Translation-based Ranking in Cross-Language Information Retrieval

Dezember 2014

Dissertation
zur Erlangung der Doktorwürde
der Neuphilologischen Fakultät
der Ruprecht-Karls-Universität Heidelberg

vorgelegt
von

Felix Hieber

Institut für Computerlinguistik
Ruprecht-Karls-Universität Heidelberg

Betreuer und Erstgutachter:	Prof. Dr. Stefan Riezler Institut für Computerlinguistik Ruprecht-Karls-Universität Heidelberg
Zweitgutachter:	Prof. Dr. Kurt Eberle Lingenio GmbH Karlsruher Str. 10 69126 Heidelberg
Datum der Einreichung:	18. Dezember 2014
Datum der Disputation:	23. April 2015

Abstract

Today's amount of user-generated, multilingual textual data generates the necessity for information processing systems, where cross-linguality, i.e the ability to work on more than one language, is fully integrated into the underlying models. In the particular context of Information Retrieval (IR), this amounts to rank and retrieve relevant documents from a large repository in language A, given a user's information need expressed in a query in language B. This kind of application is commonly termed a *Cross-Language Information Retrieval* (CLIR) system. Such CLIR systems typically involve a translation component of varying complexity, which is responsible for translating the user input into the document language. Using query translations from modern, phrase-based *Statistical Machine Translation* (SMT) systems, and subsequently retrieving monolingually is thus a straightforward choice. However, the amount of work committed to integrate such SMT models into CLIR, or even jointly model translation and retrieval, is rather small.

In this thesis, I focus on the shared aspect of ranking in translation-based CLIR: Both, translation and retrieval models, induce rankings over a set of candidate structures through assignment of scores. The subject of this thesis is to exploit this commonality in three different ranking tasks: (1) "Mate-ranking" refers to the task of mining comparable data for SMT domain adaptation through translation-based CLIR. "Cross-lingual mates" are direct or close translations of the query. I will show that such a CLIR system is able to find in-domain comparable data from noisy user-generated corpora and improves in-domain translation performance of an SMT system. Conversely, the CLIR system relies itself on a translation model that is tailored for retrieval. This leads to the second direction of research, in which I develop two ways to optimize an SMT model for retrieval, namely (2) by SMT parameter optimization towards a retrieval objective ("translation ranking"), and (3) by presenting a joint model of translation and retrieval for "document ranking". The latter abandons the common architecture of modeling both components separately. The former task refers to optimizing for preference of translation candidates that work well for retrieval. In the core task of "document ranking" for CLIR, I present a model that directly ranks documents using an SMT decoder. I present substantial improvements over state-of-the-art translation-based CLIR baseline systems, indicating that a joint model of translation and retrieval is a promising direction of research in the field of CLIR.

Kurzfassung

Die Menge an mehrsprachigen, benutzergenerierten Textdaten erzeugt zunehmend einen Bedarf an informationsverarbeitenden Systemen, in denen eine sprachenübergreifende Verarbeitung vollständig in den zugrundeliegenden Modellen integriert ist. Im Kontext der Suche von Textdokumenten, im Folgenden Information Retrieval (IR) genannt, bedeutet dies die Erzeugung eines Rankings über Dokumente in Sprache A, gegeben dem Informationsbedürfnis eines Anwenders, formuliert in Sprache B. Ein solches *Cross-Language-Information-Retrieval*-System (CLIR) besteht typischerweise aus zwei Komponenten: Einem statistischen, maschinellen Übersetzungssystem, das Suchanfragen des Anwenders übersetzt, und einem Suchmodell, das für das Ranking der Dokumente in der Zielsprache zuständig ist.

Die vorliegende Dissertation beschäftigt sich mit Ranking in CLIR-Systemen, einerseits dem des Suchmodells, und andererseits dem des Übersetzungsmodells, *Statistical Machine Translation* (SMT). Ich nähere mich diesem Thema daher über drei Anwendungsverfahren. (1) "Mate-Ranking" bezeichnet die Aufgabe eines CLIR-Systems direkte oder vergleichbare Übersetzungen der Suchanfrage ("cross-lingual mates") in der Dokumentsammlung zu finden. Paare aus Suchanfragen und "mates" können als zusätzliche Trainingsdaten für ein SMT-Modell verwendet werden, mit dem die Übersetzungsfähigkeit in der Domäne der Dokumente angepasst werden kann (Domain Adaptation). Da ein derartig angepasstes Übersetzungssystem wieder im Rahmen eines CLIR-Systems eingesetzt werden kann, ergibt sich eine gegenseitige Abhängigkeit von SMT- und CLIR-Modell, die ein effizienteres und iteratives Domain-Adaptation-Verfahren ermöglicht. (2) Im "Translation-Ranking" geht es darum, das Ranking der von einem SMT-System erstellten Übersetzungshypothesen für das anschließende Retrieval zu optimieren. Hierbei wird im diskriminativen Training des statistischen Modells anstatt einer Übersetzungsmetrik, eine Suchmetrik als Zielfunktion verwendet. (3) Im Verfahren des "Document-Rankings" geht es um die Optimierung der Suchergebnisse eines CLIR-Systems. Es wird ein Modell vorgestellt, das Übersetzung und Suche gemeinsam modelliert: Der Dekodierprozess der Übersetzung erzeugt gleichzeitig ein Ranking über die Dokumente in der Zielsprache. Durch eine gemeinsame Modellierung beider Komponenten können Übersetzung und Suche, gleichzeitig mit bekannten Learning-to-Rank-Methoden optimiert werden. Ergebnisse dieses Modells auf zwei großen Korpora zeigen signifikante Verbesserungen gegenüber CLIR-Modellen mit der oben beschriebenen, hintereinandergeschalteten Zwei-Komponenten-Architektur.

Table of Contents

Chapter 1

Introduction

Knowledge is power. Information is liberating. Education is the premise of progress, in every society, in every family.

Kofi Annan, June 1997

1.1 Cross-Lingual Information Access in Modern Society

In today's society of information, the ability to search and retrieve any kind of information quickly and reliably is of significant economical and cultural value. To satisfy our various *information needs*, be them navigational, informational, or transactional (Broder, 2002), we rely mostly on Information Retrieval (IR) systems that enable us to express our information need in form of a textual query. They almost instantaneously deliver us an ordered list of documents as search results, of which the system is confident that they are relevant to our query. From this ordered list of search results we demand two major properties: *precision*, namely the fact that included documents are relevant to our information need, and *recall*, the property that all or most of the available, relevant information is included. More mathematically, the system corresponds to a function that induces a *ranking* over a set of textual entities, given some input text. In text-based information retrieval, these entities correspond to documents in a larger collection, the input constitutes a written search query issued by some user, and the position of a document in the ranking is determined by how well it matches the query.

Besides searching in (structured) databases, such as library catalogs for example, web search has become the most prominent type of Information Retrieval for navigational, informational or transactional information needs. Internet access, and thus access to its major search engines, has been regarded as a basic right in Western and Asian societies. Not just since United Nation's special rapporteur on freedom of expression, Frank La Rue, declared that "there should be as little restriction as possible to the flow of information via the Internet, except in a few, very exceptional, and limited circumstances prescribed by international human rights law" (LaRue, 2011), Internet access in third world countries is actively developed and promoted both for commercial reasons (e.g. Google's Project Link[1]), and by non-profit organizations, such as *A Human Right*[2].

The growing number of Internet users[3] entails that online information sources, be them commercial, political, or personal, are inherently multilingual. As more and more people contribute to the web, textual content is increasingly written in multiple languages. This means that information available in some language community may

[1] `http://www.google.com/get/projectlink/`. last access: April 26, 2015

[2] `http://ahumanright.org`. last access: April 26, 2015

[3] `http://www.internetworldstats.com/stats.htm`. last access: April 26, 2015

not be available in another, localization is costly, and thus an unrestricted “flow of information” across language borders is hindered. While the amount of online textual content is certainly too large to localize exhaustively, a Cross-Language Information Retrieval (CLIR) system that is able to automatically search within foreign documents and information sources may succeed in transcending these language borders and satisfy globalized information needs. Even though search results may not be presented in the user’s own or native language, such a system may be valuable for the following reasons:

1. Relevant foreign information may be encoded in non-textual format, such as images or videos, which may be understandable even without language proficiency of the user.

2. The set of relevant documents in the same language as the query is too small, and additional cross-lingual information helps to complete a partially satisfied information need.

3. Relevant foreign documents may further point to documents in the user’s native language with specialized terminology that prevented their direct monolingual retrieval. Thus, the foreign documents may act as an intermediate interlingua-like link between query and relevant documents.

4. The linguistic disparity between *competence* and *performance* (Chomsky, 1965): Users may be competent enough to understand foreign content, but unable to paraphrase their information need in that language (insufficient performance).

5. Maximizing recall over globalized document collections is crucial for economic reasons. Specialized search tasks, such as patent prior art search, price research on products, or search for international professional competitors, require the retrieval of *all* relevant information available. For example, given the relatively fixed structure of Ebay or Amazon article pages, users that are not fluent in the foreign language can still make reasonable judgments about condition and price of a product.

A CLIR system is thus faced with the problem of *automatic translation* to infer relevance of foreign documents to a user query. The following thesis focuses on the relation between Cross-Language Information Retrieval and Statistical Machine Translation (SMT) in a cross-lingual environment. In this, both translation of foreign messages such as social media data, and recall-oriented retrieval of cross-lingual documents, such as patent prior art or encyclopedic articles, is a common task. The way this is approached is two-fold: first, by showing how CLIR can improve translation, and second by exploring how Statistical Machine Translation (SMT) models are optimized and integrated for CLIR.

A (cross-lingual) search engine however, can only be as good as the input it is given, and thus user behavior is an aspect to be taken into account. Formulating a well-defined query is not a trivial task, and efficacies of retrieval systems depend on external variables such as query length: Jansen et al. (2000) have shown positive correlation between information need specificity and query length. On Text REtrieval Conference (TREC) ad hoc queries, longer queries typically lead to better search results due to ambiguity reduction (Belkin et al., 2003). However, due to simpler matching techniques in current consumer web search engines, this effect is lessened (Downey et al., 2008). Analogously, web search queries consist on average of two or three keywords only (Downey et al., 2008). However, Duarte-Torres et al. (2010) point out that younger generations tend to use slightly longer queries and express their information needs in complete sentences rather than in sets of loosely associated keywords. These trends indicate the growing need for (CL)IR systems that take into account context-sensitive mappings between queries and documents. In the monolingual case this has been recently promoted by the implementation of natural language query understanding in algorithms such as Facebook's Graph Search[4] or Google's Knowledge Graph[5]. In the cross-lingual case, context-sensitive translation, that is, conditioning the translation of a query term on its context by using phrase-based SMT models, has been shown to reduce the danger of *query drift* (Chin et al., 2008; Ture et al., 2012b; Dong et al., 2014, inter alia). Cross-lingual query drift refers to the loss of intended meaning in a query due to inadequate translation. In the experimental setups of this thesis, we seek to accommodate for this trend towards longer, natural language queries. First by using full Twitter search messages as queries to mine additional training data for SMT, and second, by evaluating SMT-based CLIR models for tasks where queries either consist of single sentences (topic descriptions), or short, coherent texts (patent abstracts).

[4] `http://tinyurl.com/bstn776`. last access April 26, 2015

[5] `http://www.google.com/insidesearch/features/search/knowledge.html`. last access: April 26, 2015

1.2 Ranking in Statistical Machine Translation and Information Retrieval

The task of an information retrieval model is to assign a ranking over a set of documents given some input query. While translation, namely transferring the same meaning and style of an input sentence into a target language, may look different to the retrieval task at first glance, current Statistical Machine Translation systems implicitly induce rankings over structured translation outputs ("hypotheses"), given some parameterized scoring function to choose the best translation (see Section 1.6). Efficient inference in such models is possible if the structured outputs are decomposable. That is, similar (partial) hypotheses can efficiently be grouped together. The same applies to the information retrieval problem, where keyword-based search within documents is most efficient, if words within documents are considered independent of each other (see Section 1.5).

This thesis aims to exploit those similarities in multiple ways. We first utilize an established word-based CLIR model to find "cross-lingual mates" of social media messages from Twitter. Cross-lingual mates are messages in another language that correspond to an approximate translation of the input message. This "mate-ranking" approach (Chapter 2) illustrates how efficient word-based translation models for CLIR can be used to improve SMT models of higher order, namely models that translate using phrases. Second, we turn to the problem of "translation ranking" (Chapter 3) to show that by parameter optimization of SMT models for retrieval, that is, by influencing the ranking of possible translation hypotheses, overall CLIR performance is improved. Finally, this leads to the design of a novel approach to combine translation and retrieval into a single decoding model (Chapter 5). It exploits similarities in ranking and decomposability constraints to allow core CLIR, a.k.a "document ranking", directly with a machine translation decoder.

1.3 Translation Challenges in Cross-Language Information Retrieval

Besides commonalities between ranking and translation, a CLIR system needs to take into account the special challenges that arise for translation in an IR environment. In contrast to end-to-end applications of Statistical Machine Translation, translation outputs from a CLIR system are usually not visible to the user. While regular translation applications are optimized for matching human reference translations, and are required to produce readable and fluent output, the situation in CLIR is different due to fewer syntactical constraints, specialized lexical choice, or short queries providing sparse context. Most IR systems use stopword lists and stemming techniques to reduce the size of the vocabulary and avoid scoring of indiscriminative terms. This means, that translations for CLIR are most effective when they feature the correct lexical choices to match relevant documents. Disfluencies and syntactical errors in a translation, that would significantly degrade the perceived quality of the output in traditional SMT (Krings and Koby, 2001), are less of an issue in CLIR.

We illustrate the special requirements on translation for CLIR in the following and motivate the approaches presented in this thesis. Furthermore, the differences in requirements on translation for traditional SMT and CLIR are reflected in the heterogeneous way we carry out evaluation in the following chapters: "Mate-ranking" approaches are means to improve end-to-end translation of special domain data, and as such they are evaluated in terms of standard SMT measures such as BLEU[6] (Papineni et al., 2002). "Translation ranking" and "document ranking" approaches, on the other hand, are supposed to improve translation outputs for retrieval. Therefore, they are subject to a retrieval-based evaluation that requires relevance annotations for queries and documents.

1.3.1 Translation: Mapping between two Languages

In a Cross-Language Information Retrieval environment, queries and documents are written in two different languages. In order to match terms across languages, which provide different surface forms for the same term, a retrieval system needs to establish a *mapping* between words in the query vocabulary and words in the document vocabulary. This mapping is given by a Statistical Machine Translation model in the following thesis. More precisely, we use a statistical model that maps source to target words (or phrases), and assigns confidence values to these mappings such that multiple translation options are differently weighted. In CLIR, Nie (2010) distinguishes between three ways of defining such translation mappings:

1. from query language to document language (*query translation approach*),
2. from document language to query language (*document translation approach*), or

[6] BLEU is defined in Section 1.6

3. from query and document languages into a pivot language (“interlingua”).

In this work, we focus on query translation which is the approach with the least overhead: queries are typically shorter than documents, thus requiring less translation effort. Conversely, the translation of the full document repository would require large amounts of processing and storage power, and translated versions of the documents would need to be stored. In a query translation approach, however, the translation of the query can be discarded once a ranked list is returned. In an interlingua approach, overall translation effort is doubled and translation inaccuracies, leading to query drift, can occur twice. Additionally, choosing an adequate interlingua (a third natural language or a symbolic language that encodes semantic concepts) introduces another aspect that would be subject to substantial development.

1.3.2 Translation for Query Expansion: ``Bridging the Lexical Chasm"

Consider the case, in which a query only consists of words which happen to not occur in any of the relevant documents the user seeks to find. This problem is sometimes called the “lexical chasm” (Berger et al., 2000), as there are no lexical items indicating query-document similarity on the surface. With the general brevity of queries in web search (Belkin et al., 2003), such chasms are common. A popular technique to reduce this problem is to automatically expand queries with related terms, synonyms, variants from a thesaurus such as WordNet (Fellbaum, 1998), spell-check suggestions, or a list of terms with similar statistical properties (Qiu and Frei, 1993; Voorhees, 1994; Xu and Croft, 1996). This technique is commonly termed *query expansion*, and its importance has been described already in 1988 by Swanson (1988).

