Adam Jatowt. Survey of Temporal Information Retrieval and Related Applications. *ACM Computing Surveys*, 47(2):15:1–15:41, 2014. URL https://doi.org/10.1145/2619088.

Vasco Campos, Ricardo Campos, Pedro Mota, and Alípio Jorge. Tweet2Story: A Web App to Extract Narratives from Twitter. In Matthias Hagen, Suzan Verberne, Craig Macdonald, Christin Seifert, Krisztian Balog, Kjetil Nørvåg, and Vinay Setty, editors, *Advances in Information Retrieval - 44th European Conference on IR Research, ECIR 2022, Stavanger, Norway, April 10-14, 2022, Proceedings, Part II*, volume 13186 of *Lecture Notes in Computer Science*, pages 270–275. Springer, 2022. URL https://doi.org/10.1007/978-3-030-99739-7_32.

Ziqiang Cao, Wenjie Li, Sujian Li, and Furu Wei. Retrieve, Rerank and Rewrite: Soft Template Based Neural Summarization. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 152–161, Melbourne, Australia, July 2018. Association for Computational Linguistics. URL https://aclanthology.org/P18-1015.

Jaime Carbonell, Donna Harman, Eduard Hovy, Steve Maiorano, John Prange, and Karen Sparck-Jones. Vision Statement to Guide Research in Question & Answering (Q&A) and

Text Summarization. *Rapport Technique, NIST*, 2000. URL https://www-nlpir.nist.gov/projects/duc/papers/Final-Vision-Paper-v1a.pdf.

Ilias Chalkidis, Emmanouil Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. Extreme Multi-Label Legal Text Classification: A Case Study in EU Legislation. In *Proceedings of the Natural Legal Language Processing Workshop 2019*, pages 78–87, Minneapolis, Minnesota, June 2019a. Association for Computational Linguistics. URL https://aclanthology.org/W19-2209.

Ilias Chalkidis, Emmanouil Fergadiotis, Prodromos Malakasiotis, and Ion Androutsopoulos. Large-Scale Multi-Label Text Classification on EU Legislation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6314–6322, Florence, Italy, July 2019b. Association for Computational Linguistics. URL https://aclanthology.org/P19-1636.

Ilias Chalkidis, Manos Fergadiotis, and Ion Androutsopoulos. MultiEURLEX - A multi-lingual and multi-label legal document classification dataset for zero-shot cross-lingual transfer. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6974–6996, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.emnlp-main.559.

Jeanne Sternlicht Chall. *Readability: An appraisal of research and application*. Number 34. Ohio State University, 1958.

Branden Chan, Stefan Schweter, and Timo Möller. German's Next Language Model. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics. URL https://aclanthology.org/2020.coling-main.598.

Chunkit Chan, Cheng Jiayang, Weiqi Wang, Yuxin Jiang, Tianqing Fang, Xin Liu, and Yangqiu Song. Exploring the Potential of ChatGPT on Sentence Level Relations: A Focus on Temporal, Causal, and Discourse Relations. In Yvette Graham and Matthew Purver, editors, *Findings of the Association for Computational Linguistics: EACL 2024*, pages 684–721, St. Julian's, Malta, March 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.findings-eacl.47.

Hou Pong Chan, Lu Wang, and Irwin King. Controllable Summarization with Constrained Markov Decision Process. *Transactions of the Association for Computational Linguistics*, 9: 1213–1232, 2021. doi: 10.1162/tacl_a_00423. URL https://aclanthology.org/2021.tacl-1.72.

Muthu Kumar Chandrasekaran, Guy Feigenblat, Eduard Hovy, Abhilasha Ravichander, Michal Shmueli-Scheuer, and Anita de Waard. Overview and Insights from the Shared Tasks at Scholarly Document Processing 2020: CL-SciSumm, LaySumm and LongSumm. In *Proceedings of the First Workshop on Scholarly Document Processing*, pages 214–224, Online, November 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.sdp-1.24.

Angel Chang and Christopher D. Manning. SUTime: Evaluation in TempEval-3. In Suresh Manandhar and Deniz Yuret, editors, *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 78–82, Atlanta, Georgia, USA, June 2013. Association for Computational Linguistics. URL `https://aclanthology.org/S13-2013`.

Niladri Chatterjee and Raksha Agarwal. Studying the Effect of Syntactic Simplification on Text Summarization. *IETE Technical Review*, 40(2):155–166, 2023. doi: 10.1080/02564602.2022.2055670.

Chung-Chi Chen, Hen-Hsen Huang, Hiroya Takamura, and Hsin-Hsi Chen, editors. *Proceedings of the Third Workshop on Financial Technology and Natural Language Processing*, Online, 19 August 2021. -. URL `https://aclanthology.org/2021.finnlp-1.0`.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading Wikipedia to Answer Open-Domain Questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada, July 2017. Association for Computational Linguistics. URL `https://aclanthology.org/P17-1171`.

Sanxing Chen, Guoxin Wang, and Börje Karlsson. Exploring Word Representations on Time Expression Recognition. Technical report, Technical Report, Microsoft Research Asia, 2019.

Paul F. Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep Reinforcement Learning from Human Preferences. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL `https://proceedings.neurips.cc/paper_files/paper/2017/file/d5e2c0adad503c91f91df240d0cd4e49-Paper.pdf`.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. Scaling Instruction-Finetuned Language Models. *CoRR*, abs/2210.11416, 2022. URL `https://doi.org/10.48550/arXiv.2210.11416`.

Elizabeth Clark, Shruti Rijhwani, Sebastian Gehrmann, Joshua Maynez, Roee Aharoni, Vitaly Nikolaev, Thibault Sellam, Aditya Siddhant, Dipanjan Das, and Ankur Parikh. SEAHORSE: A Multilingual, Multifaceted Dataset for Summarization Evaluation. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9397–9413, Singapore, December 2023. Association for Computational Linguistics. URL `https://aclanthology.org/2023.emnlp-main.584`.

Orphée De Clercq, Véronique Hoste, Bart Desmet, Philip van Oosten, Martine De Cock, and Lieve Macken. Using the crowd for readability prediction. *Natural Language Engineering*, 20 (3):293–325, 2014. URL `https://doi.org/10.1017/S1351324912000344`.

Maximin Coavoux, Hady Elsahar, and Matthias Gallé. Unsupervised Aspect-Based Multi-Document Abstractive Summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 42–47, Hong Kong, China, November 2019. Association for Computational Linguistics. URL `https://aclanthology.org/D19-5405`.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. A Discourse-Aware Attention Model for Abstractive Summarization of Long Documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. URL `https://aclanthology.org/N18-2097`.

Jack G. Conrad, Jochen L. Leidner, Frank Schilder, and Ravi Kondadadi. Query-based Opinion Summarization for Legal Blog Entries. In *The 12th International Conference on Artificial Intelligence and Law, Proceedings of the Conference, June 8-12, 2009, Barcelona, Spain*, pages 167–176. ACM, 2009. URL `https://doi.org/10.1145/1568234.1568253`.

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms, 3rd Edition*. MIT Press, 2009. ISBN 978-0-262-03384-8. URL `http://mitpress.mit.edu/books/introduction-algorithms`.

William Coster and David Kauchak. Simple English Wikipedia: A New Text Simplification Task. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 665–669, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/P11-2117`.

Liam Cripwell, Joël Legrand, and Claire Gardent. Document-Level Planning for Text Simplification. In Andreas Vlachos and Isabelle Augenstein, editors, *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 993–1006, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. URL `https://aclanthology.org/2023.eacl-main.70`.

James T. Croasmun and Lee Ostrom. Using Likert-Type Scales in the Social Sciences. *Journal of Adult Education*, 40(1):19–22, 2011.

Hoa Trang Dang. Overview of DUC 2005. In *Proceedings of the Document Understanding Conference*, volume 2005, pages 1–12. Citeseer, 2005.

Hoa Trang Dang. Overview of DUC 2006. In *Proceedings of the Document Understanding Conference*, volume 2006, 2006. URL `https://duc.nist.gov/pubs/2006papers/duc2006.pdf`.

Hoa Trang Dang and Karolina Owczarzak. Overview of the TAC 2008 Opinion Question Answering and Summarization Tasks. In *Proceedings of the First Text Analysis Conference*, volume 2, 2008.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and Memory-Efficient Exact Attention with IO-Awareness. In Sanmi

Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/67d57c32e20fd0a7a302cb81d36e40d5-Abstract-Conference.html.

Koen Dercksen, Arjen P. de Vries, and Bram van Ginneken. SimpleRad: Patient-Friendly Dutch Radiology Reports. In Jaap Kamps, Lorraine Goeuriot, Fabio Crestani, Maria Maistro, Hideo Joho, Brian Davis, Cathal Gurrin, Udo Kruschwitz, and Annalina Caputo, editors, *Advances in Information Retrieval - 45th European Conference on Information Retrieval, ECIR 2023, Dublin, Ireland, April 2-6, 2023, Proceedings, Part III*, volume 13982 of *Lecture Notes in Computer Science*, pages 224–229. Springer, 2023. doi: 10.1007/978-3-031-28241-6_18. URL https://doi.org/10.1007/978-3-031-28241-6_18.

Daniel Deutsch, Tania Bedrax-Weiss, and Dan Roth. Towards Question-Answering as an Automatic Metric for Evaluating the Content Quality of a Summary. *Transactions of the Association for Computational Linguistics*, 9:774–789, 2021a. doi: 10.1162/tacl_a_00397. URL https://aclanthology.org/2021.tacl-1.47.

Daniel Deutsch, Rotem Dror, and Dan Roth. A Statistical Analysis of Summarization Evaluation Metrics Using Resampling Methods. *Transactions of the Association for Computational Linguistics*, 9:1132–1146, 2021b. doi: 10.1162/tacl_a_00417. URL https://aclanthology.org/2021.tacl-1.67.

Daniel Deutsch, Rotem Dror, and Dan Roth. Re-Examining System-Level Correlations of Automatic Summarization Evaluation Metrics. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 6038–6052, Seattle, United States, July 2022a. Association for Computational Linguistics. URL https://aclanthology.org/2022.naacl-main.442.

Daniel Deutsch, Rotem Dror, and Dan Roth. On the Limitations of Reference-Free Evaluations of Generated Text. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10960–10977, Abu Dhabi, United Arab Emirates, December 2022b. Association for Computational Linguistics. URL https://aclanthology.org/2022.emnlp-main.753.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019. doi: 10.18653/v1/n19-1423. URL https://doi.org/10.18653/v1/n19-1423.

Zi-Yi Dou, Pengfei Liu, Hiroaki Hayashi, Zhengbao Jiang, and Graham Neubig. GSum: A General Framework for Guided Neural Abstractive Summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human*

Language Technologies, pages 4830–4842, Online, June 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.naacl-main.384.

Olive J. Dunn. Multiple Comparisons Among Means. *Journal of the American Statistical Association*, 56(293):52–64, 1961.

Esin Durmus, He He, and Mona Diab. FEQA: A Question Answering Evaluation Framework for Faithfulness Assessment in Abstractive Summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5055–5070, Online, July 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.acl-main.454.

Alberto Díaz and Pablo Gervás. User-model based personalized summarization. *Information Processing & Management*, 43(6):1715–1734, 2007. ISSN 0306-4573. URL https://www.sciencedirect.com/science/article/pii/S0306457307000386. Text Summarization.

Harold P. Edmundson. Problems in Automatic Abstracting. *Communications of the ACM*, 7(4): 259–263, apr 1964. ISSN 0001-0782. URL https://doi.org/10.1145/364005.364088.

Harold P. Edmundson and R. E. Wyllys. Automatic Abstracting and Indexing–Survey and Recommendations. *Communications of the ACM*, 4(5):226–234, may 1961. ISSN 0001-0782. URL https://doi.org/10.1145/366532.366545.

Mahmoud El-Haj, Ahmed AbuRa'ed, Marina Litvak, Nikiforos Pittaras, and George Giannakopoulos. The Financial Narrative Summarisation Shared Task (FNS 2020). In *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*, pages 1–12, Barcelona, Spain (Online), December 2020. COLING. URL https://aclanthology.org/2020.fnp-1.1.

Mahmoud El-Haj, Paul Rayson, and Nadhem Zmandar, editors. *Proceedings of the 4th Financial Narrative Processing Workshop @LREC2022*, Marseille, France, June 2022a. European Language Resources Association. URL https://aclanthology.org/2022.fnp-1.0.

Mahmoud El-Haj, Nadhem Zmandar, Paul Rayson, Ahmed AbuRa'ed, Marina Litvak, Nikiforos Pittaras, George Giannakopoulos, Aris Kosmopoulos, Blanca Carbajo-Coronado, and Antonio Moreno-Sandoval. The Financial Narrative Summarisation Shared Task (FNS 2022). In *Proceedings of the 4th Financial Narrative Processing Workshop @LREC2022*, pages 43–52, Marseille, France, June 2022b. European Language Resources Association. URL https://aclanthology.org/2022.fnp-1.6.

Mohamed Elaraby and Diane Litman. ArgLegalSumm: Improving Abstractive Summarization of Legal Documents with Argument Mining. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6187–6194, Gyeongju, Republic of Korea, October 2022. International Committee on Computational Linguistics. URL https://aclanthology.org/2022.coling-1.540.

Brigitte Endres-Niggermeyer, Jerry Hobbs, and Karen Sparck Jones. Summarizing Text for Intelligent Communication (Dagstuhl Seminar 9350). Dagstuhl Seminar Report 79, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 1995. URL https://drops.dagstuhl.de/opus/volltexte/2021/14967.

Güneş Erkan and Dragomir R. Radev. LexPageRank: Prestige in Multi-Document Text Summarization. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 365–371, Barcelona, Spain, July 2004a. Association for Computational Linguistics. URL https://aclanthology.org/W04-3247.

Günes Erkan and Dragomir R. Radev. LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. *Journal of Artificial Intelligence Research*, 22:457–479, 2004b. doi: 10.1613/jair.1523.

Liana Ermakova, Jean Valère Cossu, and Josiane Mothe. A survey on evaluation of summarization methods. *Information Processing & Management*, 56(5):1794–1814, 2019.

Liana Ermakova, Eric SanJuan, Jaap Kamps, Stéphane Huet, Irina Ovchinnikova, Diana Nurbakova, Sílvia Araújo, Radia Hannachi, Élise Mathurin, and Patrice Bellot. Overview of the CLEF 2022 SimpleText Lab: Automatic Simplification of Scientific Texts. In Alberto Barrón-Cedeño, Giovanni Da San Martino, Mirko Degli Esposti, Fabrizio Sebastiani, Craig Macdonald, Gabriella Pasi, Allan Hanbury, Martin Potthast, Guglielmo Faggioli, and Nicola Ferro, editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction - 13th International Conference of the CLEF Association, CLEF 2022, Bologna, Italy, September 5-8, 2022, Proceedings*, volume 13390 of *Lecture Notes in Computer Science*, pages 470–494. Springer, 2022. URL https://doi.org/10.1007/978-3-031-13643-6_28.

Shahul Es, Jithin James, Luis Espinosa Anke, and Steven Schockaert. RAGAs: Automated Evaluation of Retrieval Augmented Generation. In Nikolaos Aletras and Orphee De Clercq, editors, *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 150–158, St. Julians, Malta, March 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.eacl-demo.16.

Alexander R. Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. SummEval: Re-evaluating Summarization Evaluation. *Transactions of the Association for Computational Linguistics*, 9:391–409, 2021. doi: 10.1162/tacl_a_00373. URL https://aclanthology.org/2021.tacl-1.24.

Tobias Falke, Leonardo F. R. Ribeiro, Prasetya Ajie Utama, Ido Dagan, and Iryna Gurevych. Ranking Generated Summaries by Correctness: An Interesting but Challenging Application for Natural Language Inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2214–2220, Florence, Italy, July 2019. Association for Computational Linguistics. URL https://aclanthology.org/P19-1213.

Angela Fan, David Grangier, and Michael Auli. Controllable Abstractive Summarization. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 45–

54, Melbourne, Australia, July 2018. Association for Computational Linguistics. URL https://aclanthology.org/W18-2706.

Jing Fan, Dennis Aumiller, and Michael Gertz. Evaluating Factual Consistency of Texts with Semantic Role Labeling. In Alexis Palmer and Jose Camacho-collados, editors, *Proceedings of the 12th Joint Conference on Lexical and Computational Semantics (*SEM 2023)*, pages 89–100, Toronto, Canada, July 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.starsem-1.9.

Mehwish Fatima and Michael Strube. A Novel Wikipedia based Dataset for Monolingual and Cross-Lingual Summarization. In *Proceedings of the Third Workshop on New Frontiers in Summarization*, pages 39–50, Online and in Dominican Republic, November 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.newsum-1.5.

Pascal Fecht, Sebastian Blank, and Hans-Peter Zorn. Sequential Transfer Learning in NLP for German Text Summarization. In Mark Cieliebak, Don Tuggener, and Fernando Benites, editors, *Proceedings of the 4th Swiss Text Analytics Conference, SwissText 2019, Winterthur, Switzerland, June 18-19, 2019*, volume 2458 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2019. URL http://ceur-ws.org/Vol-2458/paper8.pdf.

Yutao Feng, Jipeng Qiang, Yun Li, Yunhao Yuan, and Yi Zhu. Sentence Simplification via Large Language Models. *CoRR*, abs/2302.11957, 2023. URL https://doi.org/10.48550/arXiv.2302.11957.

Lisa Ferro, Robyn Kozierok, Laurie Gerber, Beth Sundheim, Inderjeet Mani, and George Wilson. Annotating temporal information: from theory to practice. In *Proceedings of the second international conference on Human Language Technology Research*, pages 226–230, 2002.

Katja Filippova, Mihai Surdeanu, Massimiliano Ciaramita, and Hugo Zaragoza. Company-Oriented Extractive Summarization of Financial News. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 246–254, Athens, Greece, March 2009. Association for Computational Linguistics. URL https://aclanthology.org/E09-1029.

Tim Fischer, Steffen Remus, and Chris Biemann. Measuring Faithfulness of Abstractive Summaries. In *Proceedings of the 18th Conference on Natural Language Processing (KONVENS 2022)*, pages 63–73, Potsdam, Germany, 12–15 September 2022. KONVENS 2022 Organizers. URL https://aclanthology.org/2022.konvens-1.8.

Rudolph Flesch. A new readability yardstick. *Journal of Applied Psychology*, 32(3):221–233, 1948. URL https://doi.org/10.1037/h0057532.

Dominik Frefel. Summarization Corpora of Wikipedia Articles. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6651–6655, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL https://www.aclweb.org/anthology/2020.lrec-1.821.

Dominik Frefel, Manfred Vogel, and Fabian Märki. 2nd german text summarization challenge. In Sarah Ebling, Don Tuggener, Manuela Hürlimann, Mark Cieliebak, and Martin Volk, editors, *Proceedings of the 5th Swiss Text Analytics Conference and the 16th Conference on Natural Language Processing, SwissText/KONVENS 2020, Zurich, Switzerland, June 23-25, 2020 [online only]*, volume 2624 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2020. URL http://ceur-ws.org/Vol-2624/germeval-task3-paper1.pdf.

Markus Freitag and Yaser Al-Onaizan. Beam Search Strategies for Neural Machine Translation. In Thang Luong, Alexandra Birch, Graham Neubig, and Andrew Finch, editors, *Proceedings of the First Workshop on Neural Machine Translation*, pages 56–60, Vancouver, August 2017. Association for Computational Linguistics. URL https://aclanthology.org/W17-3207.

Lea Frermann and Alexandre Klementiev. Inducing Document Structure for Aspect-based Summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6263–6273, Florence, Italy, July 2019. Association for Computational Linguistics. URL https://aclanthology.org/P19-1630.

Danilo Fum, Giovanni Guida, and Carlo Tasso. Forward and Backward Reasoning in Automatic Abstracting. In *Coling 1982: Proceedings of the Ninth International Conference on Computational Linguistics*, 1982. URL https://aclanthology.org/C82-1013.

Yifu Gao, Linbo Qiao, Zhigang Kan, Zhihua Wen, Yongquan He, and Dongsheng Li. Two-stage Generative Question Answering on Temporal Knowledge Graph Using Large Language Models. *CoRR*, abs/2402.16568, 2024. URL https://doi.org/10.48550/arXiv.2402.16568.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. AllenNLP: A Deep Semantic Natural Language Processing Platform. In Eunjeong L. Park, Masato Hagiwara, Dmitrijs Milajevs, and Liling Tan, editors, *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6, Melbourne, Australia, July 2018. Association for Computational Linguistics. URL https://aclanthology.org/W18-2501.

Nikhil Garg, Benoît Favre, Korbinian Riedhammer, and Dilek Hakkani-Tür. Clusterrank: a graph based method for meeting summarization. In *INTERSPEECH 2009, 10th Annual Conference of the International Speech Communication Association, Brighton, United Kingdom, September 6-10, 2009*, pages 1499–1502. ISCA, 2009. URL https://doi.org/10.21437/Interspeech.2009-456.

Aparna Garimella, Abhilasha Sancheti, Vinay Aggarwal, Ananya Ganesh, Niyati Chhaya, and Nandakishore Kambhatla. Text Simplification for Legal Domain: Insights and Challenges. In Nikolaos Aletras, Ilias Chalkidis, Leslie Barrett, Cătălina Goanță, and Daniel Preoțiuc-Pietro, editors, *Proceedings of the Natural Legal Language Processing Workshop 2022*, pages 296–304, Abu Dhabi, United Arab Emirates (Hybrid), December 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.nllp-1.28.

Christoph Gebendorfer and Ahmed Elnaggar. Legal JRC-Acquis Sum – Text Summarization Corpus. In *Technical University of Munich*. (Date accessed: 21.06.2022), 2018. doi: 10.14459/2018md1446654.

Sebastian Gehrmann, Yuntian Deng, and Alexander Rush. Bottom-Up Abstractive Summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4098–4109, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. URL `https://aclanthology.org/D18-1443`.

Michael Gertz and Dennis Aumiller. Legal Tech und Deep Learning – Eine Bestandsaufnahme. *Legal Tech Zeitschrift - LTZ*, 2022/01:30–36, March 2022.

Demian Gholipour Ghalandari and Georgiana Ifrim. Examining the State-of-the-Art in News Timeline Summarization. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1322–1334, Online, July 2020. Association for Computational Linguistics. URL `https://aclanthology.org/2020.acl-main.122`.

John M. Giorgi, Luca Soldaini, Bo Wang, Gary D. Bader, Kyle Lo, Lucy Lu Wang, and Arman Cohan. Exploring the Challenges of Open Domain Multi-Document Summarization. *CoRR*, abs/2212.10526, 2022. URL `https://doi.org/10.48550/arXiv.2212.10526`.

Ingo Glaser, Sebastian Moser, and Florian Matthes. Sentence Boundary Detection in German Legal Documents. In Ana Paula Rocha, Luc Steels, and H. Jaap van den Herik, editors, *Proceedings of the 13th International Conference on Agents and Artificial Intelligence, ICAART 2021, Volume 2, Online Streaming, February 4-6, 2021*, pages 812–821. SCITEPRESS, 2021a. URL `https://doi.org/10.5220/0010246308120821`.

Ingo Glaser, Sebastian Moser, and Florian Matthes. Summarization of German Court Rulings. In *Proceedings of the Natural Legal Language Processing Workshop 2021*, pages 180–189, Punta Cana, Dominican Republic, November 2021b. Association for Computational Linguistics. URL `https://aclanthology.org/2021.nllp-1.19`.

Daniel J. Goff and Thomas W. Loehfelm. Automated Radiology Report Summarization Using an Open-Source Natural Language Processing Pipeline. *Journal of Digital Imaging*, 31(2): 185–192, 2018. URL `https://doi.org/10.1007/s10278-017-0030-2`.

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. Making Science Simple: Corpora for the Lay Summarisation of Scientific Literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. URL `https://aclanthology.org/2022.emnlp-main.724`.

Jade Goldstein and Jaime Carbonell. Summarization: (1) Using MMR for Diversity- Based Reranking and (2) Evaluating Summaries. In *TIPSTER TEXT PROGRAM PHASE III: Proceedings of a Workshop held at Baltimore, Maryland, October 13-15, 1998*, pages 181–195, Baltimore, Maryland, USA, October 1998. Association for Computational Linguistics. doi: 10.3115/1119089.1119120. URL `https://aclanthology.org/X98-1025`.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

Sian Gooding and Ekaterina Kochmar. CAMB at CWI Shared Task 2018: Complex Word Identification with Ensemble-Based Voting. In Joel Tetreault, Jill Burstein, Ekaterina Kochmar, Claudia Leacock, and Helen Yannakoudakis, editors, *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 184–194, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. URL `https://aclanthology.org/W18-0520`.