While longer queries generally increase the likelihood of lexical matches between query and documents in monolingual retrieval, and lead to improved performance (Belkin et al., 2003), the situation in Cross-Language Information Retrieval is far more severe. Here, the correctness of the mapping between query and document terms is fully dependent on the performance of the translation component. It should not only adequately translate query terms, but also provide additional translation alternatives as expansion terms to increase the likelihood of retrieving relevant documents. Xu et al. (2001) presented an integration of word translation probabilities in a probabilistic retrieval framework (Section 1.5). The use of alternatives from a fully-trained, phrase-based SMT model for cross-lingual query expansion in the framework of Probabilistic Structured Queries (Darwish and Oard, 2003) was explored by Ture et al.

(2012a,b) recently. Even in the monolingual case, query expansion through statistical translation models can be achieved by query re-writes with a statistical machine translation system trained on monolingual data (Riezler and Liu, 2010), or integration of word-based translation models for question-answering tasks (Xue et al., 2008; Berger et al., 2000; Hieber and Riezler, 2011).

In general, translation-based CLIR benefits from query expansion through alternatives given by the translation model. However, a common approach to CLIR, due to its simplicity and efficacy, is to translate the query using an existing phrase-based SMT model and forward the single first-best translation to a monolingual retrieval model (Direct Translation (DT)). This pipeline approach is justified in the case where the SMT model is only available as a "black box" and provides only its single preferred output translation (Chin et al., 2008). However, any translation errors produced in such black box SMT are propagated through the pipeline to the retrieval model (Dong et al., 2014). Thus, opening the "black box" of SMT for CLIR further, initiated by Ture et al. (2012b) inter alia, is one of the central goals in this thesis.

1.3.3 Avoiding Query Drift: Context-Sensitive Query Translations

While query expansion provides means to "broaden" the search in the document collection, excessive use of expansion terms increases the risk of query drift, that is, retrieving documents through expansion terms that do not correspond to the user's information need. In CLIR, such semantic shifts are far more likely, since all terms of the source language query are mapped into a foreign language, and translation errors due to context sparsity lead retrieval in the wrong direction. In monolingual retrieval, limiting the number of expansion terms by assigning weights for statistical associativity to original query terms is the standard option (Qiu and Frei, 1993; Voorhees, 1994; Xu and Croft, 1996). In CLIR, such confidence scores for translation alternatives are given by a statistical translation model. One of the central arguments for the use of "higher-order" translation models, i.e. phrase-based systems, is to avoid query drift by selecting query term translations that depend on the surrounding context. SMT models that use context-sensitive feature functions such as n-gram language models ensure fluency of the output and promote context-sensitive translations of multi-word expressions. An n-gram language model estimates the conditional probability of word t_i given its predecessors, $P(t_i|t_{i-1}, \dots, t_{i-n-1})$. In traditional SMT, such language models ensure fluency of the sentence in the target language, i.e. more probability mass is given to likely sequences of words in the target language.

For CLIR however, context-sensitive feature functions such as language models can have mixed effects: On the one hand, they provide means to ensure adequate selection of term translations based on previous contexts, which improves translation of multi-word expressions or multi-word named entities. On the other hand, they largely determine the structure of the search space from which translation hypotheses are produced (see Chapter 3 and 5). CLIR models that exploit only the n-best translation hypotheses of SMT models (Ture et al., 2012b) for query expansion may see very little diversity among translation hypotheses, thus having only limited query expansion capabilities. We show in experiments, that a CLIR model should be able to explore the full search space of the translation model to select the best possible translation for the task of retrieval, either at parameter optimization time (Chapter 3), or at test time (Chapter 5).

1.3.4 Modeling Human-like Translation in CLIR

A good translation of an input sentence is usually the one that conveys the correct meaning and converts the original style fluently to an adequate expression in the target language. Achieving this, is not only a complex task for professional human translators but even more for statistical translation systems with no world knowledge. Nevertheless, research in SMT strives to achieve human-like translation quality, for example through consistency constraints (Carpuat and Simard, 2012) or by integration of syntactic and semantic constraints into the decoding process (Wu and Fung, 2009; Liu and Gildea, 2010; Huang et al., 2010; Marton and Resnik, 2008, interalia). However, such efforts in SMT can actually be counterproductive for translation-based CLIR, since common terms may actually be downweighted regularly by IR term weighting schemes that promote rare and discriminative terms (see Section 1.5). A CLIR system should take the user's ignorance of terminology for informational queries into account: users may only have a very vague idea of terminology in documents that are relevant to their information need. The best translation of a very common source language query term may thus not be its common counterpart in the target language, but rather a more discriminative variant. Recovering the *intended* meaning of the user query can hence be more effective if the model explores less common, but highly discriminative terms in the translation model search space. This problem is intensified in specialized retrieval tasks such as cross-lingual article research (Chapter 4) or patent prior art search, where usage of highly domain-specific terms are common, but may not be known to (casual) users. For them, the disparity between query and document domain may lead to larger “lexical chasms”.

While existing work has proposed diversity exploration in the context of system combination for SMT (Cer et al., 2013), we argue that promoting diverse query translations for CLIR is achieved best if the translation model is aware of its use in retrieval. We will present a discriminative training approach to SMT to optimize lexical choice in Chapter 3. In Chapter 5, we will furthermore design a model that balances the objective of adequate translation and query expansion to retrieve relevant documents. Such a joint model naturally selects non-standard translations if a document candidate demands so.

1.4 Research Contributions and Outline of this Thesis

Given the trend towards natural language queries and increased multilinguality in modern information society, as presented in the previous sections, robust CLIR systems should perform context-sensitive query translations. I show that state-of-the-art SMT models can provide such context-sensitivity through the use of language models. However, simple pipeline approaches are suboptimal due to optimization of SMT models for the task of standalone translation. In the context of CLIR, translation should be optimized for retrieval, that is, for the task of providing the user with a list of the most relevant documents. I thus propose to abandon the commonly used approach of pipelining translation and retrieval, with or without query expansion techniques, and integrate retrieval functionality directly into the SMT model (Chapter 5). Experiments on two large-scale data sets suggest that such an approach significantly outperforms standard CLIR pipeline methods.

The contributions of my thesis are of cumulative nature. Each chapter corresponds to published or submitted research papers by me, including colleagues and my supervisor. A list of my contributions precedes each chapter to distinguish between personal work, and work done by co-authors.

Research contributions to the field of Statistical Machine Translation:

- A method to crawl large amounts of Twitter messages in two languages, providing means to create comparable sentence pairs using Cross-Language Information Retrieval. Furthermore, usage of such data for adapting a general-domain SMT model towards the domain of Twitter messages (Jehl et al., 2012; Hieber et al., 2013).

- A decomposable proxy for retrieval quality, suitable for large-scale discriminative training of SMT systems for CLIR (Sokolov et al., 2014a).
- A new model of CLIR in hierarchical phrase-based SMT decoding for optimal query expansion (Hieber and Riezler, 2015).

Research contributions to the field of Cross-Language Information Retrieval:

- Large-scale experimental evaluations of state-of-the-art SMT-based CLIR models: extrinsic evaluation through domain adaptation experiments (Jehl et al., 2012; Hieber et al., 2013); intrinsic evaluations on relevance annotated data (Sokolov et al., 2013; Schamoni et al., 2014; Sokolov et al., 2014a).
- A method to automatically extract relevance annotations from Wikipedia data, allowing large-scale training of ranking models (Schamoni et al., 2014).

The following sections of this chapter provide the reader with a brief overview over the field of Information Retrieval and Statistical Machine Translation. They introduce recurrent concepts and definitions used throughout this work, and are not meant to provide an exhaustive overview of research in IR and SMT. Such an attempt would be out of the scope of this thesis.

1.5 Basics in (Cross-Language) Information Retrieval

Let us briefly introduce recurrent IR-related concepts, ranking models, and evaluation metrics that are used throughout this work. The task of information retrieval is defined as follows:

Definition 1 *Let $d \in \mathcal{S}$ be some document string from the set of strings $\mathcal{S}$, with length ℓ_d, where d_j denotes the jth word and d is part of a document collection $\mathbf{C} = \{d^1, \dots, d^N\}$ with N documents. Given a user's information need, expressed as query string $q \in \mathcal{S}$ with length ℓ_q and words $q_1, ..., q_{\ell_q}$, an Information Retrieval system induces a ranking r over the set of documents in $\mathbf{C}$ and presents an ordered list of the top-k documents to the user. Formally, r is a binary relation $\mathbf{C} \times \mathbf{C}$ that fulfills the properties of a weak ordering. Ties between documents are allowed and we denote the preference relation under r with $d^k > d^g$, indicating that d^k is ranked above d^g. A ranking r is produced by a retrieval or scoring function $score : \mathcal{S} \times \mathcal{S} \mapsto \mathbb{R}$ that computes similarity scores between two strings. Given query string q, a retrieval system evaluates $score(q, d^k)$ for each $d^k \in \mathbf{C}$, thereby producing r under retrieval function $score$.*

In order to define a retrieval function, one needs to establish a suitable representation for query and document strings. The choice of representation determines the type of retrieval model used. One of the most common representations is the *Bag-of-Words*. A string is segmented into a list of words (terms) by splitting at any white-space character. The vocabulary V is the set of distinct terms that occur in the document collection $\mathbf{C}$. A common technique to reduce the size of vocabulary V is to apply stemming, which establishes homonymy of inflected words by converting them back to their common stem (e.g. Porter, 1997). Furthermore, very common terms, usually referred to as "stopwords", are unlikely to provide information about single documents and are removed from V. A Bag-of-Words representation of texts discards positional information of terms but typically includes frequency information to incorporate a notion of term importance. The Bag-of-Words representation of a text entails one of the most simplifying modeling assumptions in IR, namely term independence. On one hand, this allows very fast indexing procedures and retrieval in huge collections. On the other hand, this assumption clearly does not accurately model natural language and drops contextual information. Several methods have been presented to overcome this limitation, and especially for Cross-Language Information Retrieval, where an adequate mapping between source and target terms is sought, performance can be significantly improved with models of weaker independence assumptions, providing more contextual information.

Mathematically, a Bag-of-Words is represented as a *document vector* $\mathbf{d} \in \mathbb{R}^{|V|}$ in vocabulary space V. Values within $\mathbf{d}$ correspond to term weights assigned by a *term weighting scheme* $f : V \mapsto \mathbb{R}$, such as simple term frequency counting $tf(t, d) = |\{d_j | d_j = t\}|$. The idea behind such schemes is to assign weights according to information value and discriminative strength of different terms. Retrieval function that use such term weights are able to assign more fine-grained similarity scores to queries and documents.

1.5.1 Types of Retrieval Models

Retrieval models describe means to estimate relevance of documents given queries. Among the various models proposed in IR research, we briefly introduce the most important types of models based on term representations in the following: Boolean retrieval models (Section 1.5.1.1) for the sake of completeness and simple introduction, vector-space models (Section 1.5.1.2) which are the most common choice in regular retrieval systems due to easy integration of various term weighting schemes, and probabilistic and language models (Section 1.5.1.4 and 1.5.1.3, respectively). The latter type provides the theoretical framework to integrate Statistical Machine Translation into Cross-Language Information Retrieval systems. Although vector-space and probabilistic models build on different theoretical frameworks, namely geometrical spaces and probability theory, they often use very similar term weighting schemes. (Manning et al., 2008, p. 212).

1.5.1.1 Boolean Retrieval

In a boolean model, documents are represented as binary vectors $\mathbf{d} \in \{0, 1\}^{|V|}$ that encode the presence or absence of terms within a documents. The set of occurring terms t_i is interpreted as a logical conjunction $t_1 \wedge t_2 \wedge t_3$. Queries are expressed as logical expressions composed of conjunction, disjunction, or negation, e.g. $(t_1 \wedge \neg t_2) \vee t_3$. Retrieval of document d given query q is understood as a logical implication, i.e. d is only returned if it satisfies the logical constraints of q: $d \rightarrow q$. A boolean retrieval model only divides the collection $\mathbf{C}$ into a set of matching documents, $\mathbf{C}_m = \{d | q : d \rightarrow q\}$, and a set of non-matching documents, $\mathbf{C}_{\bar{m}} = \mathbf{C} \backslash \mathbf{C}_m$. It does not induce a ranking over $\mathbf{C}_m$. Due to the absence of graded relevance, it is mainly used for expert queries in special domains where the set of search results can be reduced by issuing complex logical expressions. In regular web search, information needs are usually a loose combination of keywords without strict logical constraints. Especially for longer natural language queries, a logical interpretation of the query is infeasible.

1.5.1.2 Vector-Space Models

In a vector-space model, the ranking over documents is given by the (geometrical) distance of documents to the query in the common vector-space for V. Terms in document and query vectors $\mathbf{d}, \mathbf{q} \in \mathbb{R}^{|V|}$ are typically weighted using some variations of $tfidf$ weights (Sparck-Jones, 1988):

$$tfidf(t; \mathbf{d}, \mathbf{C}) = tf(t, \mathbf{d}) \times idf(t, \mathbf{C}) \tag{1.1}$$

$$idf(t, \mathbf{C}) = log \frac{|\mathbf{C}|}{df(t, \mathbf{C})}, \tag{1.2}$$

where $tf(t, d)$ is the term frequency of t in document d and $df(t, \mathbf{C})$ is the *document frequency*, indicating the number of times term t occurs in collection $\mathbf{C}$, $|\mathbf{C}| = N$. The $tfidf$ weighting scheme assigns large weights to representative and discriminative terms. A term is representative if it occurs prominently in a single document (large tf). It is discriminative if it only occurs in few documents (small df). A vector-space retrieval model evaluates similarity between query and document vectors according to the *cosine* score, i.e. the angle between length normalized $\mathbf{q}$ and $\mathbf{d}^k$ vectors:

$$score(q, d) = cosine(\mathbf{q}, \mathbf{d}) = \frac{\mathbf{q} \cdot \mathbf{d}}{||\mathbf{q}|| \times ||\mathbf{d}||}. \tag{1.3}$$

Fast inverted index representations (Zobel and Moffat, 2006) and a variety of empirically motivated term weighting schemes (Zobel and Moffat, 1998) guaranteed the popularity of vector-space models in the past. For Cross-Language Information Retrieval, defining a common vector space between query and document vocabularies was done by Latent Semantic Indexing (Deerwester et al., 1990) in the dimensionality-reduced common latent space (Littman et al., 1998). However, subject of this work is the integration of statistical machine translation models into probabilistic retrieval frameworks to provide a direct mapping between query and document language.

1.5.1.3 Probabilistic Relevance Models and Okapi $BM25$

Probabilistic relevance models estimate the probability of binary random variable relevance, R, given document and query:

$$P(R = 1|\mathbf{q}, \mathbf{d}) + P(R = 0|\mathbf{q}, \mathbf{d}) = 1. \tag{1.4}$$

The first probabilistic relevance model, the Binary Independence Model (BIM), introduced this idea by using only boolean Bag-of-Words representations, i.e only encoding presence or absence of terms in documents (Robertson and Sparck-Jones, 1976; Rijsbergen, 1979). A ranking of documents in $\mathbf{C}$ is induced by sorting documents according to the odds of relevance:

$$O(R|\mathbf{q}, \mathbf{d}) = \frac{P(R = 1|\mathbf{q}, \mathbf{d})}{P(R = 0|\mathbf{q}, \mathbf{d})}, \tag{1.5}$$

which is monotone with respect to the probability of relevance (Manning et al., 2008, p. 206). From this, a Retrieval Status Value (RSV) is derived as a ranking function (Manning et al., 2008, p. 207):

$$score(q, d) = RSV(\mathbf{q}, \mathbf{d}) = \sum_{t \in \mathbf{q} \cap \mathbf{d}} log \frac{P(t \in \mathbf{d}|R = 1, \mathbf{q}) \cdot (1 - P(t \in \mathbf{d}|R = 0, \mathbf{q}))}{P(t \in \mathbf{d}|R = 0, \mathbf{q}) \cdot (1 - P(t \in \mathbf{d}|R = 1, \mathbf{q}))}, \tag{1.6}$$

where the quantities in the fraction can be estimated from available relevance judgments. Similar to a boolean model (1.5.1.1), the BIM was mainly designed for expert search in catalog records and does not take term frequencies or document lengths into account. Analogously to the $tfidf$ term weighting scheme in vector-space models, probabilistic relevance modeling moved towards the integration of term frequency weights.