Sian Gooding and Ekaterina Kochmar. Complex Word Identification as a Sequence Labelling Task. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1148–1153, Florence, Italy, July 2019. Association for Computational Linguistics. URL `https://aclanthology.org/P19-1109`.

Sian Gooding and Manuel Tragut. One Size Does Not Fit All: The Case for Personalised Word Complexity Models. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 353–365, Seattle, United States, July 2022. Association for Computational Linguistics. URL `https://aclanthology.org/2022.findings-naacl.27`.

Sian Gooding, Ekaterina Kochmar, Seid Muhie Yimam, and Chris Biemann. Word Complexity is in the Eye of the Beholder. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4439–4449, Online, June 2021. Association for Computational Linguistics. URL `https://aclanthology.org/2021.naacl-main.351`.

Ben Goodrich, Vinay Rao, Peter J. Liu, and Mohammad Saleh. Assessing The Factual Accuracy of Generated Text. In Ankur Teredesai, Vipin Kumar, Ying Li, Rómer Rosales, Evimaria Terzi, and George Karypis, editors, *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 166–175. ACM, 2019. doi: 10.1145/3292500.3330955.

Tanya Goyal, Junyi Jessy Li, and Greg Durrett. News Summarization and Evaluation in the Era of GPT-3. *CoRR*, abs/2209.12356, 2022. URL `https://doi.org/10.48550/arXiv.2209.12356`.

Yvette Graham. Re-evaluating Automatic Summarization with BLEU and 192 Shades of ROUGE. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 128–137, Lisbon, Portugal, September 2015. Association for Computational Linguistics. URL `https://aclanthology.org/D15-1013`.

Gintarė Grigonyte, Maria Kvist, Sumithra Velupillai, and Mats Wirén. Improving Readability of Swedish Electronic Health Records through Lexical Simplification: First Results. In *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR)*, pages 74–83, Gothenburg, Sweden, April 2014. Association for Computational Linguistics. doi: 10.3115/v1/W14-1209. URL `https://aclanthology.org/W14-1209`.

Max Grusky, Mor Naaman, and Yoav Artzi. Newsroom: A Dataset of 1.3 Million Summaries with Diverse Extractive Strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 708–719, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. URL https://aclanthology.org/N18-1065.

Yanzhu Guo, Chloé Clavel, Moussa Kamal Eddine, and Michalis Vazirgiannis. Questioning the Validity of Summarization Datasets and Improving Their Factual Consistency. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5716–5727, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.emnlp-main.386.

Yi Guo and George Stylios. An intelligent summarization system based on cognitive psychology. *Information Sciences*, 174(1-2):1–36, 2005. URL https://doi.org/10.1016/j.ins.2004.08.004.

Julia Hancke, Sowmya Vajjala, and Detmar Meurers. Readability Classification for German using Lexical, Syntactic, and Morphological Features. In *Proceedings of COLING 2012*, pages 1063–1080, Mumbai, India, December 2012. The COLING 2012 Organizing Committee. URL https://www.aclweb.org/anthology/C12-1065.

Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar. XL-Sum: Large-Scale Multilingual Abstractive Summarization for 44 Languages. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4693–4703, Online, August 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.findings-acl.413.

Philip Hausner, Dennis Aumiller, and Michael Gertz. TiCCo: Time-Centric Content Exploration. In Mathieu d'Aquin, Stefan Dietze, Claudia Hauff, Edward Curry, and Philippe Cudré-Mauroux, editors, *CIKM '20: The 29th ACM International Conference on Information and Knowledge Management, Virtual Event, Ireland, October 19-23, 2020*, pages 3413–3416. ACM, 2020a. doi: 10.1145/3340531.3417432.

Philip Hausner, Dennis Aumiller, and Michael Gertz. Time-centric Exploration of Court Documents. In Ricardo Campos, Alípio Mário Jorge, Adam Jatowt, and Sumit Bhatia, editors, *Proceedings of Text2Story - Third Workshop on Narrative Extraction From Texts co-located with 42nd European Conference on Information Retrieval, Text2Story@ECIR 2020, Lisbon, Portugal, April 14th, 2020 [online only]*, volume 2593 of *CEUR Workshop Proceedings*, pages 31–37. CEUR-WS.org, 2020b. URL https://ceur-ws.org/Vol-2593/paper4.pdf.

Hiroaki Hayashi, Prashant Budania, Peng Wang, Chris Ackerson, Raj Neervannan, and Graham Neubig. WikiAsp: A Dataset for Multi-domain Aspect-based Summarization. *Transactions of the Association for Computational Linguistics*, 9:211–225, 2021. URL https://aclanthology.org/2021.tacl-1.13.

Junxian He, Wojciech Kryscinski, Bryan McCann, Nazneen Rajani, and Caiming Xiong. CTRLsum: Towards Generic Controllable Text Summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5879–5915, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. URL `https://aclanthology.org/2022.emnlp-main.396`.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: decoding-Enhanced Bert with Disentangled Attention. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL `https://openreview.net/forum?id=XPZIaotutsD`.

Karl Moritz Hermann, Tomás Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching Machines to Read and Comprehend. In Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 1693–1701, 2015. URL `https://proceedings.neurips.cc/paper/2015/hash/afdec7005cc9f14302cd0474fd0f3c96-Abstract.html`.

Freya Hewett and Manfred Stede. Automatically evaluating the conceptual complexity of German texts. In *Proceedings of the 17th Conference on Natural Language Processing (KONVENS 2021)*, pages 228–234, Düsseldorf, Germany, 6–9 September 2021. KONVENS 2021 Organizers. URL `https://aclanthology.org/2021.konvens-1.23`.

Freya Hewett and Manfred Stede. Extractive Summarisation for German-language Data: A Text-level Approach with Discourse Features. In Nicoletta Calzolari, Chu-Ren Huang, Hansaem Kim, James Pustejovsky, Leo Wanner, Key-Sun Choi, Pum-Mo Ryu, Hsin-Hsi Chen, Lucia Donatelli, Heng Ji, Sadao Kurohashi, Patrizia Paggio, Nianwen Xue, Seokhwan Kim, Younggyun Hahm, Zhong He, Tony K. Lee, Enrico Santus, Francis Bond, and Seung-Hoon Na, editors, *Proceedings of the 29th International Conference on Computational Linguistics*, pages 756–765, Gyeongju, Republic of Korea, October 2022. International Committee on Computational Linguistics. URL `https://aclanthology.org/2022.coling-1.63`.

Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory. *Neural Computation*, 9 (8):1735–1780, 1997. URL `https://doi.org/10.1162/neco.1997.9.8.1735`.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training Compute-Optimal Large Language Models. *CoRR*, abs/2203.15556, 2022. URL `https://doi.org/10.48550/arXiv.2203.15556`.

Jeremy Howard and Sebastian Ruder. Universal Language Model Fine-tuning for Text Classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia, July 2018. Association for Computational Linguistics. URL `https://aclanthology.org/P18-1031`.

Hanyang Hu, Cor-Paul Bezemer, and Ahmed E. Hassan. Studying the consistency of star ratings and the complaints in 1 & 2-star user reviews for top free cross-platform Android and iOS apps. *Empirical Software Engineering*, 23(6):3442–3475, 2018. URL https://doi.org/10.1007/s10664-018-9604-y.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P. Xing. Toward Controlled Generation of Text. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pages 1587–1596. JMLR.org, 2017.

Dandan Huang, Leyang Cui, Sen Yang, Guangsheng Bao, Kun Wang, Jun Xie, and Yue Zhang. What Have We Achieved on Text Summarization? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 446–469, Online, November 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.emnlp-main.33.

William Hwang, Hannaneh Hajishirzi, Mari Ostendorf, and Wei Wu. Aligning Sentences from Standard Wikipedia to Simple Wikipedia. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 211–217, Denver, Colorado, May–June 2015. Association for Computational Linguistics. doi: 10.3115/v1/N15-1022. URL https://www.aclweb.org/anthology/N15-1022.

Stephanie Inglis, Ehud Reiter, and Somayajulu Sripada. Textually Summarising Incomplete Data. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 228–232, Santiago de Compostela, Spain, September 2017. Association for Computational Linguistics. URL https://aclanthology.org/W17-3535.

Neslihan Iskender, Tim Polzehl, and Sebastian Möller. Best Practices for Crowd-based Evaluation of German Summarization: Comparing Crowd, Expert and Automatic Evaluation. In *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems*, pages 164–175, Online, November 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.eval4nlp-1.16.

Myungha Jang and James Allan. Explaining Controversy on Social Media via Stance Summarization. In Kevyn Collins-Thompson, Qiaozhu Mei, Brian D. Davison, Yiqun Liu, and Emine Yilmaz, editors, *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018*, pages 1221–1224. ACM, 2018. doi: 10.1145/3209978.3210143.

Katharina Jeblick, Balthasar Schachtner, Jakob Dexl, Andreas Mittermeier, Anna Theresa Stüber, Johanna Topalis, Tobias Weber, Philipp Wesp, Bastian O. Sabel, Jens Ricke, and Michael Ingrisch. ChatGPT Makes Medicine Easy to Swallow: An Exploratory Case Study on Simplified Radiology Reports. *European Radiology*, 2023. URL https://doi.org/10.1007/s00330-023-10213-1.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of Hallucination in Natural Language Generation.

ACM Computing Surveys, 55(12), mar 2023. ISSN 0360-0300. URL https://doi.org/10.1145/3571730.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de Las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mixtral of Experts. *CoRR*, abs/2401.04088, 2024. URL https://doi.org/10.48550/arXiv.2401.04088.

Chao Jiang, Mounica Maddela, Wuwei Lan, Yang Zhong, and Wei Xu. Neural CRF Model for Sentence Alignment in Text Simplification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7943–7960, Online, July 2020. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2020.acl-main.709.

Hanqi Jin, Tianming Wang, and Xiaojun Wan. SemSUM: Semantic Dependency Guided Neural Abstractive Summarization. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8026–8033. AAAI Press, 2020. URL https://doi.org/10.1609/aaai.v34i05.6312.

Zhiwei Jin, Juan Cao, Yongdong Zhang, and Jiebo Luo. News Verification by Exploiting Conflicting Social Viewpoints in Microblogs. In Dale Schuurmans and Michael P. Wellman, editors, *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA*, pages 2972–2978. AAAI Press, 2016. URL http://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/12128.

Baoyu Jing, Zeya Wang, and Eric Xing. Show, Describe and Conclude: On Exploiting the Structure Information of Chest X-ray Reports. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6570–6580, Florence, Italy, July 2019. Association for Computational Linguistics. URL https://aclanthology.org/P19-1657.

Timo Johner, Abhik Jana, and Chris Biemann. Error Analysis of using BART for Multi-Document Summarization: A Study for English and German Language. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 391–397, Reykjavik, Iceland (Online), May 31–2 June 2021. Linköping University Electronic Press, Sweden. URL https://aclanthology.org/2021.nodalida-main.43.

Karen Sparck Jones. Automatic Summarising: Factors and Directions. *Advances in Automatic Text Summarization*, 1999.

Karen Spärck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 60(5):493–502, 2004. doi: 10.1108/00220410410560573. URL https://doi.org/10.1108/00220410410560573.

Anirudh Joshi, Namit Katariya, Xavier Amatriain, and Anitha Kannan. Dr. Summarize: Global Summarization of Medical Dialogue by Exploiting Local Structures. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3755–3763, Online, November 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.findings-emnlp.335.

Dan Jurafsky and James H. Martin. *Speech and Language Processing: an Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, 2nd Edition*. Prentice Hall series in artificial intelligence. Prentice Hall, Pearson Education International, 2009. ISBN 9780135041963. URL https://www.worldcat.org/oclc/315913020.

Ryo Kamoi, Tanya Goyal, and Greg Durrett. Shortcomings of Question Answering Based Factuality Frameworks for Error Localization. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 132–146, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.eacl-main.11.

Chris Kamphuis, Arjen P. de Vries, Leonid Boytsov, and Jimmy Lin. Which BM25 Do You Mean? A Large-Scale Reproducibility Study of Scoring Variants. In Joemon M. Jose, Emine Yilmaz, João Magalhães, Pablo Castells, Nicola Ferro, Mário J. Silva, and Flávio Martins, editors, *Advances in Information Retrieval - 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14-17, 2020, Proceedings, Part II*, volume 12036 of *Lecture Notes in Computer Science*, pages 28–34. Springer, 2020. URL https://doi.org/10.1007/978-3-030-45442-5_4.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online, November 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.emnlp-main.550.

David Kauchak. Improving Text Simplification Language Modeling Using Unsimplified Text Data. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1537–1546, Sofia, Bulgaria, August 2013. Association for Computational Linguistics. URL https://aclanthology.org/P13-1151.

Nitish Shirish Keskar, Bryan McCann, Lav R. Varshney, Caiming Xiong, and Richard Socher. CTRL: A conditional transformer language model for controllable generation. *CoRR*, abs/1909.05858, 2019. URL http://arxiv.org/abs/1909.05858.

Tannon Kew, Alison Chi, Laura Vásquez-Rodríguez, Sweta Agrawal, Dennis Aumiller, Fernando Alva-Manchego, and Matthew Shardlow. BLESS: Benchmarking Large Language Models on Sentence Simplification. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13291–13309, Singapore, December 2023a. Association for Computational Linguistics. URL https://aclanthology.org/2023.emnlp-main.821.

Tannon Kew, Marek Kostrzewa, and Sarah Ebling. 20 Minuten: A Multi-task News Summarisation Dataset for German. In Hatem Ghorbel, Maria Sokhn, Mark Cieliebak, Manuela Hürlimann, Emmanuel de Salis, and Jonathan Guerne, editors, *Proceedings of the 8th edition of the Swiss Text Analytics Conference*, pages 1–13, Neuchatel, Switzerland, June 2023b. Association for Computational Linguistics. URL `https://aclanthology.org/2023.swisstext-1.1`.

Yuta Kikuchi, Graham Neubig, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura. Controlling Output Length in Neural Encoder-Decoders. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1328–1338, Austin, Texas, November 2016. Association for Computational Linguistics. URL `https://aclanthology.org/D16-1140`.

David Klaper, Sarah Ebling, and Martin Volk. Building a German/Simple German Parallel Corpus for Automatic Text Simplification. In *Proceedings of the Second Workshop on Predicting and Improving Text Readability for Target Reader Populations*, pages 11–19, Sofia, Bulgaria, August 2013. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/W13-2902`.

Svea Klaus, Ria Van Hecke, Kaweh Djafari Naini, Ismail Sengor Altingovde, Juan Bernabé-Moreno, and Enrique Herrera-Viedma. Summarizing Legal Regulatory Documents using Transformers. In Enrique Amigó, Pablo Castells, Julio Gonzalo, Ben Carterette, J. Shane Culpepper, and Gabriella Kazai, editors, *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, pages 2426–2430. ACM, 2022. doi: 10.1145/3477495.3531872.

Philipp Koehn. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of Machine Translation Summit X: Papers*, pages 79–86, Phuket, Thailand, September 13-15 2005. URL `https://aclanthology.org/2005.mtsummit-papers.11`.

Anastassia Kornilova and Vladimir Eidelman. BillSum: A Corpus for Automatic Summarization of US Legislation. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 48–56, Hong Kong, China, November 2019. Association for Computational Linguistics. URL `https://aclanthology.org/D19-5406`.

Mahnaz Koupaee and William Yang Wang. WikiHow: A Large Scale Text Summarization Dataset. *CoRR*, abs/1810.09305, 2018. URL `http://arxiv.org/abs/1810.09305`.

Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmungkol Sarin, Sokhar Samb, Benoît Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Ortiz Suarez, Iroro Orife, Kelechi Ogueji, Andre Niyongabo Rubungo, Toan Q. Nguyen, Mathias Müller, André Müller, Shamsuddeen Hassan Muhammad, Nanda Muhammad, Ayanda Mnyakeni, Jamshidbek Mirzakhalov, Tapiwanashe Matangira, Colin Leong, Nze Lawson, Sneha Kudugunta, Yacine Jernite, Mathias Jenny, Orhan Firat, Bonaventure F. P. Dossou, Sakhile Dlamini, Nisansa de Silva, Sakine Çabuk Ballı, Stella Biderman, Alessia Battisti, Ahmed Baruwa, Ankur Bapna, Pallavi Baljekar, Israel Abebe Azime, Ayodele Awokoya, Duygu Ataman, Orevaoghene Ahia, Oghenefego Ahia, Sweta Agrawal, and Mofetoluwa Adeyemi. Quality at a Glance: An Audit of Web-Crawled Multilingual Datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72, 2022. doi: 10.1162/tacl_a_00447. URL https://aclanthology.org/2022.tacl-1.4.

Kalpesh Krishna, Erin Bransom, Bailey Kuehl, Mohit Iyyer, Pradeep Dasigi, Arman Cohan, and Kyle Lo. LongEval: Guidelines for Human Evaluation of Faithfulness in Long-form Summarization. In Andreas Vlachos and Isabelle Augenstein, editors, *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1650–1669, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.eacl-main.121.

Kundan Krishna and Balaji Vasan Srinivasan. Generating Topic-Oriented Summaries Using Neural Attention. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1697–1705, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. URL https://aclanthology.org/N18-1153.

Wojciech Kryscinski, Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. Neural Text Summarization: A Critical Evaluation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 540–551, Hong Kong, China, November 2019. Association for Computational Linguistics. URL https://aclanthology.org/D19-1051.

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. Evaluating the Factual Consistency of Abstractive Text Summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346, Online, November 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.emnlp-main.750.

Wojciech Kryscinski, Nazneen Rajani, Divyansh Agarwal, Caiming Xiong, and Dragomir R. Radev. BOOKSUM: A Collection of Datasets for Long-form Narrative Summarization. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6536–6558, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.findings-emnlp.488.

Sayali Kulkarni, Sheide Chammas, Wan Zhu, Fei Sha, and Eugene Ie. CoMSum and SIBERT: A Dataset and Neural Model for Query-Based Multi-document Summarization. In *Document Analysis and Recognition–ICDAR 2021: 16th International Conference, Lausanne, Switzerland, September 5–10, 2021, Proceedings, Part II 16*, pages 84–98. Springer, 2021.

Sachin Kumar, Biswajit Paria, and Yulia Tsvetkov. Gradient-based Constrained Sampling from Language Models. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings* *of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2251–2277, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. URL `https://aclanthology.org/2022.emnlp-main.144`.

Moreno La Quatra and Luca Cagliero. End-to-end Training For Financial Report Summarization. In *Proceedings of the 1st Joint Workshop on Financial Narrative Processing and MultiLing Financial Summarisation*, pages 118–123, Barcelona, Spain (Online), December 2020. COLING. URL `https://aclanthology.org/2020.fnp-1.20`.

Faisal Ladhak, Esin Durmus, Claire Cardie, and Kathleen McKeown. WikiLingua: A New Benchmark Dataset for Cross-Lingual Abstractive Summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4034–4048, Online, November 2020. Association for Computational Linguistics. URL `https://aclanthology.org/2020.findings-emnlp.360`.

Lukas Lange, Anastasiia Iurshina, Heike Adel, and Jannik Strötgen. Adversarial Alignment of Multilingual Models for Extracting Temporal Expressions from Text. In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 103–109, Online, July 2020. Association for Computational Linguistics. URL `https://aclanthology.org/2020.repl4nlp-1.14`.

Egoitz Laparra, Dongfang Xu, and Steven Bethard. From Characters to Time Intervals: New Paradigms for Evaluation and Neural Parsing of Time Normalizations. *Transactions of the Association for Computational Linguistics*, 6:343–356, 2018. doi: 10.1162/tacl_a_00025. URL `https://aclanthology.org/Q18-1025`.

John Lawrence and Chris Reed. Argument Mining: A Survey. *Computational Linguistics*, 45 (4):765–818, December 2019. doi: 10.1162/coli_a_00364. URL `https://aclanthology.org/J19-4006`.

John Lee and Chak Yan Yeung. Personalizing Lexical Simplification. In Emily M. Bender, Leon Derczynski, and Pierre Isabelle, editors, *Proceedings of the 27th International Conference on Computational Linguistics*, pages 224–232, Santa Fe, New Mexico, USA, August 2018. Association for Computational Linguistics. URL `https://aclanthology.org/C18-1019`.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. End-to-end Neural Coreference Resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. URL `https://aclanthology.org/D17-1018`.

Jochen L Leidner. Summarization in the Financial and Regulatory Domain. In *Trends and Applications of Text Summarization Techniques*, pages 187–215. IGI Global, 2020.

Sharon Levy, Emily Allaway, Melanie Subbiah, Lydia Chilton, Desmond Patton, Kathleen McKeown, and William Yang Wang. SafeText: A Benchmark for Exploring Physical Safety in Language Models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2407–2421, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. URL `https://aclanthology.org/2022.emnlp-main.154`.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, July 2020a. Association for Computational Linguistics. URL https://aclanthology.org/2020.acl-main.703.

Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020b. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Chenliang Li, Weiran Xu, Si Li, and Sheng Gao. Guiding Generation for Abstractive Text Summarization Based on Key Information Guide Network. In Marilyn Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 55–60, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. URL https://aclanthology.org/N18-2009.

Jiahui Li. Styled Text Summarization via Domain-Specific Paraphrasing. Master's thesis, Heidelberg University, 2023.

Yiyang Li, Lei Li, Qing Yang, Marina Litvak, Natalia Vanetik, Dingxin Hu, Yuze Li, Yanquan Zhou, Dongliang Xu, and Xuanyu Zhang. Just ClozE! A Fast and Simple Method for Evaluating the Factual Consistency in Abstractive Summarization. *CoRR*, abs/2210.02804, 2022. URL https://arxiv.org/abs/2210.02804.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards General Text Embeddings with Multi-stage Contrastive Learning. *CoRR*, abs/2308.03281, 2023. URL https://doi.org/10.48550/arXiv.2308.03281.

Siting Liang, Klaus Kades, Matthias Fink, Peter Full, Tim Weber, Jens Kleesiek, Michael Strube, and Klaus Maier-Hein. Fine-tuning BERT Models for Summarizing German Radiology Findings. In *Proceedings of the 4th Clinical Natural Language Processing Workshop*, pages 30–40, Seattle, WA, July 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.clinicalnlp-1.4.

Rensis Likert. A technique for the Measurement of Attitudes. *Archives of Psychology*, 140, 1932.

Jung-Min Lim, In-Su Kang, and Jong-Hyeok Lee. Multi-Document Summarization Using Cross-Language Texts. In Noriko Kando and Haruko Ishikawa, editors, *Proceedings of the Fourth*

NTCIR Workshop on Research in Information Access Technologies Information Retrieval, Question Answering and Summarization, NTCIR-4, National Center of Sciences, Tokyo, Japan, June 2-4, 2004. National Institute of Informatics (NII), 2004. URL http://research.nii.ac.jp/ntcir/workshop/OnlineProceedings4/TSC/NTCIR4-TSC-LimJM.pdf.

Chin-Yew Lin. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://aclanthology.org/W04-1013.

Chin-Yew Lin and Eduard Hovy. From Single to Multi-document Summarization. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 457–464, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073160. URL https://aclanthology.org/P02-1058.

Jimmy Lin, Rodrigo Frassetto Nogueira, and Andrew Yates. Pretrained Transformers for Text Ranking: BERT and Beyond. *CoRR*, abs/2010.06467, 2020. URL https://arxiv.org/abs/2010.06467.

Marina Litvak and Mark Last. Cross-lingual training of summarization systems using annotated corpora in a foreign language. *Information Retrieval*, 16(5):629–656, 2013. URL https://doi.org/10.1007/s10791-012-9210-3.

Hao Liu, Matei Zaharia, and Pieter Abbeel. Ring Attention with Blockwise Transformers for Near-Infinite Context. *CoRR*, abs/2310.01889, 2023a. URL https://doi.org/10.48550/arXiv.2310.01889.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. Recurrent Neural Network for Text Classification with Multi-Task Learning. In Subbarao Kambhampati, editor, *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 2873–2879. IJCAI/AAAI Press, 2016. URL http://www.ijcai.org/Abstract/16/408.

Peter J. Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. Generating Wikipedia by Summarizing Long Sequences. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018a. URL https://openreview.net/forum?id=Hyg0vbWC-.

Yang Liu and Mirella Lapata. Hierarchical Transformers for Multi-Document Summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5070–5081, Florence, Italy, July 2019. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P19-1500.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-Eval: NLG Evaluation using GPT-4 with Better Human Alignment. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore, December 2023b. Association for Computational Linguistics. URL https://aclanthology.org/2023.emnlp-main.153.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. Multilingual Denoising Pre-training for Neural Machine Translation. *Transactions of the Association for Computational Linguistics*, 8:726–742, 2020. doi: 10.1162/tacl_a_00343.