One of the most successful approaches is the Okapi $BM25$ weighting scheme (Robertson et al., 1998; Robertson and Zaragoza, 2009), which extends the idea of binary relevance to incorporate (sub-linear) term frequencies and lengths. The $BM25$ term weighting scheme became a strong baseline in the TREC Web Retrieval Tasks (Robertson et al., 1998; Craswell et al., 2003; Clarke et al., 2004; Robertson, 2005). It resembles a $tfidf$-like term weighting scheme in a vector-space model but is derived from a 2-Poisson probabilistic model of relevance:

$$O(R|q,d) = \cdots = BM25(q,d) = \sum_{t \in q} bm25(t,d) = \sum_{t \in q} rsj(t,\mathbf{C}) \cdot tf_{bm25}(t,d). \quad (1.7)$$

A detailed derivation of the Okapi $BM25$ formula can be found in Robertson and Zaragoza (2009). The $rsj(t)$ term is the Robertson/Sparck-Jones weight and its approximation, in the absence of available relevance judgments, corresponds to a smoothed inverse document frequency (idf) term weight:

$$idf(t,\mathbf{C}) \approx rsj(t,\mathbf{C}) = log\Big(\frac{|\mathbf{C}| - df(t,\mathbf{C}) + 0.5}{df(t,\mathbf{C}) + 0.5}\Big), \quad (1.8)$$

where df is the number of documents t occurs in. The second part of (1.7) is a term saturation formula that limits the impact of observing a term multiple times (Svore and Burges, 2009):

$$tf_{bm25}(t,d) = \frac{tf(t,d)}{k_1((1-b) + b\frac{dl}{avdl}) + tf(t,d)}, \quad (1.9)$$

where $tf(t,d)$ is the frequency of term t in document d, and k_1 a saturation parameter controlling the sub-linear growth $tf_{bm25}(t,d)$ with respect to $tf(t,d)$. Figure 1.1 shows saturated term frequencies for various values of k_1. The function satisfies three key properties (Svore and Burges, 2009):

1. $tf_{bm25}(t,d) = 0$ if $tf(t,d) = 0$,
2. $tf_{bm25}(t,d)$ increases monotonically with $tf(t,d)$,
3. but has an asymptotic limit of 1.

For $k_1 = 0$, tf_{bm25} reduces to a boolean term indicator. With large k_1, tf_{bm25} scales nearly linear with growing $tf(t,d)$. Typically k_1 is set to small values, such that tf_{bm25} quickly saturates even for small term frequencies. Parameter b controls the weight of document length normalization, which is defined with respect to the average document length $avdl$ in collection $\mathbf{C}$. If b is small, the effect of length normalization is reduced. Based on the asymptotic limit of $tf_{bm25}(t,d)$, we note that for $k_1 > 0$, $rsj(t)$ is an upper bound for $bm25(t,d)$:

$$\forall tf(t,d) : bm25(t,d) < rsj(t). \quad (1.10)$$

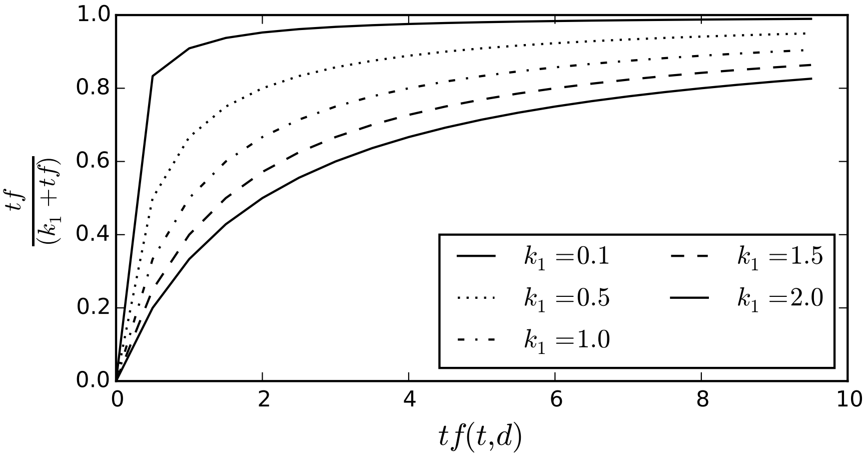


Figure 1.1: $BM25$ term frequency saturation function.

Previous work has presented ways to tune $BM25$ parameters via grid-search or gradient descent methods (Taylor et al., 2006; Svore and Burges, 2009). Other approaches have extended $BM25$ to respect multiple document “fields”, $BM25F$, to learn different parameters for headlines or paragraphs in documents (Zaragoza et al., 2004). For the following work, we fix k_1 and b to commonly used values of $k_1 = 1.2$ and $b = 0.75$ in the popular open source search engine Lucene.[7] Section 2.4 utilizes a cross-lingual extension to $BM25$-based retrieval in the framework of *Probabilistic Structured Queries* (Darwish and Oard, 2003), where term and document frequencies are computed as expected values over term translation distributions (Ture et al., 2012a,b). Chapter 3 presents an approach to optimize a Direct Translation baseline, where query translations are used for monolingual $BM25$ retrieval. Chapter 5 uses $BM25$-inspired sparse lexicalized features to model retrieval quality within an SMT decoder. All experiments with $BM25$-based models were carried out with my own C++ implementations, available at `https://github.com/fhieber/cclir`.

1.5.1.4 Language Models in IR

Another probabilistic retrieval approach are language models. Language model-based approaches make use of the idea that a user formulates his information need by imagining the document that (s)he would like to find. (S)he consequently attempts to express the query such that it is likely to resemble the language of that document.

[7] `http://lucene.apache.org/`. last access: April 26, 2015

The generated query can be viewed as a distorted version of the document (Ponte and Croft, 1998). If documents are represented as language models, one can compute the likelihood of the query with respect to each document language model. To extend this idea to a ranking function, documents are ranked by the their probability of generating the query string (Ponte and Croft, 1998). The general form of the *query likelihood model* (Berger and Lafferty, 1999) is:

$$score(q,d) = P(d|q) = \frac{P(d)P(q|d)}{P(q)} \approx P(q|d), \tag{1.11}$$

where, by application of Bayes' rule, $P(d|q)$ reduces to $P(q|d)$ under a uniform prior over documents, since $P(q)$ is constant and does not affect the document ranking.

If documents are modeled as uni-gram language models, the strict assumption of term independence allows a Bag-of-Words representation of the documents with relative frequencies as term weights. The likelihood of query q under the language model m_d for document d is then written as

$$\hat{P}(q|m_d) = \prod_{t \in q} \hat{P}(t|m_d) \approx \prod_{t \in q} \frac{tf(t,d)}{\ell_d}, \tag{1.12}$$

where the last term is a maximum likelihood estimate of term t under model m_d (Manning et al., 2008). A problem with the conjunctive nature of the product evaluation is the sparsity of the document model: query terms that do not occur in document d let the whole product become zero. This is, similar to the logical evaluation of Boolean models, not a desirable behavior in common retrieval applications where a graded notion of relevance is required. To return nonzero probabilities for these cases, one resorts to smoothing, that is, reserving some probability mass to unseen events. Besides standard techniques such as add-α smoothing, a common way of smoothing language models is interpolation, also known as Jelinek-Mercer smoothing (Zhai and Lafferty, 2001).

$$\hat{P}(t|d) = \lambda \hat{P}(t|m_d) + (1-\lambda)\hat{P}(t|m_{\mathbf{C}}), \tag{1.13}$$

where the weight of term t is a linear interpolation between the likelihood in document d and an overall *expected* weight such as

$$\hat{P}(t|m_{\mathbf{C}}) = \frac{tf(t,\mathbf{C})}{|C|}. \tag{1.14}$$

For numerical stability, scoring is usually carried out with the sum of logarithms:

$$logP(q|d) = log\prod_{t\in q} P(t|d) = \sum_{t\in q} logP(t|d). \tag{1.15}$$

Cross-lingual probabilistic retrieval. For Cross-Language Information Retrieval, weighted alternatives are usually integrated from probabilistic dictionaries, such as *lexical translation tables* which are obtained from unsupervised word alignment on parallel sentence data (Och and Ney, 2003). Such tables encode multinomial probability distributions over target language words for each source language term in the vocabulary and are integrated into a probabilistic retrieval framework as such (Xu et al., 2001):

$$P(q|d) = \sum_{t\in q} logP(t|d), \tag{1.16}$$

$$P(t|d) = \sum_{t\in V} T(t|u)\hat{P}(u|m_d), \tag{1.17}$$

where T is a lexical translation table that encodes the likelihood of document term u being a translation of term t. We will apply such a model in Section 2.2 for "mate ranking".

1.5.2 Evaluation of Retrieval Systems

The often empirical nature of term weighting schemes and retrieval function requires extensive and thorough evaluations of different retrieval systems. In this work, we evaluate performance of CLIR models in two ways, namely intrinsically, that is, by judging retrieval performance given labeled test data (see Chapters 3 and 5), and extrinsically, by measuring performance of a larger system, in which the retrieval model is only part of the pipeline (see Chapter 2).

1.5.2.1 Intrinsic Evaluation with Relevance Labels

As described in the introduction, an intrinsic evaluation of retrieval performance amounts to compare recall and precision of system-produced rankings against the optimal ranking given by annotated relevance data: Relevant documents to a given query should be placed above irrelevant documents (prevision), and we seek to find all relevant documents (recall). Common evaluation measures differ in their way of balancing both ranking aspects, and how they integrate a graded notion of relevance.

In order to provide a thorough evaluation of rankings, we briefly introduce the three measures used in this work that provide a representative selection over both recall-oriented and precision-oriented evaluation metrics.

Annotated relevance data consists of *relevance judgments* that are available for a test (or development) set of queries. Relevance judgments flag a subset of documents as relevant with respect to a query. More formally, we write relevance labels as the output of a function $rl : \mathbf{C} \times Q \mapsto \mathbb{N}_0^+$, assigning a positive *relevance level* to each relevant document $\mathbf{d}^+ \in \mathbf{C}$ w.r.t q: $rl(d, q) > 0$. For irrelevant or non-judged documents d^-, $rl(d_i, q) = 0$. Higher relevance levels encode the preference of highly relevant documents in a ranking. Note that relevance judgments usually do not encode a full ranking over documents, such that there may be multiple "optimal" rankings that satisfy the preference constraints encoded by rl. Evaluation is carried out by comparing the system-induced rankings with (one of) the optimal rankings from rl. In Chapter 4, we describe an approach to automatically extract relevance judgments from Wikipedia data, and create a large-scale cross-lingual retrieval data set.

In the following, let k be the number of documents retrieved per query, $q \in Q$ a set of queries, $\pi_q(i)$ indicating the document ranked at position i, and $r_q = \sum_{d \in \mathbf{C}} [\![rl(d, q) > 0]\!]$ be the total number of relevant documents in $\mathbf{C}$ for q, where $[\![a]\!] = 1$, if $a = True$, 0 otherwise.

Mean Average Precision (MAP) A standard measure among the TREC community is Mean Average Precision (Baeza-Yates and Ribeiro-Neto, 1999), which returns a single score of retrieval performance across all recall levels (Manning et al., 2008). It only considers the case of binary relevance judgments. For a set of queries Q, it is the mean of the *Average Precisions* (AP) for each query:

$$MAP(Q) = \frac{\sum_{q \in Q} AP(q)}{|Q|}, \tag{1.18}$$

with Average Precision for q in the top-k documents defined as

$$AP(q) = \frac{1}{r_q} \cdot \sum_{i=1}^{k} \frac{1}{i} \sum_{j=1}^{i} [\![rl(\pi_q(j), q) > 0]\!]. \tag{1.19}$$

The MAP score is a number between 0 and 1, where higher is better.

Normalized Discounted Cumulative Gain (NDCG) To evaluate performance in tasks with a graded notion of document relevance, we compute the Normalized Discounted Cumulative Gain (Järvelin and Kekäläinen, 2002). NDCG is a number between 0 and 1, where larger gains are received when documents with higher relevance levels are ranked on top:

$$NDCG@k(q) = \frac{1}{Z_k} \sum_{i=1}^{k} \frac{2^{rl(i,q)} - 1}{log_2(i+1)}, \tag{1.20}$$

$$Z_k = \max_{\pi_q} \sum_{i=1}^{k} \frac{2^{rl(i,q)} - 1}{log_2(\pi_q^{-1}(i)+1)}, \tag{1.21}$$

where Z_k is a normalization constant so that a perfect ranking for $NDCG@k$ would be 1. We adopt the notation of Chaudhuri and Tewari (2014), where π_q^{-1} as the inverse of π returns the *rank* of document i in the ranking given for query q. NDCG for a set of queries Q is the mean of individual NDCG scores.

Patent Retrieval Evaluation Score (PRES) The Patent Retrieval Evaluation Score, as introduced by Magdy and Jones (2010), defines a measure for recall-oriented retrieval tasks such as patent prior art search retrieval. While MAP and NDCG are traditionally precision-oriented metrics, PRES, a refined version of normalized recall, measures the quality of a ranking with respect to the recall up to a cutoff parameter N_{max}, controlling the number of results a user is willing to skim through. This stems from the observation that patent application examiners, given the importance of finding prior art, are inclined to look through a longer list of search results than in regular web search, where only the very first results are considered. PRES is defined as

$$PRES(q) = 1 - \frac{SR - r_q \frac{(r_q+1)}{2}}{r_q \cdot N_{max}}, \tag{1.22}$$

$$SR(q) = \sum_{i=1}^{k} i + (r_q - f_q) \cdot (N_{max} + r_q - \frac{(r_q - f_q) - 1}{2}),$$

where r_q is again the total number of relevant documents in $\mathbf{C}$ for query q, and

$$f_q = \sum_{i=1}^{k} [\![rl(\pi_q(i), q) > 0]\!] \tag{1.23}$$

is the number of relevant documents retrieved. PRES is a score between 0 and 1 and "[...] a portion of the recall depending on the quality of ranking of the relevant documents relative to N_{max}" (Magdy and Jones, 2010). PRES of a set of queries Q is the mean of individual PRES scores.

Practical Remarks For MAP and NDCG, we use the `trec_eval` script[8]. PRES scores are computed with the script provided by the authors of the PRES[9] metric (Magdy and Jones, 2010). Note that the original version of the `trec_eval` script computes evaluation measures for system rankings via (re-)sorting the system output by model score, even though explicit rank information is given in the system output. In some cases, retrieval models of this work returned retrieval scores that exceeded the precision range of `trec_eval`'s variable for these scores, which led to inconsistent results. To guarantee correct evaluation of system outputs in Chapter 5, we changed the variable type to a double precision float.