Yizhu Liu, Zhiyi Luo, and Kenny Zhu. Controlling Length in Abstractive Summarization Using a Convolutional Neural Network. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4110–4119, Brussels, Belgium, October-November 2018b. Association for Computational Linguistics. URL https://aclanthology.org/D18-1444.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V. Le, Barret Zoph, Jason Wei, and Adam Roberts. The Flan Collection: Designing Data and Methods for Effective Instruction Tuning. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 22631–22648. PMLR, 2023. URL https://proceedings.mlr.press/v202/longpre23a.html.

Manuel J. Maña López, Manuel de Buenaga Rodríguez, and José María Gómez Hidalgo. Using and Evaluating User Directed Summaries to Improve Information Access. In Serge Abiteboul and Anne-Marie Vercoustre, editors, *Research and Advanced Technology for Digital Libraries, Third European Conference, ECDL'99, Paris, France, September 22-24, 1999, Proceedings*, volume 1696 of *Lecture Notes in Computer Science*, pages 198–214. Springer, 1999. URL https://doi.org/10.1007/3-540-48155-9_14.

Eneldo Loza Mencía and Johannes Fürnkranz. Efficient Multilabel Classification Algorithms for Large-Scale Problems in the Legal Domain. In Enrico Francesconi, Simonetta Montemagni, Wim Peters, and Daniela Tiscornia, editors, *Semantic Processing of Legal Texts – Where the Language of Law Meets the Law of Language*, volume 6036 of *Lecture Notes in Artificial Intelligence*, pages 192–215. Springer-Verlag, 1 edition, May 2010. ISBN 978-3-642-12836-3. doi: 10.1007/978-3-642-12837-0_11. URL http://www.ke.tu-darmstadt.de/publications/papers/loza10eurlex.pdf.

Hans Peter Luhn. The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development*, 2(2):159–165, 1958. doi: 10.1147/rd.22.0159.

Liang Ma, Shuyang Cao, Robert L. Logan IV, Di Lu, Shihao Ran, Ke Zhang, Joel Tetreault, and Alejandro Jaimes. BUMP: A Benchmark of Unfaithful Minimal Pairs for Meta-Evaluation of Faithfulness Metrics. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12788–12812, Toronto, Canada, July 2023a. Association for Computational Linguistics. URL https://aclanthology.org/2023.acl-long.716.

Yubo Ma, Yixin Cao, Yong Hong, and Aixin Sun. Large Language Model Is Not a Good Few-shot Information Extractor, but a Good Reranker for Hard Samples! In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10572–10601, Singapore, December 2023b. Association for Computational Linguistics. URL https://aclanthology.org/2023.findings-emnlp.710.

Sean MacAvaney, Sajad Sotudeh, Arman Cohan, Nazli Goharian, Ish A. Talati, and Ross W. Filice. Ontology-Aware Clinical Abstractive Summarization. In Benjamin Piwowarski, Max Chevalier, Éric Gaussier, Yoelle Maarek, Jian-Yun Nie, and Falk Scholer, editors, *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*, pages 1013–1016. ACM, 2019. URL https://doi.org/10.1145/3331184.3331319.

Mounica Maddela, Mayank Kulkarni, and Daniel Preotiuc-Pietro. EntSUM: A Data Set for Entity-Centric Extractive Summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3355–3366, Dublin, Ireland, May 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.acl-long.237.

Jonathan Mallinson, Rico Sennrich, and Mirella Lapata. Zero-Shot Crosslingual Sentence Simplification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5109–5126, Online, November 2020. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/2020.emnlp-main.415.

Inderjeet Mani. Summarization Evaluation: An Overview. In *Proceedings of the Third Second Workshop Meeting on Evaluation of Chinese & Japanese Text Retrieval and Text Summarization, NTCIR-2, Tokyo, Japan, March 7-9, 2001.* National Institute of Informatics (NII), 2001. URL http://research.nii.ac.jp/ntcir/workshop/OnlineProceedings2/sum-mani.pdf.

Inderjeet Mani, Gary Klein, David House, Lynette Hirschman, Therese Firmin, and Beth Sundheim. SUMMAC: a text summarization evaluation. *Natural Language Engineering*, 8(1): 43–68, 2002. URL https://doi.org/10.1017/S1351324901002741.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. The Stanford CoreNLP Natural Language Processing Toolkit. In Kalina Bontcheva and Jingbo Zhu, editors, *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-5010. URL https://aclanthology.org/P14-5010.

Laura Manor and Junyi Jessy Li. Plain English Summarization of Contracts. In *Proceedings of the Natural Legal Language Processing Workshop 2019*, pages 1–11, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. URL https://aclanthology.org/W19-2201.

Max Marion, Ahmet Üstün, Luiza Pozzobon, Alex Wang, Marzieh Fadaee, and Sara Hooker. When Less is More: Investigating Data Pruning for Pretraining LLMs at Scale. *CoRR*, abs/2309.04564, 2023. URL https://doi.org/10.48550/arXiv.2309.04564.

Lluís Màrquez, Xavier Carreras, Kenneth C. Litkowski, and Suzanne Stevenson. Semantic Role Labeling: An Introduction to the Special Issue. *Computational Linguistics*, 34(2):145–159, 2008. doi: 10.1162/coli.2008.34.2.145.

Elaine Marsh, Henry Hamburger, and Ralph Grishman. A Production Rule System for Message Summarization. In *Proceedings of the Fourth AAAI Conference on Artificial Intelligence*, AAAI'84, pages 243–246. AAAI Press, 1984.

Sebastian Martschat and Katja Markert. A Temporally Sensitive Submodularity Framework for Timeline Summarization. In Anna Korhonen and Ivan Titov, editors, *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 230–240, Brussels, Belgium, October 2018. Association for Computational Linguistics. URL https://aclanthology.org/K18-1023.

Laura Mascarell, Ribin Chalumattu, and Julien Heitmann. Entropy-based Sampling for Abstractive Multi-document Summarization in Low-resource Settings. In C. Maria Keet, Hung-Yi Lee, and Sina Zarrieß, editors, *Proceedings of the 16th International Natural Language Generation Conference*, pages 123–133, Prague, Czechia, September 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.inlg-main.9.

Mark T. Maybury. *Karen Spärck Jones and Summarization*, pages 99–103. Springer Netherlands, Dordrecht, 2005. ISBN 978-1-4020-3467-1. doi: 10.1007/1-4020-3467-9_7. URL https://doi.org/10.1007/1-4020-3467-9_7.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On Faithfulness and Factuality in Abstractive Summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online, July 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.acl-main.173.

Rada Mihalcea and Paul Tarau. TextRank: Bringing Order into Text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://aclanthology.org/W04-3252.

Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. In Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119, 2013. URL https://proceedings.neurips.cc/paper/2013/hash/9aa42b31882ec039965f3c4923ce901b-Abstract.html.

Benjamin Minixhofer, Fabian Paischer, and Navid Rekabsaz. WECHSEL: Effective initialization of subword embeddings for cross-lingual transfer of monolingual language models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3992–4006, Seattle, United States, July 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.naacl-main.293.

Anshuman Mishra, Dhruvesh Patel, Aparna Vijayakumar, Xiang Lorraine Li, Pavan Kapanipathi, and Kartik Talamadupula. Looking Beyond Sentence-Level Natural Language Inference for Question Answering and Text Summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1322–1336, Online, June 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.naacl-main.104.

Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model Cards for Model Reporting. In danah boyd and Jamie H. Morgenstern, editors, *Proceedings of the Conference on Fairness, Accountability, and Transparency, FAT* 2019, Atlanta, GA, USA, January 29-31, 2019*, pages 220–229. ACM, 2019. doi: 10.1145/3287560.3287596. URL https://doi.org/10.1145/3287560.3287596.

Salar Mohtaj, Babak Naderi, and Sebastian Möller. Overview of the GermEval 2022 Shared Task on Text Complexity Assessment of German Text. In Sebastian Möller, Salar Mohtaj, and Babak Naderi, editors, *Proceedings of the GermEval 2022 Workshop on Text Complexity Assessment of German Text*, pages 1–9, Potsdam, Germany, September 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.germeval-1.1.

Ines Montani, Matthew Honnibal, Matthew Honnibal, Sofie Van Landeghem, Adriane Boyd, Henning Peters, Paul O'Leary McCann, jim geovedi, Jim O'Regan, Maxim Samsonov, György Orosz, Daniël de Kok, Marcus Blättermann, Duygu Altinok, Raphael Mitsch, Madeesh Kannan, Søren Lind Kristiansen, Edward, Raphaël Bournhonesque, Lj Miranda, Peter Baumgartner, Richard Hudson, Explosion Bot, Roman, Leander Fiedler, Ryn Daniels, Wannaphong Phatthiyaphaibun, Grégory Howard, and Yohei Tamura. explosion/spaCy: Industrial-strength Natural Language Processing, April 2023. URL https://doi.org/10.5281/zenodo.7820813.

Ramesh Nallapati, Bowen Zhou, Cícero Nogueira dos Santos, Çaglar Gülçehre, and Bing Xiang. Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond. In Yoav Goldberg and Stefan Riezler, editors, *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, Berlin, Germany, August 11-12, 2016*, pages 280–290. ACL, 2016. URL https://doi.org/10.18653/v1/k16-1028.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. SummaRuNNer: A Recurrent Neural Network Based Sequence Model for Extractive Summarization of Documents. In Satinder Singh and Shaul Markovitch, editors, *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 3075–3081. AAAI Press, 2017. URL http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14636.

Feng Nan, Cicero Nogueira dos Santos, Henghui Zhu, Patrick Ng, Kathleen McKeown, Ramesh Nallapati, Dejiao Zhang, Zhiguo Wang, Andrew O. Arnold, and Bing Xiang. Improving Factual Consistency of Abstractive Summarization via Question Answering. In *Proceedings* *of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6881–6894, Online, August 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.acl-long.536.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. Don't Give Me the Details, Just the Summary! Topic-Aware Convolutional Neural Networks for Extreme Summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. URL https://aclanthology.org/D18-1206.

David Fraile Navarro, Mark Dras, and Shlomo Berkovsky. Few-shot fine-tuning SOTA summarization models for medical dialogues. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Student Research Workshop*, pages 254–266, Hybrid: Seattle, Washington + Online, July 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.naacl-srw.32.

Ani Nenkova and Rebecca Passonneau. Evaluating Content Selection in Summarization: The Pyramid Method. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 145–152, Boston, Massachusetts, USA, May 2 - May 7 2004. Association for Computational Linguistics. URL https://aclanthology.org/N04-1019.

Ani Nenkova and Lucy Vanderwende. The Impact of Frequency on Summarization. *Microsoft Research, Redmond, Washington, Technical Report MSR-TR-2005*, 101, 2005.

Khanh Nguyen and Hal Daumé III. Global Voices: Crossing Borders in Automatic News Summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 90–97, Hong Kong, China, November 2019. Association for Computational Linguistics. URL https://aclanthology.org/D19-5411.

Matthias Nitsche. Towards German Abstractive Text Summarization using Deep Learning. Master's thesis, Hochschule für angewandte Wissenschaften Hamburg, 2019.

Rodrigo Frassetto Nogueira and Kyunghyun Cho. Passage Re-ranking with BERT. *CoRR*, abs/1901.04085, 2019. URL http://arxiv.org/abs/1901.04085.

Curtis G. Northcutt, Anish Athalye, and Jonas Mueller. Pervasive Label Errors in Test Sets Destabilize Machine Learning Benchmarks. In Joaquin Vanschoren and Sai-Kit Yeung, editors, *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/hash/f2217062e9a397a1dca429e7d70bc6ca-Abstract-round1.html.

OpenAI. GPT-4 technical report. *CoRR*, abs/2303.08774, 2023. URL https://doi.org/10.48550/arXiv.2303.08774.

Paul Over, Hoa Dang, and Donna Harman. DUC in Context. *Information Processing & Management*, 43(6):1506–1520, 2007. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2007.01.019. URL `https://www.sciencedirect.com/science/article/pii/S0306457307000404`. Text Summarization.

Gustavo Paetzold, Fernando Alva-Manchego, and Lucia Specia. MASSAlign: Alignment and Annotation of Comparable Documents. In *Proceedings of the IJCNLP 2017, System Demonstrations*, pages 1–4, Tapei, Taiwan, November 2017. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/I17-3001`.

Gustavo H. Paetzold and Lucia Specia. A Survey on Lexical Simplification. *Journal of Artificial Intelligence Research*, 60:549–593, 2017.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank Citation Ranking: Bring Order to the Web. Technical report, Technical report, Stanford University, 1998.

Artidoro Pagnoni, Vidhisha Balachandran, and Yulia Tsvetkov. Understanding Factuality in Abstractive Summarization with FRANK: A Benchmark for Factuality Metrics. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4812–4829, Online, June 2021. Association for Computational Linguistics. URL `https://aclanthology.org/2021.naacl-main.383`.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL `https://www.aclweb.org/anthology/P02-1040`.

Andrei Paraschiv and Dumitru-Clementin Cercel. UPB at germeval-2020 task 3: Assessing summaries for german texts using bertscore and sentence-bert. In Sarah Ebling, Don Tuggener, Manuela Hürlimann, Mark Cieliebak, and Martin Volk, editors, *Proceedings of the 5th Swiss Text Analytics Conference and the 16th Conference on Natural Language Processing, SwissText/KONVENS 2020, Zurich, Switzerland, June 23-25, 2020 [online only]*, volume 2624 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2020. URL `http://ceur-ws.org/Vol-2624/germeval-task3-paper2.pdf`.

Shantipriya Parida and Petr Motlícek. Idiap Abstract Text Summarization System for German Text Summarization Task. In Mark Cieliebak, Don Tuggener, and Fernando Benites, editors, *Proceedings of the 4th Swiss Text Analytics Conference, SwissText 2019, Winterthur, Switzerland, June 18-19, 2019*, volume 2458 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2019. URL `http://ceur-ws.org/Vol-2458/paper9.pdf`.

Ankur Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqui, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. ToTTo: A Controlled Table-To-Text Generation Dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1173–1186, Online, November 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.emnlp-main.89.

Daraksha Parveen, Hans-Martin Ramsl, and Michael Strube. Topical Coherence for Graph-based Extractive Summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1949–1954, Lisbon, Portugal, September 2015. Association for Computational Linguistics. URL https://aclanthology.org/D15-1226.

Razvan Pascanu, Tomás Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pages 1310–1318. JMLR.org, 2013. URL http://proceedings.mlr.press/v28/pascanu13.html.

Laura Perez-Beltrachini and Mirella Lapata. Models and Datasets for Cross-Lingual Summarisation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9408–9423, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.emnlp-main.742.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep Contextualized Word Representations. In Marilyn Walker, Heng Ji, and Amanda Stent, editors, *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. URL https://aclanthology.org/N18-1202.

Jason Phang, Yao Zhao, and Peter Liu. Investigating Efficiently Extending Transformers for Long Input Summarization. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3946–3961, Singapore, December 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.emnlp-main.240.

Jakub Piskorski, Vanni Zavarella, Martin Atkinson, and Marco Verile. Timelines: Entity-centric Event Extraction from Online News. In Ricardo Campos, Alípio Mário Jorge, Adam Jatowt, and Sumit Bhatia, editors, *Proceedings of Text2Story - Third Workshop on Narrative Extraction From Texts co-located with 42nd European Conference on Information Retrieval, Text2Story@ECIR 2020, Lisbon, Portugal, April 14th, 2020 [online only]*, volume 2593 of *CEUR Workshop Proceedings*, pages 105–114. CEUR-WS.org, 2020. URL https://ceur-ws.org/Vol-2593/paper13.pdf.

Emily Pitler and Ani Nenkova. Revisiting Readability: A Unified Framework for Predicting Text Quality. In Mirella Lapata and Hwee Tou Ng, editors, *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 186–195, Honolulu, Hawaii, October 2008. Association for Computational Linguistics. URL https://aclanthology.org/D08-1020.

Barbara Plank. The "Problem" of Human Label Variation: On Ground Truth in Data, Modeling and Evaluation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10671–10682, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.emnlp-main.731.

Jason Portenoy and Jevin D West. Constructing and evaluating automated literature review systems. *Scientometrics*, 125(3):3233–3251, 2020.

James Pustejovsky, Patrick Hanks, Roser Sauri, Andrew See, Robert Gaizauskas, Andrea Setzer, Dragomir R. Radev, Beth Sundheim, David Day, Lisa Ferro, et al. The TIMEBANK Corpus. In *Corpus Linguistics*, volume 2003, pages 647–656. Lancaster, UK, 2003.

Avinesh P.V.S and Christian M. Meyer. Joint Optimization of User-desired Content in Multi-document Summaries by Learning from User Feedback. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1353–1363, Vancouver, Canada, July 2017. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P17-1124.

Dragomir R. Radev, Sasha Blair-Goldensohn, and Zhu Zhang. Experiments in Single and Multi-Document Summarization Using MEAD. In *First Document Understanding Conference*, pages 1–7, 2001.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving Language Understanding by Generative Pre-Training. 2018.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Resesarch*, 21:140:1–140:67, 2020. URL http://jmlr.org/papers/v21/20-074.html.

George J. Rath, Albert Resnick, and Terry R. Savage. The formation of abstracts by the selection of sentences. Part I. Sentence selection by men and machines. *American Documentation*, 12(2): 139–141, 1961. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/asi.5090120210.

Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, and Stefan Carlsson. CNN features off-the-shelf: An astounding baseline for recognition. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2014, Columbus, OH, USA, June 23-28, 2014*, pages 512–519. IEEE Computer Society, 2014. doi: 10.1109/CVPRW.2014.131.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/D19-1410.

Nils Reimers and Iryna Gurevych. Making Monolingual Sentence Embeddings Multilingual using Knowledge Distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online, November 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.emnlp-main.365.

Norbert Reithinger, Michael Kipp, Ralf Engel, and Jan Alexandersson. Summarizing Multilingual Spoken Negotiation Dialogues. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 310–317, Hong Kong, October 2000. Association for Computational Linguistics. doi: 10.3115/1075218.1075258. URL https://aclanthology.org/P00-1040.

Albert Resnick. Part II. The reliability of people in selecting sentences. *American Documentation*, 12(2):141–143, 1961. doi: https://doi.org/10.1002/asi.5090120211. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/asi.5090120211.

Albert Resnick and Terry R Savage. A Re-Evaluation of Machine-Generated Abstracts. *Human Factors*, 2(3):141–146, 1960. URL https://doi.org/10.1177/001872086000200305.

Stefan Riezler and John T. Maxwell. On Some Pitfalls in Automatic Evaluation and Significance Testing for MT. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 57–64, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. URL https://aclanthology.org/W05-0908.

Stephen E. Robertson and Karen Spärck Jones. Relevance Weighting of Search Terms. *Journal of the American Society for Information Science*, 27(3):129–146, 1976. doi: 10.1002/asi.4630270302. URL https://doi.org/10.1002/asi.4630270302.

Alexander M. Rush, Sumit Chopra, and Jason Weston. A Neural Attention Model for Abstractive Sentence Summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal, September 2015. Association for Computational Linguistics. URL https://aclanthology.org/D15-1044.

Michael Ryan, Tarek Naous, and Wei Xu. Revisiting non-English Text Simplification: A Unified Multilingual Benchmark. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4898–4927, Toronto, Canada, July 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.acl-long.269.

Horacio Saggion, Dragomir R. Radev, Simone Teufel, Wai Lam, and Stephanie M. Strassel. Developing Infrastructure for the Evaluation of Single and Multi-document Summarization Systems in a Cross-lingual Environment. In *Proceedings of the Third International Conference on Language Resources and Evaluation, LREC 2002, May 29-31, 2002, Las Palmas, Canary Islands, Spain*. European Language Resources Association, 2002. URL http://www.lrec-conf.org/proceedings/lrec2002/sumarios/158.htm.

Horacio Saggion, Sanja Štajner, Daniel Ferrés, Kim Cheng Sheang, Matthew Shardlow, Kai North, and Marcos Zampieri. Findings of the TSAR-2022 Shared Task on Multilingual Lexical Simplification. In Sanja Štajner, Horacio Saggion, Daniel Ferrés, Matthew Shardlow, Kim Cheng Sheang, Kai North, Marcos Zampieri, and Wei Xu, editors, *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 271–283, Abu Dhabi, United Arab Emirates (Virtual), December 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.tsar-1.31.

Itsumi Saito, Kyosuke Nishida, Kosuke Nishida, and Junji Tomita. Abstractive Summarization with Combination of Pre-trained Sequence-to-Sequence and Saliency Models. *CoRR*, abs/2003.13028, 2020. URL https://arxiv.org/abs/2003.13028.

Evan Sandhaus. The New York Times Annotated Corpus, LDC2008T19, 2008.

Aleksandar Savkov, Francesco Moramarco, Alex Papadopoulos Korfiatis, Mark Perera, Anya Belz, and Ehud Reiter. Consultation Checklists: Standardising the Human Evaluation of Medical Note Generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 111–120, Abu Dhabi, UAE, December 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.emnlp-industry.10.

Christopher Scanlan. *Reporting and Writing: Basics for the 21st Century*. Harcourt College Publishers, 2000.

Frank Schilder and Christopher Habel. From Temporal Expressions To Temporal Information: Semantic Tagging Of News Messages. In *Proceedings of the ACL 2001 Workshop on Temporal and Spatial Information Processing*, 2001. URL https://aclanthology.org/W01-1309.

Roy Schubiger. German Summarization with Large Language Models. Master's thesis, ETH Zurich, 2024.

Michael Schudson. The News Media as Political Institutions. *Annual Review of Political Science*, 5(1):249–269, 2002. doi: 10.1146/annurev.polisci.5.111201.115816. URL https://doi.org/10.1146/annurev.polisci.5.111201.115816.

Michael Schulte and Ziko van Dijk. Free Children's Encyclopedia Project. 2015.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, and Jacopo Staiano. MLSUM: The Multilingual Summarization Corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8051–8067, Online, November 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.emnlp-main.647.

Abigail See, Peter J. Liu, and Christopher D. Manning. Get To The Point: Summarization with Pointer-Generator Networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada, July 2017. Association for Computational Linguistics. URL https://aclanthology.org/P17-1099.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. BLEURT: Learning Robust Metrics for Text Generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online, July 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.acl-main.704.

Christian Sengstock, Michael Gertz, and Tran Van Canh. Spatial Interestingness Measures for Co-location Pattern Mining. In Jilles Vreeken, Charles Ling, Mohammed Javeed Zaki, Arno Siebes, Jeffrey Xu Yu, Bart Goethals, Geoffrey I. Webb, and Xindong Wu, editors, *12th IEEE International Conference on Data Mining Workshops, ICDM Workshops, Brussels, Belgium, December 10, 2012*, pages 821–826. IEEE Computer Society, 2012. URL https://doi.org/10.1109/ICDMW.2012.116.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany, August 2016. Association for Computational Linguistics. URL https://aclanthology.org/P16-1162.

Matthew Shardlow. A Survey of Automated Text Simplification. *International Journal of Advanced Computer Science and Applications*, 4(1):58–70, 2014.

Matthew Shardlow, Richard Evans, Gustavo H. Paetzold, and Marcos Zampieri. SemEval-2021 Task 1: Lexical Complexity Prediction. In Alexis Palmer, Nathan Schneider, Natalie Schluter, Guy Emerson, Aurelie Herbelot, and Xiaodan Zhu, editors, *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 1–16, Online, August 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.semeval-1.1.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=B1ckMDqlg.

Peng Shi and Jimmy Lin. Simple BERT Models for Relation Extraction and Semantic Role Labeling. *CoRR*, abs/1904.05255, 2019. URL http://arxiv.org/abs/1904.05255.

Advaith Siddharthan. A survey of research on text simplification. *ITL-International Journal of Applied Linguistics*, 165(2):259–298, 2014.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Kumar Tanwani, Heather Cole-Lewis, Stephen Pfohl, Perry Payne, Martin Seneviratne, Paul Gamble, Chris Kelly, Nathaneal Schärli, Aakanksha Chowdhery, Philip Andrew Mansfield, Blaise Agüera y Arcas, Dale R. Webster, Gregory S. Corrado, Yossi Matias, Katherine Chou, Juraj Gottweis, Nenad Tomasev, Yun Liu, Alvin Rajkomar, Joelle K. Barral, Christopher Semturs, Alan Karthikesalingam, and Vivek Natarajan. Large Language Models Encode Clinical Knowledge. *Nature*, 620(7972):172–180, 2023. URL https://doi.org/10.1038/s41586-023-06291-2.

Yan Song, Yuanhe Tian, Nan Wang, and Fei Xia. Summarizing Medical Conversations via Identifying Important Utterances. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 717–729, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics. URL https://aclanthology.org/2020.coling-main.63.

Sajad Sotudeh Gharebagh, Nazli Goharian, and Ross Filice. Attend to Medical Ontologies: Content Selection for Clinical Abstractive Summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1899–1905, Online, July 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.acl-main.172.