Statistical Significance Testing of Retrieval Systems through Paired Randomization Tests. Given two retrieval systems A and B and their evaluation scores on a set of queries Q, we would like to determine which one is better than the other. For example, if system A produces a MAP score of 0.25, and system B a MAP score of 0.26, we would like to know if the difference of $\delta = 0.01$ was created by chance, or if the result of B is truly better than A, even if δ is small. Adequate statistical significance tests are designed to detect this. They are *powerful*, if significant differences are reliably detected, and *accurate*, if type-I errors, i.e. rejections of the true null hypothesis, are unlikely. In this thesis, we carry out a variant of Fisher's randomization test (Cohen, 1995), called the *paired randomization test*, which was shown to be most adequate and efficient for Information Retrieval evaluation by Smucker et al. (2007). The test statistic is the absolute difference of per-query decomposable IR measures between two systems, for example $\delta = |MAP_A(Q) - MAP_B(Q)|$. Under the null hypothesis, the test statistic is as likely as any other permutation of scores. More concretely, we can create all $N = 2^{|Q|}$ score permutations δ'_i and count the number of times, where $\delta'_i \geq \delta$, $c = \sum_{i=1}^{N} [\![\delta'_i \geq \delta]\!]$. Normalization by the total number

[8] `http://trec.nist.gov/trec_eval/`. last access: April 26, 2015

[9] `http://alt.qcri.org/~wmagdy/PRES.htm`. last access: April 26, 2015

of permutations yields the two-sided p-value for the null hypothesis:

$$p = \frac{c}{N}.$$

If the number of test queries is large, creating all possible permutations is infeasible. We thus sample score permutations between both systems, record their differences and normalize c by the number of samples S:

$$p = \frac{\sum_{i=1}^{S} [\![\delta'_i \geq \delta]\!]}{S}.$$

The more samples we create, the more accurate our p-value will be. If the p-value lies below our specified rejection level, we can reject the null hypothesis, concluding that both systems are significantly different from each other. We carry out paired randomization tests for experiments in this thesis using the script provided by Smucker et al. (2007).

1.5.2.2 Extrinsic Evaluation of CLIR Models in Larger Pipelines

Besides an intrinsic evaluation of retrieval performance, we also assess performance of retrieval models by using their search results in a larger pipeline. For example, pairs of queries and search results returned by a Cross-Language Retrieval System can be used as additional parallel data for training a Statistical Machine Translation System. The extrinsic measure of retrieval quality is then the performance of such re-trained translation systems in terms of common translation metrics, for example the corpus-based BLEU score (Papineni et al., 2002) (see Section 1.6). Parallel data mining, i.e. *mate-finding*, refers to the task of finding the (single) document (the query's "cross-lingual mate") that constitutes an exact, or comparable translation of the query. In Chapter 2, we present two approaches to cross-lingual mate-finding on social media data to provide additional training data for Statistical Machine Translation. We thus evaluate the quality of retrieval in terms of the model's ability to provide accurate and precise training data for re-training an SMT system.

1.6 Basics in Statistical Machine Translation

Section 1.5.1.4 described how to incorporate context-free probabilistically weighted translations into a language-model based retrieval model. In order to provide context-sensitive translation of queries, the integration of full Statistical Machine Translation systems into CLIR was recently shown to be helpful (Ture et al., 2012b). Such translation systems use bilingual multi-word phrases as minimal translation units.

Translation Formalisms. In this thesis, we work with two translation formalisms, namely phrase-based machine translation (Koehn et al., 2003), and hierarchical phrase-based machine translation (Chiang, 2007). The main difference between these frameworks is that hierarchical bilingual phrases can contain gaps. These gaps are encoded as non-terminal symbols in a context-free grammar, allowing the encoding of long-distance dependencies and reordering directly at the phrase level. Phrase-based models (Chapter 2 and 3) use `Moses` (Koehn et al., 2007). Hierarchical phrase-based models (Chapter 3 and 5) use `cdec` (Dyer et al., 2010). The following paragraphs provide a brief overview of statistical machine translation, independent from the underlying translation formalisms.

Model & Inference. SMT models estimate the conditional probability of output sentence $\bar{e}$ given input sentence $\bar{f}$ (Koehn, 2010):

$$P(\bar{e}|\bar{f}) = \sum_{h \in \mathcal{E}_{\bar{f}}} P(h, \bar{e}|\bar{f}), \tag{1.24}$$

where h is an hypothesis (or derivation) in the search space $\mathcal{E}_{\bar{f}}$ over all possible hypotheses for a given input $\bar{f}$. The *Maximum A Posteriori* (MAP) inference is given by

$$\bar{e}^* = \arg\max_{\bar{e}} P(\bar{e}|\bar{f}) = \arg\max_{\bar{e}} \sum_{h \in \mathcal{E}_{\bar{f}}} P(h, \bar{e}|\bar{f}). \tag{1.25}$$

Since many derivations can yield the same output string $\bar{e}$, such inference is an NP-hard problem (Knight, 1999). Most SMT systems (Koehn et al., 2003; Chiang, 2007) thus resort to a Viterbi approximation that returns the output corresponding to the most probable hypothesis:

$$\bar{e}^* = \arg\max_{\bar{e}} \max_{h \in \mathcal{E}_{\bar{f}}} P(h, \bar{e}|\bar{f}). \tag{1.26}$$

The probability of a translation output $\bar{e}$ under derivation h given $\bar{f}$ is usually modeled in a log-linear model (Koehn et al., 2003; Chiang, 2007),

$$P(h, \bar{e}|\bar{f}; \mathbf{w}) = \frac{e^{\sigma(h,\bar{e},\bar{f})}}{\sum_{\bar{e},h} e^{\sigma(h,\bar{e},\bar{f})}}, \tag{1.27}$$

where $\sigma(h, \bar{e}, \bar{f})$ is a learned linear combination of input-output features, that is the dot product between parameter column vector $\mathbf{w}$ and feature column vector given by feature map $\mathbf{\Phi}$,

$$\sigma(h, \bar{e}, \bar{f}) = \mathbf{w}^T \mathbf{\Phi}(h, \bar{e}, \bar{f}). \tag{1.28}$$

Word Alignments. Standard features for SMT models are typically estimated on large amounts of parallel bilingual data that are aligned on the sentence level. In order to extract word or phrase correspondences from such data, word alignments need to be constructed. An alignment is a function mapping a word in the target sentence to word(s) in the source sentence. Various (generative) alignment models (Vogel et al., 1996; Och and Ney, 2003; Dyer et al., 2013) that describe word correspondences in parallel data were presented in the past. Mostly unsupervised, they maximize the likelihood of the parallel data through algorithms such as Expectation Maximization (EM) and return the most likely ("Viterbi") alignments under their model. For stability, alignment is usually performed in both directions (source-to-target and target-to-source) and the resulting alignments are symmetrized using an established heuristic (Och and Ney, 2004). In this thesis we use GIZA++(Och and Ney, 2003) and `fast_align` (Dyer et al., 2013) to create word alignments from parallel data. The parameters of the alignment models usually form a lexical translation table that can be used for context-free translation of query terms in CLIR as described in Section 1.5.1.4.

Extraction of Translation Units. Minimal bilingual translation units, such as phrases (Koehn et al., 2003) or synchronous grammar rules (Chiang, 2007), are extracted from the parallel data if they are consistent with the word alignments (Koehn et al., 2007; Chiang, 2007). These bilingual units form a phrase table (or grammar) of the translation model, where each phrase pair or translation rule is enriched with a set of "dense" features describing the (maximum likelihood) probability of translation from source to target, from target to source, and the individual likelihood of words contained in the unit ("lexical weighting"). Other common features are language models for the target language that ensure fluency of the output, as well as word penalty features that control the length of the output.

Parameter Optimization. Parameter weights $\mathbf{w}$ of the log-linear model (1.27) are optimized such that translation outputs match human reference translations. A variety of optimization techniques for SMT have been presented in the past. Common algorithms are Minimum Error Rate Training (MERT), Margin Infused Relaxed Algorithm (MIRA), Pairwise Ranking Optimization (PRO), and others. MERT (Och, 2003) directly optimizes BLEU (see below) using a line-search method, but is restricted to models with only a handful of (dense) features. We will use MERT in Chapter 2. PRO (Hopkins and May, 2011) optimizes translation outputs via pair-wise ranking of translation hypotheses. MIRA (Chiang, 2012) is an online discriminative training algorithm applicable to models with large amounts of (sparse) features and thus suited for our approaches in Chapter 3 and for pre-training SMT weights in Chapter 5.

Evaluation. We use BLEU (Papineni et al., 2002) to evaluate translation outputs of an SMT model against a set of reference translations. BLEU measures the geometric mean of n-gram precisions p_n of translation outputs $\bar{e}_i$ with respect to their reference translations r_i in a corpus of system outputs and references[10], $R = \{(\bar{e}, r)\}_1^M$:

$$BLEU = BP \cdot \prod_{i=1}^{n} p_i, \tag{1.29}$$

where BP is a brevity penalty that penalizes short translations, computed over the whole corpus R:

$$BP = \begin{cases} 1 & \text{if } l_c > l_r \\ exp(1 - \frac{l_r}{l_c}) & \text{if } l_c \leq l_r \end{cases},$$

with $l_r = \sum_{\bar{e},r \in R} |r|$ and $l_r = \sum_{\bar{e},r \in R} |\bar{e}|$. The n-gram precisions p_n are computed from counting n-gram matches in $\bar{e}$ w.r.t r over the whole corpus. Counts are clipped to the number of occurrences of an n-gram in the reference to penalize long translations that overproduce words. n-gram precisions are usually computed up to $n = 4$.

Approximate Randomization Tests for Statistical Significance. Similar to intrinsic IR evaluation with retrieval measures, we compare BLEU scores of two SMT

[10] We disregard multiple references for simplicity here.

systems A and B by testing the absolute difference for statistical significance using approximate randomization tests (Noreen, 1989; Riezler and Maxwell, 2005). Here, the test statistic is the absolute value of the BLEU score difference produced by two systems on the same test set. Since BLEU is a corpus-based metric, we randomly shuffle translation outputs between both systems with probability 0.5 and re-compute corpus BLEU. Let S be the number of random shuffles. The two-sided p-value is given by the number of times the absolute score differences of shuffled outputs is greater or equal than the observed test statistic:

$$p = \frac{\sum_{i=1}^{S} [\![\delta'_i \geq \delta]\!]}{S}$$

If the p-value lies below our specified rejection level, we can reject the null hypothesis, concluding that both systems are significantly different from each other. Rejection levels are given in the description of corresponding experiments.

Chapter 2

Mate Ranking: SMT Domain Adaptation through Cross-Language Information Retrieval

Abstract

Microblogging services such as Twitter have become popular media for real-time user-created news reporting. Such communication often happens in parallel in different languages, e.g., microblog posts related to the same events of the Arab spring were written in Arabic and in English. The approaches presented in this chapter aim to exploit this parallelism in order to eliminate the main bottleneck in automatic Twitter translation, namely the lack of bilingual sentence pairs for training Statistical Machine Translation (SMT) systems. We propose an initial batch approach to show that translation-based Cross-Language Information Retrieval (CLIR) can retrieve (millions of) microblog messages across languages that are similar enough to be used to train a standard phrase-based SMT pipeline. This method outperforms other approaches to domain adaptation for SMT such as language model adaptation, meta-parameter tuning, or self-translation. It provides an extrinsic evaluation study on the use of CLIR for mate-ranking, that is retrieving additional training data for SMT by finding translations or comparable texts ("cross-lingual mates") to the query. Furthermore, we present an extension to this method that iteratively alternates between the tasks of retrieval and translation and allows an initial general-domain model to incrementally adapt to in-domain data. Domain adaptation is done by training the translation system on a few thousand sentences retrieved in the step before. This alternating setup is time- and memory-efficient and of similar quality as batch adaptation on millions of parallel sentences. Both methods have been published in Jehl et al. (2012) and Hieber et al. (2013).

Author's Contributions

- Keyword-based crawling of Twitter messages for Arab Spring related events.
- Implementation of a mate-finding retrieval model on top of Lucene and Hadoop, including the concept of "self-translation".
- Research and development for context-sensitive SMT-based retrieval (Section 2.4.1). Almost identical research was published beforehand by Ture et al. (2012a,b), which are hence referenced as the original work in the following.
- An iterative algorithm to alternate between SMT re-training and CLIR steps using an updated SMT model to provide time-and memory-efficient adaptation effects, equal to a time-consuming batch approach.

2.1 Comparable Data Mining on Twitter

Among the various social media platforms, microblogging services such as Twitter[1] have become popular communication tools. This is due to the easy accessibility of microblogging platforms via Internet or mobile phones, and due to the need for a fast mode of communication that microblogging satisfies: Twitter messages are short (limited to 140 characters) and simultaneous (due to frequent updates by prolific microbloggers). Twitter users form a social network by "following" the updates of other users, either reciprocal or one-way. The topics discussed in Twitter messages range from private chatter to important real-time witness reports.

Events such as the Arab spring have shown the power and also the shortcomings of this new mode of communication. Microblogging services played a crucial role in quickly spreading the news about important events, furthermore they were useful in helping organizers plan their protest. The fact that news on microblogging platforms is sometimes ahead of newswire is one of the most interesting facets of this new medium. However, while Twitter messaging is happening in multiple languages, most networks of "friends" and "followers" are monolingual and only about 40% of all messages are written in English[2]. One solution to sharing news quickly and internationally was crowdsourcing manual translations, for example at Meedan[3]. Meedan is a nonprofit organization built to share news and opinions between the Arabic and English speaking world, by translating articles and blogs, using machine translation and human expert corrections.

An automated translation of microblogging messages is facing two main problems. First, there are no bilingual sentence pair data from microblogging domains available. Statistical Machine Translation however, crucially relies on large amounts of bilingual data (Brown et al., 1993b). Second, the colloquial, non-standard language of many microblogging messages, often interspersed with markup and dialectal vocabulary, makes it very difficult to adapt a machine translation system trained on any of the available bilingual resources such as transcriptions from political organizations or news text.

The approaches presented here aim to exploit the fact that microblogging often

[1] `http://twitter.com/`. last access: April 26, 2015

[2] `http://semiocast.com/publications/2011_11_24_Arabic_highest_growth_on_Twitter`. last access: April 26, 2015

[3] `http://news.meedan.net`. last access: April 26, 2015

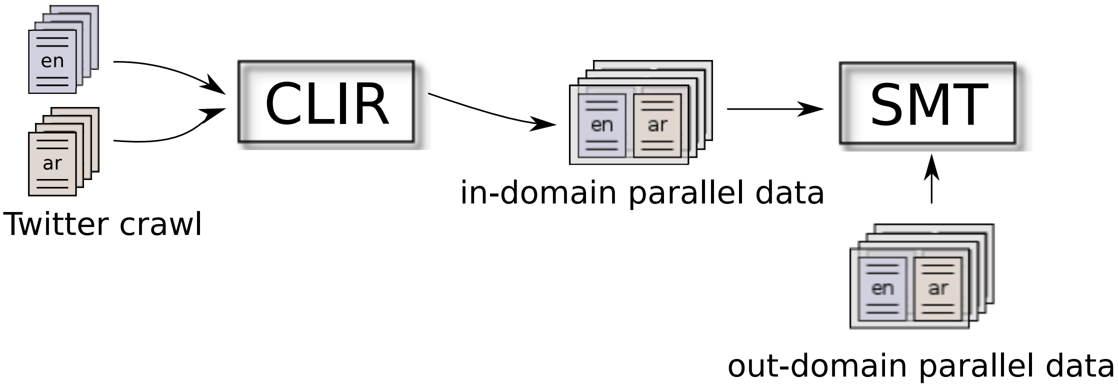


Figure 2.1: Comparable data mining for Twitter using Cross-Language Information Retrieval.

happens in parallel in different languages, e.g., microblog posts related to the same events of the Arab spring were published in parallel in Arabic and in English. In line with the greater focus of this thesis, we view this as a Cross-Language Information Retrieval problem, where the task consists of finding parallel or comparable microblogging messages to provide additional in-domain training data. The central idea, as illustrated in Figure 2.1, is to crawl a large set of topically related Arabic and English microblogging messages (i.e. a *Twitter Crawl*), and use Arabic microblog messages as search queries in a CLIR setup. The pairs of search queries and retrieved documents are then filtered and used as additional input (*in-domain parallel data*) to a standard SMT pipeline to train translation models adapted to the non-standard language of the microblogging domain.

We will present two approaches to mine and evaluate the use of comparable data from Twitter: (1) A single adaptation step (Section 2.2) using all data from the cross-product of both Arabic and English microblogging messages, and (2) an iterative method that alternates between the steps of retrieval and (re-)training of the SMT model. The latter illustrates comparable adaptation performance with less in-domain data per iteration (Section 2.4).

2.2 Batch SMT Domain Adaptation with Translation-based CLIR

In a first approach we carry out a single retrieval step with the probabilistic translation-based retrieval model of Xu et al. (2001) that naturally integrates translation tables

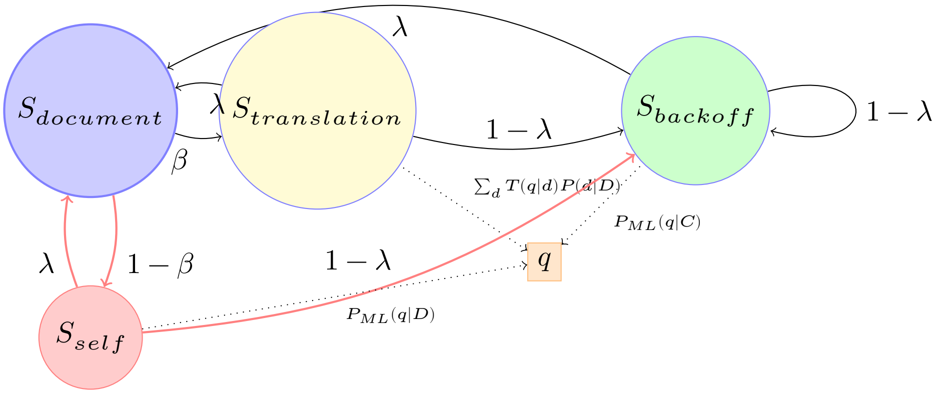


Figure 2.2: Translation-based CLIR as a Hidden Markov Model (Xu et al., 2001).

for cross-language information retrieval. For this approach to be successful, retrieval precision is key such that pairs of queries and retrieved documents consist of at least comparable segments. We investigate several techniques for filtering retrieval output and improving phrase extraction on noisy data (Munteanu and Marcu, 2006; Snover et al., 2008). We find a straightforward application of phrase extraction from symmetrized alignments to be optimal. These post-retrieval filtering techniques are relevant for an extrinsic evaluation of the retrieval model, but not subject of this thesis and part of the author's contributions. For the sake of completeness in presenting the evaluated method, and to establish the notion of mutual dependency between SMT and CLIR, they are included here nevertheless. The close dependency between retrieval and translation performance will then motivate an iterative approach (Section 2.4), published in Hieber et al. (2013), where an updated SMT model directly influences retrieval quality in subsequent CLIR steps.

2.2.1 A Word-based Cross-lingual Retrieval Model

In order to select comparable candidates for domain adaptation, we draw from the probabilistic retrieval framework (Section 1.5.1.3), that is, we estimate the probability of a relevant microblog message d given a query microblog message q, $P(d|q)$:

$$score(q,d) = P(d|q) = \frac{P(d)P(q|d)}{P(q)} \approx P(q|d). \tag{2.1}$$

$P(d|q)$ is reduced to $P(q|d)$, when a uniform prior over documents, $P(d)$, is used and we note that $P(q)$ is constant and thus does not affect the ranking. Under assumption of term independence, the quantity $P(q|d)$ is factorized as follows:

$$score(q,d) = P(q|d) = \prod_{t \in q} P(t|d) = \sum_{t \in q} logP(t|d), \tag{2.2}$$

where $P(t|d)$ is a linear interpolation between a mixture model and a maximum likelihood estimate of term importance in the query language L_q:

$$P(t|d) = \lambda P_{mix}(t|d) + (1-\lambda) \underbrace{\hat{P}(t|L_q)}_{\text{backoff}}, \tag{2.3}$$

$$P_{mix}(t|d) = \underbrace{\beta \sum_{u \in d} T(t|u)\hat{P}(u|d)}_{\text{translation}} + (1-\beta) \underbrace{\hat{P}(t|d)}_{\text{self-translation}} . \tag{2.4}$$

The model is closely related to a *query likelihood model* (Ponte and Croft, 1998) and its monolingual statistical translation approach by Berger and Lafferty (1999). Xu et al. (2001) have extended this to the cross-lingual setting by adding a term translation table, T. They describe their model in terms of a Hidden Markov Model with two states that generate query terms (Figure 2.2):

1. A *document state* generating document terms u and translating them to query term t.
2. A *backoff state* generating query terms t directly in the query language L_q.

In the *document state* the probability of emitting t depends on all u that translate into t, according to distribution T. This is estimated by marginalizing out u as $\sum_u T(t|u)P(u|d)$. In the *backoff state* the probability $\hat{P}(t|L_q)$ of emitting a query term is estimated as the relative frequency of this term within some corpus in the query language L_q. The probability of transitioning into document state or backoff state is given by λ and $1-\lambda$.

We view this model from a smoothing perspective where the backoff state is linearly interpolated with the translation probability using a mixture weight λ to control the weighting between both terms. Furthermore, we expand the generative model of Xu et al. (2001) to incorporate the concept of "self-translation", introduced by Xue et al. (2008) in a monolingual question-answering context: Twitter messages across languages usually share relevant terms such as hashtags, named entities or user mentions. Therefore, we model the event of a query term literally occurring in the document in a separate model that is itself linearly interpolated with a parameter β with the translation model.

The model is implemented based on a Lucene[4] index, which allows efficient storage of term-document and document-term vectors. To minimize retrieval time, we consider only those documents as retrieval candidates where at least one term translates to a query term, according to the translation table T. Stopwords were removed for both queries and documents. Compared to common inverted index retrieval implementations, the described model is quite slow since the document-term vectors have to be loaded. However, multi-threading and an implementation for MapReduce on a Hadoop cluster make the model tractable.

2.2.2 Filtering Retrieval Results for Precision

A retrieval function such as in the previous section induces a total ordering on the set of documents for each query, and documents positioned low in the ranking are unlikely to be comparable to the query. To select appropriate training data for Statistical Machine Translation, we conduct an additional filtering step to reduce noise in the data.

In order to extract phrases from retrieved in-domain data, we first run full-scale cross-lingual retrieval in both directions: retrieving Arabic documents using English microblog messages as queries and vice versa. For each run we keep the top k retrieved documents. Each document is then paired with its query to generate pseudo-parallel data. We explore two approaches on this kind of data to perform domain adaptation on a baseline MT system: (E1) A method resembling the work of Munteanu and Marcu (2006) that makes use of the translation table T employed by the retrieval step. An alignment point between a query term q and a document term d is created, if and only if $T(q|d)$ or $T(d|q)$ exist in the translation tables $D \to Q$ or $Q \to D$. Based on such word-alignments, we extract phrases by applying the *grow-diag-final-and* heuristic and using the phrase extraction algorithm of Och and Ney (2004) as implemented in `Moses`[5] (Koehn et al., 2007). While this method only induces alignment points in the vicinity of existing or known alignment points through the *grow-diag-final-and* heuristic, we also try a bolder approach by simply treating retrieval data as parallel and running unsupervised word alignment (E2). In contrast to previous work on longer texts (Snover et al., 2008; Daumé and Jagarlamudi, 2011), we can take advantage of the sentence-like character of microblog messages and treat queries and retrieval results

[4] `http://lucene.apache.org/core/`. last access: April 26, 2015

[5] `http://statmt.org/moses/`. last access: April 26, 2015

al-Gaddafi, al-Qaddhafi, assad, babrain, bahrain, egypt, gadaffi, gaddaffi, gaddafi, Gheddafi, homs, human rights, human-rights, humanrights, libia, libian, libya, libyan, lybia, lybian, lybya, lybyan, manama, Misrata, nabeelrajab, nato, oman, PositiveLibyaTweets, Qaddhafi, sirte, syria, tripoli, tripolis, yemen;

Table 2.1: Twitter crawl keywords.

similar to sentence aligned data.

For both extraction methods, the standard five translation features from the new phrase table (phrase translation probability, lexical weights in both directions, and a phrase penalty) were added to the translation features in `Moses`. We tried different modes of combining the new and original phrase table, namely using either one, or using the new phrase table as backoff in case no phrase translation is found in the original phrase table.

2.2.3 Data: Keyword-based Crawling of Twitter Messages

In order to obtain large amounts of in-domain candidate messages for retrieval, we crawled Twitter messages from September 20, 2011 until January 23, 2012 via the Twitter Streaming API[6] in keyword-tracking mode, obtaining 25.5M (*tweets*) in various languages. Table 2.1 shows the list of manually chosen keywords to retrieve microblog messages related to the events of the Arab spring. The Twitter Streaming API allows up to 400 tracking keywords that are matched to uppercase, lowercase and quoted variations of the keywords. Partial matching such as "tripolis" matching "tripoli" as well as Arabic Unicode characters are not supported.[7] We extended our keywords over time by analyzing initial crawl results, for example, by introducing spelling variants and hashtags.

Language identification. To separate microblog messages by languages, we applied a Naive Bayes language identifier[8]. This yielded a distribution with the six most common languages (of 52) being Arabic (57%), English (33%), Somali (2%), Spanish (2%), Indonesian (1.5%), and German (0.7%). We retained only microblog messages classified as English or Arabic with a classification confidence greater 0.9. Keyword-based

[6] `https://dev.twitter.com/docs/streaming-api/`. last access: April 26, 2015

[7] The restriction to ASCII characters has been removed by the time of writing this thesis.

[8] Language Detection Library for Java, by Shuyo Nakatani (`http://code.google.com/p/language-detection/`). last access: April 26, 2015

Bahrain (4991110), bahrain (3393085), Egypt (2570836), Syria (2239425), egypt (1095182), 14feb (828342), tahrir (706293), syria (699127), Libya (630068), Tahrir (566910), 14Feb (554794), jan25 (458570), ksa (404909), Yemen (382962), alwefaq (382752), feb14 (381768), kuwait (320879), BAHRAIN (301644), Jan25 (280848), GCC (259526);

Table 2.2: Twitter crawl: 20 most common hashtags.

	Arabic	English
tweets + retweets	14,565,513	8,501,788
tweets	6,614,126	5,129,829
avg. retweet/tweet	11.62	7.27
unique users	180,271	865,202
avg. tweets/user	36.6	5.9

Table 2.3: Twitter crawl statistics.

crawling creates a strong bias towards the domain of the keywords and it is not guaranteed that all microblog messages regarding a certain topic or region are retrieved, or that all retrieved messages are related to the Arab Spring and human rights issues in the middle east. Furthermore, the constraint to ASCII characters in keyword selection minimizes the ability to receive Arabic tweets by matching Arabic terms within sentences. Thus, the majority of microblog messages classified as Arabic were crawled due to hashtag matching. Table 2.2 displays the 20 most common hashtags in the data, which indicate a topical bias towards events related to Arab Spring.

Duplicate removal. Additionally, retweets artificially inflate the size of the data, and do not constitute relevant candidates for comparable data. Therefore, we removed all duplicate retweets that did not introduce additional terms to the original tweet. Table 2.3 explains the shrinkage of the data set after removing retweets - compared to English users, a smaller number of Arabic users produced a much larger number of retweets. Supporting the initial hypothesis of bilingual Twitter usage, about 56,087 users in the crawl tweet a substantial amount in both languages.

Preprocessing. Preprocessing removed Twitter markup strings and used several preprocessing steps such as digit normalization. The Arabic side of the data was transliterated using the Buckwalter Arabic transliteration scheme[9] and preprocessed using the MADA+Tokan toolkit (Habash et al., 2009). See Jehl et al. (2012) for more details.

In-domain test corpus. For evaluating domain adaptation performance with crawled and retrieved in-domain comparable data, an in-domain Twitter test set with references is required. Due to the lack of such a set, an evaluation corpus was created with Amazon Mechanical Turk[10], following exploratory work of Zaidan and Callison-Burch (2009): A random sample of 2,000 Arabic microblog messages was selected from the crawl and each message was translated by three different Turkers to form an in-domain evaluation set with three reference translations per input sentence. To avoid artificially inflating BLEU scores at test time, messages with large portions of Twitter markup were removed, yielding a final set of 1,022 parallel sentences. The evaluation set was further split in half, one for development and one for testing. More details on the design of Human Intelligence Tasks (HITs) are given in Jehl et al. (2012).

2.2.4 Experiments & Extrinsic Evaluation: Twitter Translation

To evaluate our strategy of using CLIR to extract comparable data in the Twitter domain, we conducted a series of experiments contrasting standard ways of domain adaptation (Bertoldi and Federico, 2009) with our technique of data mining using retrieval.

Baseline SMT system. All experiments were conducted using the `Moses` machine translation system[11] (Koehn et al., 2007) with standard settings. Language models were built using the SRILM toolkit[12] (Stolcke, 2002). For all experiments, we report lowercased BLEU-4 scores (Papineni et al., 2002) as calculated by `Moses`' `multi-bleu` script. For assessing significance, we apply the approximate randomization test (Noreen, 1989; Riezler and Maxwell, 2005). We consider pairwise differing results scoring a p-value < 0.05 as significant. The baseline model was trained using 5,823,363 parallel sentences in Modern Standard Arabic (MSA) (198,500,436 tokens) and English (193,671,201 tokens) from the NIST[13] evaluation campaign. This

[9] `http://www.qamus.org/transliteration.htm`. last access: April 26, 2015

[10] `http://www.turk.com`. last access: April 26, 2015

[11] `http://statmt.org/moses/`. last access: April 26, 2015

[12] `http://www.speech.sri.com/projects/srilm/`. last access: April 26, 2015

[13] National Institute of Standards and Technology

run	translation model	language model	dev set	BLEU
1	NIST	NIST	NIST	13.90
2	NIST	NIST	Twitter	14.83*
3	NIST	Twitter	NIST	15.98*
4	NIST	Twitter	Twitter	15.68*
5	NIST	Twitter & NIST	Twitter	16.04*
6	self-train	Twitter & NIST	Twitter	15.79*
7	self-train & NIST	Twitter & NIST	Twitter	15.94*

Table 2.4: Standard domain adaptation results. Asterisks indicate significant improvements over baseline (1).

data contains parallel text from different domains, including UN reports, newsgroups, newswire, broadcast news, and weblogs.

Standard domain adaptation techniques. The results of baseline domain adaptation experiments and their combinations are shown in Table 2.4. These experiments included the use of the in-domain development set for parameter optimization (Koehn and Schroeder, 2007), the building of an in-domain language model (Zhao et al., 2004), and a form of self-training (Ueffing et al., 2007): Arabic microblog messages were machine-translated using the system obtained from the first two adaptation techniques. This generated synthetic parallel data was then used to extract another phrase table.

Domain Adaptation using Translation-based CLIR. We ran retrieval in both directions, where Arabic and English microblog messages both served as queries or documents once. Meta-parameters $\lambda, \beta \in [0, 1]$ of the retrieval model (Section 2.2.1) were tuned in a *mate-finding* experiment on the in-domain development set. Mate-finding refers to the task of retrieving the single relevant document for a query. In our case, each Arabic tweet in the crowdsourced development set had exactly one "mate", namely the crowdsourced translation that was ranked best in a further crowdsourced ranking task (Jehl et al., 2012). Highest precision@1 scores (above 0.95) were achieved with parameters set to $\lambda = 0.9$, $\beta = 0.9$. For comparable data candidates we kept the top 10 returned documents per query from the retrieval step. The translation table T was taken from the baseline model described above.

We evaluate both filtering and extract techniques proposed in Section 2.2.2. When

run	Twitter phrases	extraction method	sentence pairs	extracted phrases	BLEU
8	top 3 retrieval results	E1	14,855,985	6,508,141	17.04*
9	top 1 retrieval results	E2	5,141,065	54,260,537	18.73**
10	retrieval intersection	E2	3,452,566	29,091,009	18.85**
11	retrieval intersection as backoff	E2	3,452,566	29,091,009	**18.93****

Table 2.5: CLIR domain adaptation results. All weights were optimized on the Twitter development set and used the Twitter and NIST language models. One asterisk indicates a significant improvement over baseline run (5) from Table 2.4. Two asterisks indicate a significant improvement over run (8). E1/E2 refer to the extraction methods explained in Section 2.2.2.

extracting a new in-domain phrase table we restrict the maximum phrase length to 3 (the default is 7), to avoid learning too much noise from the data. Results are shown in Table 2.5. For E1, we tried combinations of the following constraints to filter candidate pairs after retrieval:

1. number of alignment points in the candidate pair induced by baseline translation table T,
2. number of candidate pairs retained per query or intersecting results from both retrieval directions.