Lucia Specia, Sujay K. Jauhar, and Rada Mihalcea. SemEval-2012 Task 1: English Lexical Simplification. In Eneko Agirre, Johan Bos, Mona Diab, Suresh Manandhar, Yuval Marton, and Deniz Yuret, editors, **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 347–355, Montréal, Canada, 7-8 June 2012. Association for Computational Linguistics. URL https://aclanthology.org/S12-1046.

Andreas Spitz and Michael Gertz. Terms over LOAD: Leveraging Named Entities for Cross-Document Extraction and Summarization of Events. In Raffaele Perego, Fabrizio Sebastiani, Javed A. Aslam, Ian Ruthven, and Justin Zobel, editors, *Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval, SIGIR 2016, Pisa, Italy, July 17-21, 2016*, pages 503–512. ACM, 2016. URL https://doi.org/10.1145/2911451.2911529.

Andreas Spitz, Satya Almasian, and Michael Gertz. TopExNet: Entity-Centric Network Topic Exploration in News Streams. In J. Shane Culpepper, Alistair Moffat, Paul N. Bennett, and Kristina Lerman, editors, *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, WSDM 2019, Melbourne, VIC, Australia, February 11-15, 2019*, pages 798–801. ACM, 2019. URL https://doi.org/10.1145/3289600.3290619.

Andreas Spitz, Dennis Aumiller, Bálint Soproni, and Michael Gertz. A Versatile Hypergraph Model for Document Collections. In Elaheh Pourabbas, Dimitris Sacharidis, Kurt Stockinger, and Thanasis Vergoulis, editors, *SSDBM 2020: 32nd International Conference on Scientific and Statistical Database Management, Vienna, Austria, July 7-9, 2020*, pages 7:1–7:12. ACM, 2020. doi: 10.1145/3400903.3400919.

Sanja Štajner, Marc Franco-Salvador, Paolo Rosso, and Simone Paolo Ponzetto. CATS: A Tool for Customized Alignment of Text Simplification Corpora. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). URL https://www.aclweb.org/anthology/L18-1615.

Julius Steen and Katja Markert. Abstractive Timeline Summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 21–31, Hong Kong, China, November 2019. Association for Computational Linguistics. URL https://aclanthology.org/D19-5403.

Julius Steen and Katja Markert. How to Evaluate a Summarizer: Study Design and Statistical Analysis for Manual Linguistic Quality Evaluation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1861–1875, Online, April 2021. Association for Computational Linguistics. URL `https://aclanthology.org/2021.eacl-main.160`.

Julius Steen and Katja Markert. How to Find Strong Summary Coherence Measures? A Toolbox and a Comparative Study for Summary Coherence Measure Evaluation. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6035–6049, Gyeongju, Republic of Korea, October 2022. International Committee on Computational Linguistics. URL `https://aclanthology.org/2022.coling-1.527`.

Josef Steinberger, Hristo Tanev, Mijail A. Kabadjov, and Ralf Steinberger. JRC's Participation in the Guided Summarization Task at TAC 2010. In *Proceedings of the Third Text Analysis Conference, TAC 2010, Gaithersburg, Maryland, USA, November 15-16, 2010*. NIST, 2010. URL `https://tac.nist.gov/publications/2010/participant.papers/JRC.proceedings.pdf`.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F. Christiano. Learning to summarize with human feedback. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 3008–3021. Curran Associates, Inc., 2020. URL `https://proceedings.neurips.cc/paper_files/paper/2020/file/1f89885d556929e98d3ef9b86448f951-Paper.pdf`.

Regina Stodden. EASSE-DE: Easier Automatic Sentence Simplification Evaluation for German. *CoRR*, abs/2404.03563, 2024. URL `https://doi.org/10.48550/arXiv.2404.03563`.

Regina Stodden, Omar Momen, and Laura Kallmeyer. DEplain: A German Parallel Corpus with Intralingual Translations into Plain Language for Sentence and Document Simplification. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16441–16463, Toronto, Canada, July 2023. Association for Computational Linguistics. URL `https://aclanthology.org/2023.acl-long.908`.

Jannik Strötgen and Michael Gertz. HeidelTime: High Quality Rule-Based Extraction and Normalization of Temporal Expressions. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 321–324, Uppsala, Sweden, July 2010. Association for Computational Linguistics. URL `https://aclanthology.org/S10-1071`.

Jannik Strötgen and Michael Gertz. A Baseline Temporal Tagger for all Languages. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 541–547, Lisbon, Portugal, September 2015. Association for Computational Linguistics. URL `https://aclanthology.org/D15-1063`.

Jianlin Su, Murtadha H. M. Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. RoFormer: Enhanced transformer with Rotary Position Embedding. *Neurocomputing*, 568: 127063, 2024. doi: 10.1016/J.NEUCOM.2023.127063.

Renliang Sun, Hanqi Jin, and Xiaojun Wan. Document-Level Text Simplification: Dataset, Criteria and Baseline. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7997–8013, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. URL `https://aclanthology.org/2021.emnlp-main.630`.

Barkavi Sundararajan, Somayajulu Sripada, and Ehud Reiter. Error Analysis of ToTTo Table-to-Text Neural NLG Models. In *Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 456–470, Abu Dhabi, United Arab Emirates (Hybrid), December 2022. Association for Computational Linguistics. URL `https://aclanthology.org/2022.gem-1.43`.

Liyan Tang, Zhaoyi Sun, Betina Idnay, Jordan G. Nestor, Ali Soroush, Pierre A. Elias, Ziyang Xu, Ying Ding, Greg Durrett, Justin Rousseau, Chunhua Weng, and Yifan Peng. Evaluating Large Language Models on Medical Evidence Summarization. *medRxiv*, 2023. doi: 10.1101/2023.04.22.23288967. URL `https://www.medrxiv.org/content/early/2023/04/24/2023.04.22.23288967`.

Xiangru Tang, Alexander Fabbri, Haoran Li, Ziming Mao, Griffin Adams, Borui Wang, Asli Celikyilmaz, Yashar Mehdad, and Dragomir Radev. Investigating Crowdsourcing Protocols for Evaluating the Factual Consistency of Summaries. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5680–5692, Seattle, United States, July 2022. Association for Computational Linguistics. URL `https://aclanthology.org/2022.naacl-main.417`.

Teerapaun Tanprasert and David Kauchak. Flesch-Kincaid is Not a Text Simplification Evaluation Metric. In Antoine Bosselut, Esin Durmus, Varun P. Gangal, Sebastian Gehrmann, Yacine Jernite, Laura Perez-Beltrachini, Samira Shaikh, and Wei Xu, editors, *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021)*, pages 1–14, Online, August 2021. Association for Computational Linguistics. URL `https://aclanthology.org/2021.gem-1.1`.

Maartje Ter Hoeve, Julia Kiseleva, and Maarten Rijke. What Makes a Good and Useful Summary? Incorporating Users in Automatic Summarization Research. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 46–75, Seattle, United States, July 2022. Association for Computational Linguistics. URL `https://aclanthology.org/2022.naacl-main.4`.

Brian Thompson and Matt Post. Automatic Machine Translation Evaluation in Many Languages via Zero-Shot Paraphrasing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 90–121, Online, November 2020. Association for Computational Linguistics. URL `https://aclanthology.org/2020.emnlp-main.8`.

Craig Thomson and Ehud Reiter. A Gold Standard Methodology for Evaluating Accuracy in Data-To-Text Systems. In *Proceedings of the 13th International Conference on Natural Lan-*

guage Generation, pages 158–168, Dublin, Ireland, December 2020. Association for Computational Linguistics. URL `https://aclanthology.org/2020.inlg-1.22`.

Craig Thomson, Ehud Reiter, and Barkavi Sundararajan. Evaluating factual accuracy in complex data-to-text. *Computer Speech & Language*, 80:101482, 2023. ISSN 0885-2308. URL `https://www.sciencedirect.com/science/article/pii/S0885230823000013`.

Jörg Tiedemann and Santhosh Thottingal. OPUS-MT – Building open translation services for the World. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 479–480, Lisboa, Portugal, November 2020. European Association for Machine Translation. URL `https://aclanthology.org/2020.eamt-1.61`.

Ingrid Torjesen. Doctors are under more work pressure than during height of covid-19 pandemic in 2020. *BMJ*, 375, 2021. doi: 10.1136/bmj.n3088. URL `https://www.bmj.com/content/375/bmj.n3088`.

Jan Trienes, Jörg Schlötterer, Hans-Ulrich Schildhaus, and Christin Seifert. Patient-friendly Clinical Notes: Towards a new Text Simplification Dataset. In *Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022)*, pages 19–27, Abu Dhabi, United Arab Emirates (Virtual), December 2022. Association for Computational Linguistics. URL `https://aclanthology.org/2022.tsar-1.3`.

Andrew Trotman, Antti Puurula, and Blake Burgess. Improvements to BM25 and Language Models Examined. In *Proceedings of the 19th Australasian Document Computing Symposium*, ADCS '14, pages 58–65, New York, NY, USA, 2014. Association for Computing Machinery. ISBN 9781450330008. URL `https://doi.org/10.1145/2682862.2682863`.

Ashok Urlana, Nirmal Surange, Pavan Baswani, Priyanka Ravva, and Manish Shrivastava. TeSum: Human-Generated Abstractive Summarization Corpus for Telugu. In *Proceedings of the Language Resources and Evaluation Conference*, pages 5712–5722, Marseille, France, June 2022. European Language Resources Association. URL `https://aclanthology.org/2022.lrec-1.614`.

Ashok Urlana, Pinzhen Chen, Zheng Zhao, Shay Cohen, Manish Shrivastava, and Barry Haddow. PMIndiaSum: Multilingual and Cross-lingual Headline Summarization for Languages in India. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11606–11628, Singapore, December 2023a. Association for Computational Linguistics. URL `https://aclanthology.org/2023.findings-emnlp.777`.

Ashok Urlana, Pruthwik Mishra, Tathagato Roy, and Rahul Mishra. Controllable Text Summarization: Unraveling Challenges, Approaches, and Prospects - A Survey. *CoRR*, abs/2311.09212, 2023b. doi: 10.48550/ARXIV.2311.09212. URL `https://doi.org/10.48550/arXiv.2311.09212`.

Rafaella F. Vale, Rafael Dueire Lins, and Rafael Ferreira. An Assessment of Sentence Simplification Methods in Extractive Text Summarization. In *DocEng '20: ACM Symposium on Docu-*

ment Engineering 2020, Virtual Event, CA, USA, September 29 - October 1, 2020, pages 9:1–9:9. ACM, 2020. URL https://doi.org/10.1145/3395027.3419588.

Marc Van Opijnen and Cristiana Santos. On the Concept of Relevance in Legal Information Retrieval. *Artificial Intelligence and Law*, 25:65–87, 2017.

Lucy Vanderwende, Hisami Suzuki, Chris Brockett, and Ani Nenkova. Beyond SumBasic: Task-focused summarization with sentence simplification and lexical expansion. *Information Processing & Management*, 43(6):1606–1618, 2007. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2007.01.023.

Daniel Varab and Natalie Schluter. MassiveSumm: a very large-scale, very multilingual, news summarisation dataset. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10150–10161, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.emnlp-main.797.

Laura Vásquez-Rodríguez, Matthew Shardlow, Piotr Przybyła, and Sophia Ananiadou. Investigating Text Simplification Evaluation. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 876–882, Online, August 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.findings-acl.77.

Laura Vásquez-Rodríguez, Matthew Shardlow, Piotr Przybyła, and Sophia Ananiadou. Document-level Text Simplification with Coherence Evaluation. In Sanja Štajner, Horacio Saggio, Matthew Shardlow, and Fernando Alva-Manchego, editors, *Proceedings of the Second Workshop on Text Simplification, Accessibility and Readability*, pages 85–101, Varna, Bulgaria, September 2023. INCOMA Ltd., Shoumen, Bulgaria. URL https://aclanthology.org/2023.tsar-1.9.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html.

Valentin Venzin, Jan Deriu, Didier Orel, and Mark Cieliebak. Fact-aware Abstractive Text Summarization using a Pointer-Generator Network. In Mark Cieliebak, Don Tuggener, and Fernando Benites, editors, *Proceedings of the 4th Swiss Text Analytics Conference, SwissText 2019, Winterthur, Switzerland, June 18-19, 2019*, volume 2458 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2019. URL http://ceur-ws.org/Vol-2458/paper7.pdf.