The largest improvement over standard domain adaptation techniques was obtained when requiring at least 3 alignment points in both directions while using the top 3 retrieval results per query (Table 2.5, run 8). We also found that selecting only the top 3 retrieval results was beneficial to performance, suggesting that translation-based retrieval scores and its induced rankings are indeed an indicator for comparable microblog messages. For the bolder approach of treating retrieval results as parallel, E2, we see more significant gains in BLEU, with the best configuration achieved when only intersected candidate pairs selected and the new phrase table is used in *backoff mode* during decoding (Table 2.5, run 11).

2.2.5 Adaptation Analysis

The described cross-lingual retrieval approach succeeded in finding nearly parallel tweets, confirming our hypothesis that such data actually exists. Some examples of

Arabic tweet	الصفح الى الليبيين ويدعوا سيحاكم القذافي ان يؤكد الفرنسي الرئيس ب ف ا
gloss	*AFP confirms that the French President Gaddafi Libyans tried to call and forgiveness*
English tweet	french president assures that will be taken to court and tells the libyans to forgive each other
Arabic tweet	الخميس من ا دء مصر فى المحمول شركات جميع رقم زيادة يقرر الاتصالات تنظيم جهاز
gloss	*NTRA decide to increase the number of all mobile operators in Egypt a commencement from Thursday*
English tweet	ntra decide to increase the number of all mobile operators in starting from thursday
Arabic tweet	ناري طلق طريق عن يناير يوم احمد على امين الشهيد
gloss	*Shahid Amin AA Day January through gunshot*
English tweet	martyr amin ali ahmed on jan by gunshot

Table 2.6: Adaptation analysis: Examples of nearly parallel Twitter messages found by the translation-based retrieval model. Glosses were obtained from Google Translate.

adaptation method	% OOV/abs.	1-gram precision/abs.	2-gram precision/abs.
None	22.56/2216	51.1/5020	20.2/1882
LM and Dev	20.05/2220	51.4/5442	22.1/2227
Retrieval (E1)	17.47/1790	53.5/5484	23.6/2299
Retrieval (E2)	4.22/439	56.1/5834	26.1/2575

Table 2.7: OOV-rate and uni/bi-gram precisions for evaluated adaptation methods. Numbers behind slashes are the absolute values.

query-document pairs are given in Table 2.6. Table 2.7 shows a more detailed breakdown of the BLEU scores presented in Table 2.5. Standard domain adaptation techniques, such as an in-domain language model and in-domain tuning, barely increase n-gram precision or reduce the number of Out-Of-Vocabulary words (OOVs). This is expected since those techniques do not add any new vocabulary to the model but rather fine-tune word choice of already known words. The retrieval-based adaptation techniques however, yield a significant reduction in OOVs and increase precision for uni- and bi-grams. In contrast to the heuristic approach (E1), the bold approach of treating the retrieved candidate pairs as parallel (E2) allows the phrase extraction process to learn new words that are more distant to known words. Clearly this improvements depends on the parallelism of the retrieved data and thus the performance of the retrieval model: Enforcing a stricter cutoff or intersection of retrieval results yields less but more valuable data, as seen in Table 2.5.

Nevertheless, the overall BLEU scores of the adapted system are still fairly low and translation quality as judged by inspection of the output can be very poor. This suggests that the language used on Twitter still poses a great challenge, due to its variety of styles as well as the users' tendency to use non-standard spelling and colloquial or dialectal expressions. Jehl et al. (2012) presents more details on dialect distributions and common lexical translation errors in the crowdsourced development and test sets.

2.3 Limitations of Singular Adaptation and Context-Free Query Translation

We have presented an approach to mine nearly parallel or comparable data from a large corpus of Twitter messages to adapt an existing SMT system for Twitter translation by extracting a new in-domain phrase table either based on alignments generated in the vicinity of known words, or treating candidate pairs as parallel for unsupervised word alignment. The data mining approach relies on a Cross-Language Information Retrieval model that makes use of a lexical translation table to map terms in two languages. This translation table is created as a side-product from the baseline SMT model training and is thus bound to lexical knowledge of baseline SMT system and its general-domain data. Since the retrieval function only orders documents in the collection by scores and lacks a component of classifying parallelism, one must define precision-oriented constraints to ensure parallelism or comparability of the returned candidate pairs in a post-retrieval step. Still, the mined data contains a lot of noise and a positive adaptation result for method E2 may not always be guaranteed. One way to approach this, is to incrementally adapt the SMT system by performing smaller adaptation steps while iteratively re-training the SMT model. Since the retrieval step depends on lexical coverage of the SMT system, one can imagine a bootstrapping process to incrementally and jointly adapt both SMT and IR models in such a way. For example, by first adding new words in the vicinity of known words more conservatively, one can then re-run retrieval with this updated translation model to either learn more *distant* words or boost the retrieval score of existing candidate pairs which where previously not positioned in the top k. With this iterating scheme, smaller portions of found in-domain data can be used per iteration. In the next section we will propose such an iterative approach to domain adaptation for Twitter translation.

Besides the static nature of a single adaptation step in Figure 2.1, the IR model in Section 2.2.1 also considers only *context-free* translation options as given by the word-

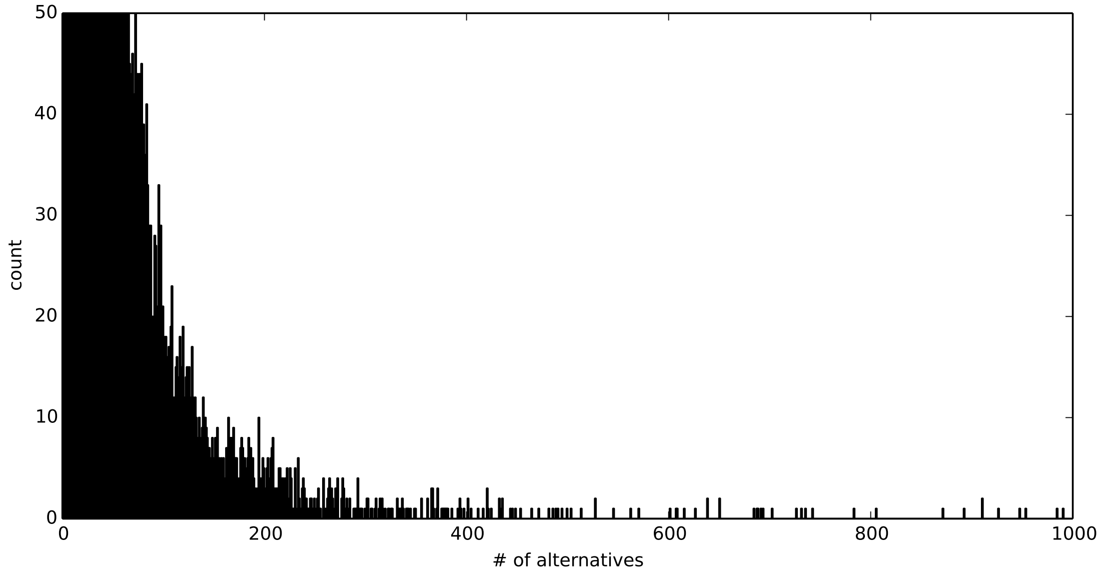


Figure 2.3: Histogram of the number of translation alternatives per source term in a lexical translation table given by the standard IBM alignment models. The y-axis is limited to 50; the maximum is 65,639; the average number of translations per source term is 8.05.

based translation table. While this restriction to uni-grams enables efficient matching of a variety of translation alternatives, the distributions usually contain a lot of noise and have high entropy. Figure 2.3 shows a histogram of the number of translation alternatives per source term. The y-axis is limited to a count of 50 to be able to display the long tail of words with more than 600 alternatives. The size of the source language vocabulary of the translation table used for this graph was 138,545, out of which 65,639 terms have only one translation. However, rare words tend to align to a lot of (noisy) alternatives. For example, there exists one term that translates to 41,402 target terms. The average number of alternatives for a source term is 8.05. This phenomenon of noisy alignments for rare words is commonly known as *garbage collection* in alignment models (Brown et al., 1993a; Moore, 2004; Och and Ney, 2003).

Clearly, a retrieval model that uses such noisy translation alternatives is prone to mistakes due to translation errors. The linear interpolation of the mixture model with the general uni-gram probability of the source term in Equation 2.4 may reduce this problem, but the general danger of propagating translation errors to the retrieval step remains for word-based CLIR models since there is no surrounding context taken into account. Obviously, this is not the case for phrase-based or hierarchical phrase-based

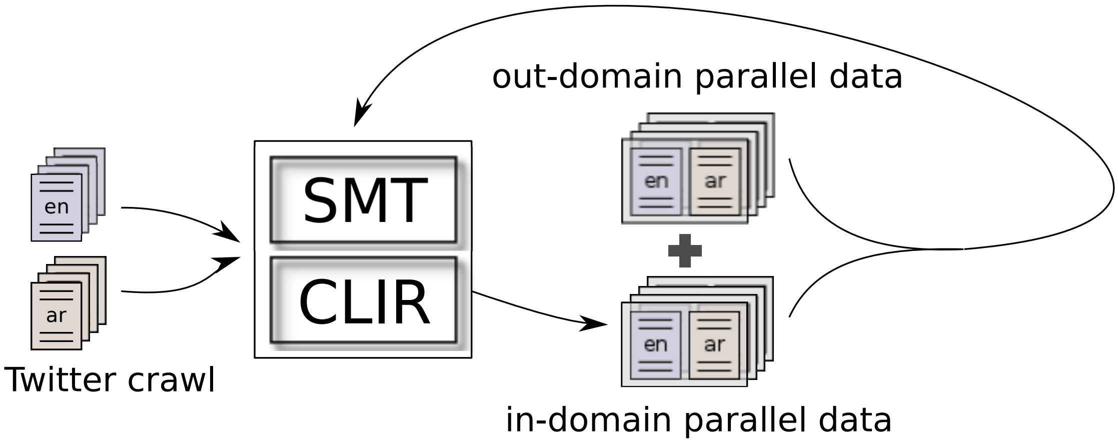


Figure 2.4: *Iterative* comparable data mining for Twitter using CLIR: the adapted SMT system is immediately used for CLIR in the next iteration.

SMT systems where the minimal translation units consist of (hierarchical) phrases and higher-order language models ensure fluency of the output. Thus, a translation-based retrieval system, especially when used for *mate-ranking*, should make use of higher-order translation models to filter out unlikely translation options through context sensitivity and thus reduce the danger of *query drift*. The next section presents a CLIR approach based on this idea by suggesting the use of Probabilistic Structured Queries, where translation alternatives are additionally selected from full-fledged Statistical Machine Translation output. Chapter 3 describes an approach to optimize query translations obtained from an SMT decoder for the retrieval task by discriminative training. Chapter 5 extends the idea of introducing the retrieval objective into the task of translation by forcing the decoding process to accommodate for matching terms in retrieval and thus using the SMT decoder directly as a retrieval function.

2.4 Mutual, Iterative Adaptation of Retrieval and Translation

Most approaches to mining comparable data, such as the one by Munteanu and Marcu (2005) and the method described in Section 2.2, try to tackle the noise inherent in automatically extracted parallel data by sheer size. However, finding good quality parallel data from noisy resources like Twitter requires sophisticated retrieval methods, complex post-retrieval filtering steps (Munteanu and Marcu, 2006), or alignment models robust to noise (Vaswani et al., 2012; Mermer et al., 2013). Running these methods on millions of queries and documents can take weeks, and training an SMT pipeline from several millions of sentence pairs is costly.

The method described in this section aims to achieve improvements similar to large-scale parallel sentence extraction approaches as described in Section 2.2, while requiring only a fraction of the extracted data (orders of magnitude smaller) and considerably less computing resources. Our key idea is to extend a straightforward application of translation-based CLIR to an iterative method: Instead of attempting to retrieve in one step as many parallel sentences as possible, we allow the retrieval model to gradually adapt to new data by using an SMT model trained on the freshly retrieved sentence pairs in the translation-based retrieval step. We alternate between the tasks of translation-based retrieval of target sentences, and the task of SMT, by re-training the SMT model on the data that were retrieved in the previous step. This task alternation is done iteratively until the number of newly added pairs stabilizes at a relatively small value.

In the following experiments on Arabic-English Twitter translation, we achieved improvements of over 1 BLEU point over a strong baseline that uses in-domain data for language modeling and parameter tuning. Compared to a CLIR approach which extracts more than 3 million parallel sentences from a noisy comparable corpus, the system produces similar results in terms of BLEU using only about 44 thousand sentences for training in each of a few iterations, thus being much more time- and resource-efficient.

In the terminology of semi-supervised learning (Abney, 2008), our iterative method resembles self-training and co-training by training a model on its own predictions. It is different in the aspect of task alternation: The SMT model trained on retrieved sentence pairs is not used for generating training data, but for scoring noisy parallel data in a translation-based retrieval setup. Our method also incorporates aspects of transductive learning in that candidate sentences used as queries are filtered for Out-of-Vocabulary words and similarity to sentences in the development set in order to maximize the impact of translation-based retrieval. Our work most closely resembles approaches that make use of variants of SMT to mine comparable corpora for parallel sentences. Recent work uses word-based translation (Munteanu and Marcu, 2005, 2006), full-sentence translation (Abdul-Rauf and Schwenk, 2009; Uszkoreit et al., 2010), or a sophisticated interpolation of word-based and contextual translation of full sentences (Snover et al., 2008; Jehl et al., 2012; Ture and Lin, 2012) to project source language sentences into the target language for retrieval. The novel aspect of iterative task alternation introduced in this section can be applied to all approaches that incorporate Statistical Machine Translation for sentence retrieval from comparable data. For our baseline system we use in-domain language models (Bertoldi and Federico, 2009) and meta-parameter tuning on in-domain development sets (Koehn and Schroeder, 2007).

2.4.1 Context-sensitive Query Translation for CLIR

The key difference to the CLIR model of Equation 2.4 is the use of a fully trained Statistical Machine Translation model for query translation. While translation options in the previous approach were given by a lexical translation table, we additionally select translation options estimated from the decoder's n-best list for translating a particular query. The central idea is to let the language model choose fluent, context-aware translations for each query term during decoding. This is especially important to the given mate-ranking task where microblogging messages constitute coherent natural language queries and are not an unordered list of keyword search terms.

Probabilistic Structured Queries (PSQ). For mapping source language query terms to target language query terms, we follow Ture et al. (2012a,b). Given a source language query q with query terms t, we project it into the target language by representing each source token t by its probabilistically weighted translations. The score of target document d is computed by calculating the Okapi $BM25$ rank (Robertson et al., 1998) (see Section 1.5.1.3) over projected term frequency and document frequency weights as follows:

$$score(q,d) = BM25(q,d) = \sum_{t \in q} bm25(tf(t,d), df(t)) \tag{2.5}$$

$$tf(t,d) = \sum_{u \in T_t} T(u|t) tf(u,d) \tag{2.6}$$

$$df(t) = \sum_{u \in T_t} T(u|t) df(t) \tag{2.7}$$

where $T_t = \{u|T(u|t) > L\}$ is the set of translation options for query term t with probability greater than L. Likewise to recent work on Probabilistic Structured Queries (Ture et al., 2012a,b), we impose a cumulative threshold C, so that only the most probable options are added to T_t until C is reached. Note that, in contrast to the generative story of the Hidden-Markov model in Equation 2.4, the translation direction is reversed and we can interpret term frequency and document frequency weights as *expectations* under translation model T. This framework of Probabilistic Structured Queries (Darwish and Oard, 2003) differs from the generative story of language model-based retrieval models as presented in Section 2.2.1 and 1.5.1.4 and can easily encode weighted query term alternatives from different sources. We can view the (cross-lingual) query as a weighted Bag-of-Words object, representing probabilistically weighted translation options for each query term, for example:

$$\begin{aligned} PSQ(q) = \{auge : \{&(eye, 0.8), (mind, 0.15), (focus, 0.05)\}, \\ schwarze : \{&(schwarze, 0.01), (black, 0.99)\}\} \end{aligned}$$

Analog to findings of Ture et al. (2012a,b), we achieved best retrieval performance when translation probabilities were calculated as an interpolation between (context-free) lexical translation probabilities T_{lex} estimated on symmetrized word alignments, and (context-aware) translation probabilities T_{nbest} estimated on the n-best list of an SMT decoder:

$$T(u|t) = \lambda T_{nbest}(u|t) + (1 - \lambda) T_{lex}(u|t) \tag{2.8}$$

$P_{nbest}(t|q)$ is the decoder's confidence to translate t into u within the context of query q. Let $a_k(u, t)$ be a function indicating alignment of target term u to source term t in the k-th derivation of query q. We can use the n-best list of the decoder as data for a maximum likelihood estimate for lexical translation probability $T_{nbest}(u|t)$:

$$T_{nbest}(u|t) = \frac{\sum_{k=1}^{n} a_k(u, t)\mathcal{D}(k, q)}{\sum_{k=1}^{n} a_k(\cdot, t)\mathcal{D}(k, q)} \tag{2.9}$$

where $\mathcal{D}(k, q)$ is the model score of the k-best derivation for query q.

Intuitively, the probabilistic weights assigned to terms selected from the n-best list output depend on the surrounding context of the full derivations produced by the SMT decoder. Thus, λ controls the balance between context-free *query expansion* (T_{lex}) to reduce "lexical chasms" (Berger et al., 2000), and context-sensitive alternatives selected from the top-n phrase-based translations to reduce the danger of *query drift* (T_{nbest}).

We implemented this model on top of the hierarchical phrase-based translation framework (Chiang, 2007) as implemented by `cdec` (Dyer et al., 2010). This allows us to extract alternatives from the n-best outputs using the word alignments between source and target words for q encoded in the Synchronous Context-Free Grammar

Algorithm 1 Task Alternation.

```
Require: source language Tweets $Q_{src}$, target language Tweets $D_{trg}$, general-domain parallel sentences
  $S_{gen}$, general-domain SMT model $M_{gen}$, interpolation parameter $\theta$

  procedure TaskAlternation($Q_{src}, D_{trg}, S_{gen}, M_{gen}, \theta$)
      $t \leftarrow 1$
      while true do
          $S_{in} \leftarrow \emptyset$                          ▷ Start with empty parallel in-domain sentences
          if $t == 1$ then
              $M^{(t)}_{clir} \leftarrow M_{gen}$                ▷ Start with general-domain SMT model for CLIR
          else
              $M^{(t)}_{clir} \leftarrow \theta M^{(t-1)}_{smt} + (1-\theta) M^{(t)}_{smt}$ ▷ mixture of previous and current SMT model for CLIR
          end if
          $S_{in} \leftarrow$ Clir($Q_{src}, D_{trg}, M^{(t)}_{clir}$)      ▷ Retrieve top-1 target language Tweets
          $M^{(t+1)}_{smt} \leftarrow$ Train($S_{gen} + S_{in}$)          ▷ Train SMT model on concatenated data
          $t \leftarrow t + 1$
      end while
  end procedure
```

rules of the derivations. The concept of *self-translation* as introduced in Equation 2.4 is covered by the decoder's ability to generate *pass-through* rules for unknown words or phrases: If an unknown word is a named entity, it will regularly occur in the n-best list untranslated and will hence receive a high translation weight in T_{nbest}. The code for creating Probabilistic Structured Queries from `cdec` n-best lists is available at `https://github.com/fhieber/cclir`.

2.4.2 Incremental Adaptation by Task Alternation

We describe the idea of alternating between retrieval and SMT model re-training in the following: we allow the initial general-domain CLIR model to adapt to in-domain data over multiple iterations. Adaptation is carried out by re-training the SMT model on the concatenation of the general-domain data and the previously retrieved in-domain candidate pairs.

Algorithm 1 shows the iterative task alternation procedure: Retrieval in the first iteration $t = 1$ is carried out with a general-domain SMT model, trained without any in-domain data, M_{gen}. The retrieved comparable data from the target domain, consisting of pairs of queries and documents, $_{in}$ is used in concatenation with the baseline training data, S_{gen}, to train an adapted SMT model, $M^{(t+1)}_{smt}$. This adapted model is then used in the iteration to form a CLIR model, adapted to the target domain, to retrieve more comparable data.

In terms of semi-supervised learning (Abney, 2008), we can view Algorithm 1 as non-persistent as we do not keep labels from previous iterations: candidate query-document pairs from previous iterations are not used again in the following iterations. Instead we re-retrieve a new set of in-domain pairs in every iteration. Variations of label persistencies did not yield any improvements. Label persistency is usually implemented to prevent the training procedure to diverge too far from the initial baseline model. A similar effect of preventing the SMT model to "dissolve" general-domain knowledge across iterations is achieved by mixing models from current and previous iterations. This is accomplished in two ways: First, by linearly interpolating the translation option weights for Probabilistic Structured Queries $T(u|t)$ from the current and previous model with interpolation parameter θ. Second, by always using the $T_{lex}(u|t)$ parameters from the general-domain data S_{gen}.

Similar to Section 2.2 we find that imposing constraints on retrieval results yields better results. Thus we only create candidate pairs from queries and their top-1 ranked documents, which are then used as additional training data for the SMT model in each iteration.

2.4.3 Experiments & Extrinsic Evaluation: Twitter Translation

For evaluation of the iterative data mining approach, we re-use the Twitter corpus from Section 2.2.3. However, due to the use of a hierarchical phrase-based machine translation system for generating Probabilistic Structured Queries, we re-trained the original SMT model from `Moses` (Section 2.2) in the `cdec` framework. Thereby, we created a re-implementation of the standard domain adaptation baselines, that is, the use of an in-domain language model and parameter optimization on an in-domain development set.

We trained the general domain model M_{gen} on data from the NIST evaluation campaign, including UN reports, newswire, broadcast news and blogs. Since we were interested in relative improvements rather than absolute performance, and Algorithm 1 integrates a full, time-consuming batch training of the updated SMT model, we sampled 1 million parallel sentences S_{gen} from the originally over 5.8 million parallel sentences from Section 2.2.3. Adaptation performance is again measured in BLEU scores (Papineni et al., 2002) on the (crowdsourced) development and test sets of Jehl et al. (2012) (Section 2.2.3), which provides three references per Twitter message.

To further accommodate for the computationally expensive setup of iterative retrieval and SMT model training, we follow a transductive setup of query selection: We create a small set of queries Q_{src}, consisting of the source side of the evaluation data

	BLEU	# of in-domain sentences
Standard Domain Adaptation	14.05	-
Full-scale CLIR (section 2.2, Jehl et al. (2012))	14.97	3,198,913
Task alternation	**15.31**	**∼40,000**

Table 2.8: Standard domain adaptation with in-domain LM and tuning; Full-scale CLIR yielding over 3M in-domain parallel sentences; Task alternation ($\theta = 0.1$, iteration 7) using ∼40k parallel sentences per iteration. BLEU scores are given for the Twitter test set (511 sentences).

and similar Tweets. Similarity was defined by two criteria: First, we ranked all Arabic Tweets with respect to their term overlap with the development and test Tweets. Smoothed per-sentence BLEU (Lin and Och, 2004) was used as a similarity metric. For each input sentence, we kept the top 100 candidates. OOV-coverage served as a second criterion to remedy the problem of unknown words in Twitter translation. We first created a general list of all OOVs in the evaluation data under M_{gen} (3,069 out of 7,641 types). For each of the top 100 BLEU-ranked Tweets, we counted OOV-coverage with respect to the corresponding source Tweet and the general OOV list. We only kept Tweets containing at least one OOV term from the corresponding source Tweet and two OOV terms from the general list, resulting in 65,643 Arabic queries covering 86% of all OOVs. This reduced query set Q_{src} performed better (14.76 BLEU) after one iteration than a similar-sized set of random queries (13.39 BLEU).

We compare the iterative approach with the full-scale retrieval approach of Section 2.2 using a PSQ-based CLIR model as the baseline. It took 14 days to run 5.5M Arabic queries on 3.7M English documents. In contrast, the iterative approach of algorithm 1 completed a single iteration in less than 24 hours. PSQ-based retrieval was carried out in 4 batches of about 16,411 queries on a Hadoop cluster with 190 mappers. Each mapper loads the full set of interpolated Probabilistic Structured Queries and scores disjoint subsets of the documents. In the combine and reduce phase all ranked lists for all queries are merged across these subsets and the final rankings are returned.

In the absence of Twitter relevance data for retrieval optimization, we again selected the parameters $\lambda = 0.6$ (Equation 2.8), $L = 0.005$, and $C = 0.95$ in a mate-finding task. The size of the n-best list to estimate T_{nbest} was set to 1000. All SMT models included a 5-gram language model built from the English side of the NIST data plus the English side of the Twitter corpus D_{trg}. Word alignments were created

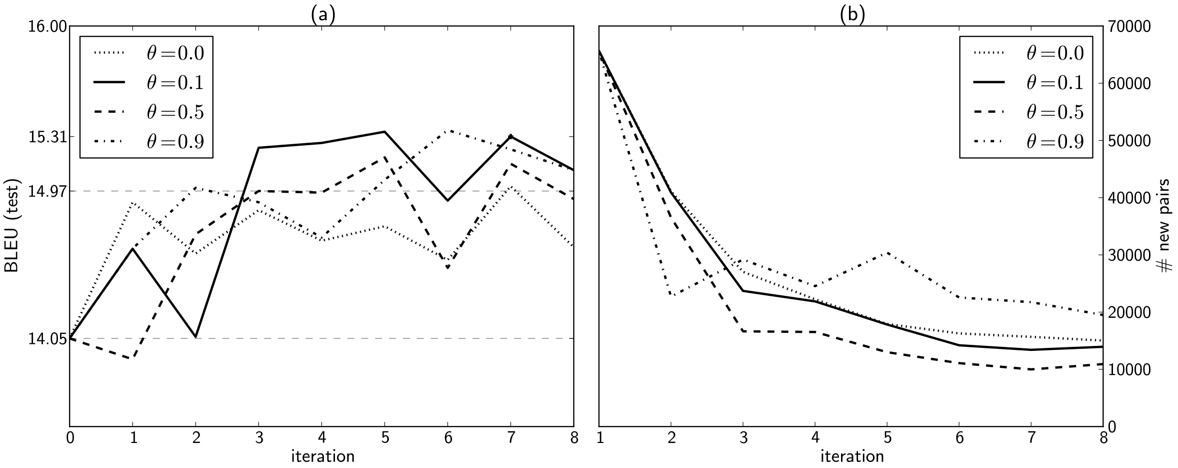


Figure 2.5: Learning curves for various θ. (a) BLEU scores and (b) number of new pairs added per iteration.

using GIZA++ (Och and Ney, 2003) with 5 iterations of IBM Model 1 and 2, and five iterations of the HMM (Vogel et al., 1996). Rule extraction and parameter tuning using Minimum Error Rate Training (Och, 2003) was carried out with the available `cdec` implementations, using standard features. We ran MERT five times per iteration, carrying over the weights which achieved median performance on the development set to the next iteration.

Table 2.8 reports median BLEU scores on the crowdsourced test set of a standard adaptation baseline with in-domain language model and development set, the full-scale retrieval approach, and the best result from our task alternation system. Approximate randomization tests (Noreen, 1989; Riezler and Maxwell, 2005) indicate that improvements of full-scale retrieval and task alternation over the baseline were statistically significant, confirming the findings of the previous section that mining for comparable data outperforms standard domain adaptation techniques. Differences in BLEU scores between full-scale retrieval and task alternation were not significant. Note that the results for the full-scale CLIR experiment are not directly comparable to those of Section 2.2, since our setup uses less than one fifth of the NIST data for initial SMT training, a different translation model, a new CLIR approach, and a different development and test split of the crowdsourced data.

Figure 2.5 illustrates the impact of θ, which controls the importance of the previous model compared to the current one, on median BLEU (a) and change of S_{in} (b) over iterations. For all θ, few iterations suffice to reach or surpass full-scale retrieval performance. Yet, no run achieved good performance after one iteration, showing that the transductive setup must be combined with task alternation to be effective. While we see fluctuations in BLEU for all θ-values, $\theta = 0.1$ achieves high scores faster and more consistently, thus pointing towards selecting a bolder updating strategy. This is also supported by plot (b), which indicates that choosing $\theta = 0.1$ leads to faster stabilization in the pairs added per iteration (S_{in}). We used this stabilization as a stopping criterion.

2.5 Limitations of Translation-Optimized SMT Models for CLIR

In this chapter we presented two methods for mining parallel sentence pairs from large amounts of user-generated, comparable data. The first method employed a full-scale retrieval approach using the cross-product of source and target Twitter messages. Secondly, we illustrated an iterative extension that allows us to obtain similar results with less in-domain training data at each iteration and in total. More importantly, the task alternation algorithm integrates a translation-based CLIR model that makes use of context-sensitive hierarchical phrase-based translation. Integrating an SMT model that is to be adapted into the mate-ranking step, allows a gradual adaptation to the target domain by alternating between the tasks of retrieval and SMT (re-)training on a few thousand parallel sentences retrieved in the step before. The number of new pairs added per iteration stabilizes to a few thousand after seven iterations, yielding an SMT model that improves 0.35 BLEU points over a model trained on millions of retrieved pairs.

While context-sensitive translation of query strings mitigates the danger of query drift, the translation component of the CLIR model remains agnostic about its use for retrieval. This entails that large amounts of modeling effort in the SMT system are spent on properties of the translation output that are not used during retrieval, namely fluency of the output and correct word ordering. However, during query evaluation in the previously described approaches, word order and stopwords are not considered by retrieval functions such as $BM25$ (see Section 1.5.1.3) or the language-model based system in Section 2.2.1. In the remainder of this thesis, we will argue that informing the SMT system about its use for retrieval not only allows direct optimization for the task at hand (see Chapter 3), but also the direct integration of the retrieval scoring function into the SMT decoder (Chapter 5). Thereby, we achieve significant gains over Direct

Translation and PSQ baselines (Section 2.4.1). Pursuing this direction of research using data-driven methods requires not only sufficient amounts of parallel sentence pairs as described previously, but also large amounts of annotated ranking data to train and optimize translation models for the task of Cross-Language Information Retrieval. Besides existing large-scale retrieval data sets for specialized retrieval tasks such as patent prior art search, we will describe a method to automatically extract relevance judgments from the user-generated database of Wikipedia in Chapter 4.

Chapter 3

Translation Ranking: Learning to Translate Queries

Abstract

The statistical machine translation (SMT) component of a Cross-Language Information Retrieval (CLIR) system is often modeled separately from the retrieval module. Even though we have presented first attempts to iteratively improve the translation component with new in-domain data found by the CLIR system in Section 2.4, no intrinsic evaluation of retrieval performance was performed due to lack of relevance judgments on the Twitter data. Recent work has presented results for tuning a translation system for retrieval in the standard SMT pipeline approach using a re-ranker on n-best lists (Nikoulina et al., 2012). In the following chapter, we propose a decomposable proxy for retrieval quality that obviates the need for costly intermediate retrieval. It enables us to explore the full search space of the SMT decoder by directly optimizing decoder parameters under a retrieval-based objective. By informing the SMT system of its use in CLIR through discriminative training for a retrieval objective, we optimize lexical choice for CLIR as mentioned in Chapter 1. An extensive evaluation of learned models on the task of patent prior art search however, indicates limited efficacy of the method when CLIR is carried out in a Direct Translation framework. This work has been published in Sokolov et al. (2014a) and was supported in part by DFG grant RI-2221/1-1 “Cross-language Learning-to-Rank for Patent Retrieval”.

Author's Contributions

- Development of a decomposable proxy for retrieval quality.
- Optimization of retrieval oracles through different types of word penalties.
- Implementation of the learning algorithms within the `cdec` framework.
- All experiments related to `cdec`.