Suzan Verberne, Evangelos Kanoulas, Gineke Wiggers, Florina Piroi, and Arjen P. de Vries. ECIR 2023 Workshop: Legal Information Retrieval. In *Advances in Information Retrieval: 45th European Conference on Information Retrieval, ECIR 2023, Dublin, Ireland, April 2–6, 2023,* *Proceedings, Part III*, pages 412–419, Berlin, Heidelberg, 2023. Springer-Verlag. ISBN 978-3-031-28240-9. URL https://doi.org/10.1007/978-3-031-28241-6_46.

Veniamin Veselovsky, Manoel Horta Ribeiro, and Robert West. Artificial Artificial Artificial Intelligence: Crowd Workers Widely Use Large Language Models for Text Production Tasks. *CoRR*, abs/2306.07899, 2023. URL https://doi.org/10.48550/arXiv.2306.07899.

Jesse Vig, Alexander Fabbri, Wojciech Kryscinski, Chien-Sheng Wu, and Wenhao Liu. Exploring Neural Models for Query-Focused Summarization. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1455–1468, Seattle, United States, July 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.findings-naacl.109.

Eric Wallace, Yizhong Wang, Sujian Li, Sameer Singh, and Matt Gardner. Do NLP Models Know Numbers? Probing Numeracy in Embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5307–5315, Hong Kong, China, November 2019. Association for Computational Linguistics. URL https://aclanthology.org/D19-1534.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019. URL https://openreview.net/forum?id=rJ4km2R5t7.

Alex Wang, Kyunghyun Cho, and Mike Lewis. Asking and Answering Questions to Evaluate the Factual Consistency of Summaries. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5008–5020, Online, July 2020. Association for Computational Linguistics. URL https://aclanthology.org/2020.acl-main.450.

Alex Wang, Richard Yuanzhe Pang, Angelica Chen, Jason Phang, and Samuel R. Bowman. SQuALITY: Building a Long-Document Summarization Dataset the Hard Way. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1139–1156, Abu Dhabi, United Arab Emirates, December 2022a. Association for Computational Linguistics. URL https://aclanthology.org/2022.emnlp-main.75.

Danqing Wang, Jiaze Chen, Hao Zhou, Xipeng Qiu, and Lei Li. Contrastive Aligned Joint Learning for Multilingual Summarization. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2739–2750, Online, August 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.findings-acl.242.

Jiaan Wang, Fandong Meng, Ziyao Lu, Duo Zheng, Zhixu Li, Jianfeng Qu, and Jie Zhou. ClidSum: A Benchmark Dataset for Cross-Lingual Dialogue Summarization. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7716–7729, Abu Dhabi, United Arab Emirates, December 2022b. Association for Computational Linguistics. URL https://aclanthology.org/2022.emnlp-main.526.

Jiaan Wang, Fandong Meng, Duo Zheng, Yunlong Liang, Zhixu Li, Jianfeng Qu, and Jie Zhou. A Survey on Cross-Lingual Summarization. *Transactions of the Association for Computational Linguistics*, 10:1304–1323, 11 2022c. ISSN 2307-387X. URL `https://doi.org/10.1162/tacl_a_00520`.

Ye Wang, Yingmin Zhou, Mengzhu Wang, Zhenghan Chen, Zhiping Cai, Junyang Chen, and Victor C. M. Leung. Multidocument Aspect Classification for Aspect-Based Abstractive Summarization. *IEEE Transactions on Computational Social Systems*, pages 1–10, 2023. doi: 10.1109/TCSS.2023.3252723.

Zarah Weiß and Detmar Meurers. Modeling the Readability of German Targeting Adults and Children: An empirically broad analysis and its cross-corpus validation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 303–317, Santa Fe, New Mexico, USA, August 2018. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/C18-1026`.

Leonie Weissweiler and Alexander Fraser. Developing a Stemmer for German Based on a Comparative Analysis of Publicly Available Stemmers. In Georg Rehm and Thierry Declerck, editors, *Language Technologies for the Challenges of the Digital Age - 27th International Conference, GSCL 2017, Berlin, Germany, September 13-14, 2017, Proceedings*, volume 10713 of *Lecture Notes in Computer Science*, pages 81–94. Springer, 2017. doi: 10.1007/978-3-319-73706-5_8.

Thompson Ramo Woolridge Inc. Final Report on the Study of Automatic Abstracting. Technical Report C107-1U12, Thompson Ramo Woolridge Computer Division, 1961.

Yuexiang Xie, Fei Sun, Yang Deng, Yaliang Li, and Bolin Ding. Factual Consistency Evaluation for Text Summarization via Counterfactual Estimation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 100–110, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. URL `https://aclanthology.org/2021.findings-emnlp.10`.

Siheng Xiong, Ali Payani, Ramana Kompella, and Faramarz Fekri. Large Language Models Can Learn Temporal Reasoning. *CoRR*, abs/2401.06853, 2024. URL `https://doi.org/10.48550/arXiv.2401.06853`.

Huihui Xu, Jaromír Savelka, and Kevin D. Ashley. Toward Summarizing Case Decisions via Extracting Argument Issues, Reasons, and Conclusions. In Juliano Maranhão and Adam Zachary Wyner, editors, *ICAIL '21: Eighteenth International Conference for Artificial Intelligence and Law, São Paulo Brazil, June 21 - 25, 2021*, pages 250–254. ACM, 2021a. URL `https://doi.org/10.1145/3462757.3466098`.

Kun Xu, Han Wu, Linfeng Song, Haisong Zhang, Linqi Song, and Dong Yu. Conversational Semantic Role Labeling. *IEEE ACM Transactions on Audio, Speech, and Language Processing*, 29:2465–2475, 2021b. doi: 10.1109/TASLP.2021.3074014.

Wei Xu, Chris Callison-Burch, and Courtney Napoles. Problems in Current Text Simplification Research: New Data Can Help. *Transactions of the Association for Computational Linguistics*, 3:283–297, 2015. doi: 10.1162/tacl_a_00139. URL `https://aclanthology.org/Q15-1021`.

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. Optimizing Statistical Machine Translation for Text Simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415, 2016. doi: 10.1162/tacl_a_00107. URL https://www.aclweb.org/anthology/Q16-1029.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online, June 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.naacl-main.41.

Liu Yang, Mingyang Zhang, Cheng Li, Michael Bendersky, and Marc Najork. Beyond 512 Tokens: Siamese Multi-Depth Transformer-Based Hierarchical Encoder for Long-Form Document Matching. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*, CIKM '20, pages 1725–1734, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450368599. URL https://doi.org/10.1145/3340531.3411908.

Xianjun Yang, Yan Li, Xinlu Zhang, Haifeng Chen, and Wei Cheng. Exploring the Limits of ChatGPT for Query or Aspect-based Text Summarization. *CoRR*, abs/2302.08081, 2023. URL https://doi.org/10.48550/arXiv.2302.08081.

Szu-ting Yi, Edward Loper, and Martha Palmer. Can Semantic Roles Generalize Across Genres? In *Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 548–555. Association for Computational Linguistics, 2007. URL https://aclanthology.org/N07-1069/.

Yi Yu, Adam Jatowt, Antoine Doucet, Kazunari Sugiyama, and Masatoshi Yoshikawa. Multi-TimeLine Summarization (MTLS): Improving Timeline Summarization by Generating Multiple Summaries. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 377–387, Online, August 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.acl-long.32.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. BARTScore: Evaluating Generated Text as Text Generation. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 27263–27277, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/e4d2b6e6fdeca3e60e0f1a62fee3d9dd-Abstract.html.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontañón, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. Big Bird: Transformers for Longer Sequences. In *Advances in Neural Information Processing Systems 33:*

Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual, 2020.

Fangfang Zhang, Jin-ge Yao, and Rui Yan. On the Abstractiveness of Neural Document Summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 785–790, Brussels, Belgium, October-November 2018a. Association for Computational Linguistics. URL https://aclanthology.org/D18-1089.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. PEGASUS: pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 11328–11339. PMLR, 2020a. URL http://proceedings.mlr.press/v119/zhang20ae.html.

Mike Zhang and Antonio Toral. The Effect of Translationese in Machine Translation Test Sets. In *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 73–81, Florence, Italy, August 2019. Association for Computational Linguistics. URL https://aclanthology.org/W19-5208.

Shiyue Zhang, David Wan, and Mohit Bansal. Extractive is not Faithful: An Investigation of Broad Unfaithfulness Problems in Extractive Summarization. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2153–2174, Toronto, Canada, July 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.acl-long.120.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. BERTScore: Evaluating Text Generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020b. URL https://openreview.net/forum?id=SkeHuCVFDr.

Yuhao Zhang, Daisy Yi Ding, Tianpei Qian, Christopher D. Manning, and Curtis P. Langlotz. Learning to Summarize Radiology Findings. In *Proceedings of the Ninth International Workshop on Health Text Mining and Information Analysis*, pages 204–213, Brussels, Belgium, October 2018b. Association for Computational Linguistics. URL https://aclanthology.org/W18-5623.

Yuhao Zhang, Derek Merck, Emily Tsai, Christopher D. Manning, and Curtis Langlotz. Optimizing the Factual Correctness of a Summary: A Study of Summarizing Radiology Reports. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5108–5120, Online, July 2020c. Association for Computational Linguistics. URL https://aclanthology.org/2020.acl-main.458.

Shaohui Zheng, Zhixu Li, Jiaan Wang, Jianfeng Qu, An Liu, Lei Zhao, and Zhigang Chen. Long-Document Cross-Lingual Summarization. In Tat-Seng Chua, Hady W. Lauw, Luo Si, Evimaria

Terzi, and Panayiotis Tsaparas, editors, *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining, WSDM 2023, Singapore, 27 February 2023 - 3 March 2023*, pages 1084–1092. ACM, 2023. URL `https://doi.org/10.1145/3539597.3570479`.

Ming Zhong, Danqing Wang, Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. A Closer Look at Data Bias in Neural Extractive Summarization Models. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 80–89, Hong Kong, China, November 2019. Association for Computational Linguistics. URL `https://aclanthology.org/D19-5410`.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-Following Evaluation for Large Language Models. *CoRR*, abs/2311.07911, 2023. URL `https://doi.org/10.48550/arXiv.2311.07911`.

Chenguang Zhu, William Hinthorn, Ruochen Xu, Qingkai Zeng, Michael Zeng, Xuedong Huang, and Meng Jiang. Enhancing Factual Consistency of Abstractive Summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 718–733, Online, June 2021a. Association for Computational Linguistics. URL `https://aclanthology.org/2021.naacl-main.58`.

Chenguang Zhu, Ziyi Yang, Robert Gmyr, Michael Zeng, and Xuedong Huang. Leveraging Lead Bias for Zero-shot Abstractive News Summarization. In Fernando Diaz, Chirag Shah, Torsten Suel, Pablo Castells, Rosie Jones, and Tetsuya Sakai, editors, *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, pages 1462–1471. ACM, 2021b. doi: 10.1145/3404835.3462846.

Junnan Zhu, Qian Wang, Yining Wang, Yu Zhou, Jiajun Zhang, Shaonan Wang, and Chengqing Zong. NCLS: Neural Cross-Lingual Summarization. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3054–3064, Hong Kong, China, November 2019. Association for Computational Linguistics. URL `https://aclanthology.org/D19-1302`.

Zhemin Zhu, Delphine Bernhard, and Iryna Gurevych. A Monolingual Tree-based Translation Model for Sentence Simplification. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 1353–1361, Beijing, China, August 2010. Coling 2010 Organizing Committee. URL `https://aclanthology.org/C10-1152`.

John Ziegler, Alexander Brand, Julian Freyberg, Tim König, Wolf J. Schünemann, Marina Walther, and Michael Gertz. EPINetz: Exploration of Political Information Networks. In Gesellschaft für Informatik, editor, *51. Jahrestagung der Gesellschaft für Informatik, INFORMATIK 2021 - Computer Science & Sustainability, Berlin, Germany, 27. September - 1. Oktober, 2021*, volume P-314 of *LNI*, pages 1603–1609. Gesellschaft für Informatik, Bonn, 2021. URL `https://doi.org/10.18420/informatik2021-134`.

Markus Zopf. Auto-hMDS: Automatic Construction of a Large Heterogeneous Multilingual Multi-Document Summarization Corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 2018. European Language Resources Association (ELRA). URL https://aclanthology.org/L18-1510.

Markus Zopf, Maxime Peyrard, and Judith Eckle-Kohler. The Next Step for Multi-Document Summarization: A Heterogeneous Multi-Genre Corpus Built with a Novel Construction Approach. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1535–1545, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee. URL https://aclanthology.org/C16-1145.