3.1 Query Translation in a Retrieval Context

Cross-Lingual Information Retrieval addresses the problem of ranking documents whose language differs from the query language. One of the simplest yet well performing approaches to CLIR is based on query translation using an existing Statistical Machine Translation system which is treated as a black box. Thus, a monolingual retrieval engine does not need to be altered after translating queries into the target language. This approach is justified in the absence of cross-lingual relevance annotations (as in Chapter 2), but in the presence of large parallel text corpora for SMT training. An example to this approach to CLIR is Google (Chin et al., 2008).

However, this pipeline approach of *Direct Translation* and subsequent monolingual retrieval entails that the SMT system has no information about the retrieval task itself and thus can not be optimized for it. Although recent work has suggested "looking inside" the black box of SMT systems and establish some SMT-based confidence weights on query expansion techniques (Ture et al., 2012b), we argue in this chapter that translations for retrieval are suboptimal when the SMT system was previously optimized towards human reference translations.

To see why this might be the case, consider a CLIR model that internally represents queries or documents as bags of words and uses stopword and stemming filters. Translation decisions to match fluency and length of human reference translations may influence retrieval results only marginally. The situation is different in CLIR, as query translations may not be shown to the user directly, but only their retrieval results. Here, mostly choosing the right lexical translations for query terms, will affect the overall probability of matching relevant documents. Nevertheless, computationally expensive features for context-sensitive translation such as language models ensure coherent translation decisions of multi-word expressions and influence the lexical choice of following words (Ture et al., 2012b).

In the following sections we will describe efforts to inform the SMT model about its use in a CLIR pipeline by defining a retrieval-based objective for discriminative training of SMT model parameters. The key idea is to use another objective, such that the linear model of the SMT system places more weight on the correct lexical output, rather than concentrating on reaching or matching the length and fluency of human reference translations. This method can also be seen as a way of domain adaptation on the level of feature weights: An SMT model is adapted towards the retrieval domain by exploiting annotated relevance data.

We show that a decomposable proxy for retrieval quality in training alleviates the problem of a costly intermediate retrieval step to compute evaluation measures as in re-ranking frameworks (Nikoulina et al., 2012). It furthermore allows us to make use of the full, and lexically more diverse, decoder search space to optimize query translations for the CLIR task.

Our approach combines information specific to translation and to retrieval in one model targeted to CLIR: basic translation units, such as phrases (Koehn, 2010) or hierarchical phrase rules (Chiang, 2007), are estimated on parallel training data. In contrast, parameter optimization for lexicalized features, that can boost or demote (multi-)word translations, will be done on relevance judgments of existing queries. We present experiments in the domain of patent prior-art search where parallel training data for machine translation and relevance judgments for retrieval are available in large amounts. The results in Section 3.2 for two open-source SMT decoders, phrase-based `Moses` (Koehn et al., 2007) and hierarchical phrase-based decoder `cdec` (Dyer et al., 2010), render our approach to be a promising alternative to the standard pipeline approach.

Section 3.1.1 presents related work to optimize translation for CLIR. Section 3.1.2 formally introduces our CLIR baseline system, and Section 3.1.3 presents the structured SVM margin-rescaling framework (Tsochantaridis et al., 2005), in which we carry out training.

3.1.1 Related Work

Common techniques for modulating query expansion with lexical variations use either comparable corpus statistics (Talvensaari et al., 2007) or the k-best lists of an SMT system (Ture et al., 2012b). Experimental results show the latter approach to be superior to state-of-the-art approaches based on Direct Translation (Sokolov et al., 2014a; Schamoni et al., 2014). In Magdy and Jones (2013), consistent preprocessing of MT and IR training data yielded some improvements for retrieval and translation speed.

The work of Nikoulina et al. (2012) is closest to our approach. They present an approach to learn a re-ranking model on k-best translations that are ordered according to retrieval performance. The approach requires expensive retrieval for each derivation in the k-best list. They show improvements over a regular SMT baseline on a small set of parallel queries. However, besides the need for costly retrieval in training, the features of the re-ranking mode cannot be integrated into an SMT decoder, thus limiting the usefulness of their approach.

A tighter integration with a decoder requires the target quality to be decomposable over transductions of its search space. Such approximations were proposed and evaluated in Sokolov et al. (2014b), however, only for translation-specific measures. Similar to this work, we design a decomposable approximation for CLIR measures (MAP, NDCG) and present a learning algorithm for tuning SMT towards retrieval quality.

In our approach we will consider the optimization of the ramp loss objective. Discriminative training of SMT systems with a ramp loss objective in a k-best list setting was evaluated in Gimpel and Smith (2012).

3.1.2 Direct Translation Baseline

Let us briefly introduce notational conventions for the following *direct translation* experiments: For a translation q_e from hypothesis h of a foreign query q_f, a (monolingual) real-valued scoring function $score(q_e, d^k)$ assigns a *retrieval score* to each document d^k in collection $\mathbf{C}$. Relevance judgments for $\mathbf{C}$ are expressed by function $rl(d, q_f) \geq 0$. It assigns to each query q_f and document d^k a *relevance level*, which is zero for irrelevant documents, and increases with higher relevance. Rankings created by retrieval function $score(q_e, d^k)$ are evaluated using common rank-based metrics as introduced in Section 1.5.2. Queries and documents are represented as Bag-of-Words vectors, and scoring function $score(q_e, d^k)$ should hence be decomposable over query terms $t \in q_e$. The Okapi $BM25$ weighting scheme, as introduced in Section 1.5.1.3, fulfills this condition:

$$score(q_e, d^k) \equiv BM25(\mathbf{q}_e, \mathbf{d}^k) = \sum_{t \in q_e} bm25(t, \mathbf{d}^k). \tag{3.1}$$

3.1.3 Discriminative Training of SMT for CLIR

State-of-the-art SMT systems compute the target-language query q_e of a foreign query q_f by recombining, through concatenation and reordering, small bilingual translation units called phrases (contiguous substrings in phrase-based SMT (Koehn et al., 2003)) or synchronous grammar rules (in hierarchical phrase-based SMT (Chiang, 2007)). These units are the result of a complex process that starts with word-to-word alignments and culminates with assigning various numerical confidence scores (*feature functions* or *models*) to the extracted units (Koehn, 2010).

The union of complete hypotheses over the large number of possible input sentence splits, applicable translation options, and reordering possibilities, is called the *search space* $\mathcal{E}$. It is commonly structured as directed acyclic graphs (*lattices*) in phrase-based systems or *hypergraphs* in hierarchical phrase-based systems. Inference (*decoding*) in SMT typically relies on maximizing the hypothesis score over the search space, i.e., maximizing the likeliness of hypothesis h, given source q_f. This is usually parameterized as a linear model

$$\sigma_{smt}(h, q_e, q_f) = \mathbf{w}^T \mathbf{\Phi}(h, q_e, q_f), \tag{3.2}$$

where $\mathbf{\Phi}(h, q_e, q_f)$ is a numerical vector of features and $\mathbf{w}$ is a parameter vector:

$$q_e = \arg\max_{h \in \mathcal{E}_{q_f}} \mathbf{w}^T \mathbf{\Phi}(h, q_e, q_f) \tag{3.3}$$

$\mathcal{E}_{q_f}$ is the set of *reachable* hypotheses that the SMT system can produce for input q_f. For notational convenience we will omit dependence of the max operator and features $\mathbf{\Phi}$ on q_e and q_f.

An important computational property of the quantity under $\arg\max$ is that its components can be decomposed (through summation) over the scores of the individual units that are used in the hypothesis h for q_f. This property is required to obtain a compact representation of the decoder search space. It can then be explored efficiently with dynamic programming, for example quantities like (3.3) are computed on lattices or hypergraphs using shortest path algorithms. The optimal value for $\mathbf{w}$ is found in a tuning process that tries to replicate human reference translations by maximizing n-gram-based precision measures such as BLEU (Papineni et al., 2002) on a development set consisting of pairs of source and target sentences. A popular procedure for settings with many sparse features is the Margin Infused Relaxed Algorithm (Chiang et al., 2008).

We use the structured SVM margin-rescaling framework (Tsochantaridis et al., 2005) to learn a new $\mathbf{w}$ adapted to the CLIR task. The framework assumes a unit-decomposable penalty $\Delta(h, h') \geq 0$, defined on structured outputs (translation hypotheses), suffered for producing h instead of h'; it is zero if $h = h'$, and gracefully increases as h deviates more and more from h'. When optimizing for *translation qual-*

ity, the following loss function is minimized:

$$\mathcal{L} = \sum_{q_f} \max_{h \in \mathcal{E}_{q_f}} [\Delta(h, h^*_{q_f}) + \mathbf{w}^T \mathbf{\Phi}(h)] - \mathbf{w}^T \mathbf{\Phi}(h^*_{q_f}), \tag{3.4}$$

where $h^*_{q_f}$ is either a desired reference translation r_f, or its reachable substitute

$$h^*_{q_f} = \max_h(-\Delta(h, r_f)) \tag{3.5}$$

with Δ approximating an inverted SMT quality measure.

When optimizing for *retrieval quality*, a single desired output does not exist, but a set $\mathbf{C}^+_{q_f}$ of relevant documents for each foreign query f: $\mathbf{C}^+_{q_f} = \{d \in \mathbf{C} | rl(d, q_f) > 0\}$. Therefore we define a new function

$$\Delta(h, \mathbf{C}^+_{q_f}) = \max_h \left(S_{rel}(h, \mathbf{C}^+_{q_f}) \right) - S_{rel}(h, \mathbf{C}^+_{q_f}), \tag{3.6}$$

that is the difference in the *best* achievable approximate retrieval quality and retrieval quality for translation hypothesis h. We will defer the definition of $S_{rel}(h, \mathbf{C}^+ q_f)$ to Section 3.1.4. Let us define *fear*, *hope* and *oracle* derivations (Chiang et al., 2008; Gimpel and Smith, 2012) for a foreign query q_f:

$$\begin{aligned} h^{fear} &= \arg\max_{h \in \mathcal{E}_{q_f}} (\mathbf{w}^T \mathbf{\Phi}(h) + \Delta(h, \mathbf{C}^+_{q_f})), \qquad (3.7) \\ h^{hope} &= \arg\max_{h \in \mathcal{E}_{q_f}} (\mathbf{w}^T \mathbf{\Phi}(h) - \Delta(h, \mathbf{C}^+_{q_f})), \\ h^{oracle} &= \arg\max_{h \in \mathcal{E}_{q_f}} (-\Delta(h, \mathbf{C}^+_{q_f})), \end{aligned}$$

and the corresponding feature vectors, $\mathbf{\Phi}^{fear}_{q_f} \equiv \mathbf{\Phi}(h^{fear})$ etc. The oracle derivation is the best derivation possible, that is the one with the smallest penalty in $\mathcal{E}_f$. The fear is the derivation maximizing the model score minus a confidence margin equal to the penalty (remember that $\Delta = 0$ if $h = h^{hope}$). As the static oracle derivation can be too idiosyncratic for the linear model to produce, the hope includes the model score to find a reasonable compromise. Additionally, a hope depending on the (changing) model score increases exploration of the search space during training.

With the new penalty we consider two losses to minimize:

$$\mathcal{L}_{svm} = \sum_{q_f} (\mathbf{w}^T \mathbf{\Phi}_{q_f}^{fear} + \Delta(h^{fear}, \mathbf{C}_{q_f}^{+})) - \mathbf{w}^T \mathbf{\Phi}_{q_f}^{oracle} \tag{3.8}$$

$$\mathcal{L}_{ramp} = \sum_{q_f} (\mathbf{w}^T \mathbf{\Phi}_{q_f}^{fear} + \Delta(h^{fear}, \mathbf{C}_{q_f}^{+})) - (\mathbf{w}^T \mathbf{\Phi}_{q_f}^{hope} - \Delta(h^{hope}, \mathbf{C}_{q_f}^{+})) \tag{3.9}$$

With learning rate α, the respective (sub)gradient descent updates at step i are:

$$\mathbf{w}_{i+1} = \mathbf{w}_i - \alpha \left(\sum_{q_f} \mathbf{\Phi}_{q_f}^{fear} - \mathbf{\Phi}_{q_f}^{oracle} \right) \tag{3.10}$$

$$\mathbf{w}_{i+1} = \mathbf{w}_i - \alpha \left(\sum_{q_f} \mathbf{\Phi}_{q_f}^{fear} - \mathbf{\Phi}_{q_f}^{hope} \right). \tag{3.11}$$

The update (3.10) for the standard structured loss (3.8) (Tsochantaridis et al., 2005) increases weights of features present in the oracle derivation, and decreases weights for features in the fear. The ramp loss objective in (3.9) (Gimpel and Smith, 2012) boosts weights of features found in the current hope derivation.

3.1.4 Oracle Query Translations

We are interested in tuning an SMT system for retrieval performance. Even though some correlation between BLEU scores and MAP has been shown (Fujii et al., 2009), we argue that an n-gram based precision metric like BLEU focuses strongly on the problem of reordering translation units to accommodate for higher n-gram matches. It is thus not a suitable optimization metric for Bag-of-Words-based retrieval models, that do not take word order into account. A suitable optimization metric should either directly optimize the rank of relevant documents (*learning-to-rank*), or, more related to the task of translation, optimize lexical choices in the translation to improve term matching and adjust weights for reordering and language models correspondingly.

Directly optimizing rank-based metrics is problematic because a full retrieval for each derivation generated by the SMT system is required. This usually restricts the search space for oracle translations to the k-best list of derivations (Nikoulina et al., 2012). To alleviate this problem, we abstract away from the ranking problem and approximate the retrieval quality of a derivation h with its *relevance score* $S_{rel}(h, \mathbf{C}_{q_f}^{+})$ to the set of relevant documents $\mathbf{C}_{q_f}^{+} = \{d \in \mathbf{C} | rl(d, q_f) > 0\}$. Let $\mathbf{C}_{q_f,k}^{+} = \{d \in \mathbf{C} | rl(d, q_f) = k\}$ be the set of relevant documents in the k-th relevance level. Since

	Moses		cdec	
	MAP	**NDCG**	**MAP**	**NDCG**
junk-word penalty	0.1797	0.3702	0.1441	0.3236
word penalty	0.1756	0.3663	0.1486	0.3301

Table 3.1: Oracle performance on small training set for phrase-based (`Moses`) and hierarchical phrase-based (`cdec`) SMT decoders.

$BM25$ decomposes over query terms, we can directly assign (term-wise) $bm25$ scores to derivation terms $t \in h$ with respect to the set of relevant documents:

$$S_{rel}(h, \mathbf{C}^{+}_{q_f}) = \sum_{t \in h} S_{rel}(t, \mathbf{C}^{+}_{q_f}) = \sum_{t \in h} \sum_{k} \omega_k \frac{\sum_{d \in \mathbf{C}^{+}_{q_f,k}} bm25(t,d)}{|\mathbf{C}^{+}_{q_f,k}|}, \tag{3.12}$$

where the ω_k are *relevance weights* adjusting the importance of each relevance level k in $\mathbf{C}$. To ensure good retrieval quality of the oracle translations, we found optimal values for ω_k by grid search with a step size of 0.1 and constraint $\sum_k |\omega_k| = 1$.

So far we only reward terms that appear in $\mathbf{C}^{+}_{f}$. While the SMT system thrives to generate relevant terms, it produces them in phrases, together with connecting words as dictated by the translation model. If such ‘by-product’ terms appear sufficiently often in irrelevant documents, this can inadvertently boost their ranks. To counterbalance this effect, we experimented with two penalties, with weight $\omega_0 \leq 0$:

1. a *junk-word penalty* that fires on insertion of irrelevant terms, or

2. a *word penalty* that fires on each word in the derivation h.

A comparison of oracle configurations in terms of maximal performance over the tested range of ω_k and ω_0 found on a small training set is given in Table 3.1. Given these results, we used oracles with junk-word penalty for experiments with `Moses`, and oracles with word penalty for `cdec`.

3.2 Intrinsic Evaluation: Patent Prior Art Search

3.2.1 Data & Systems

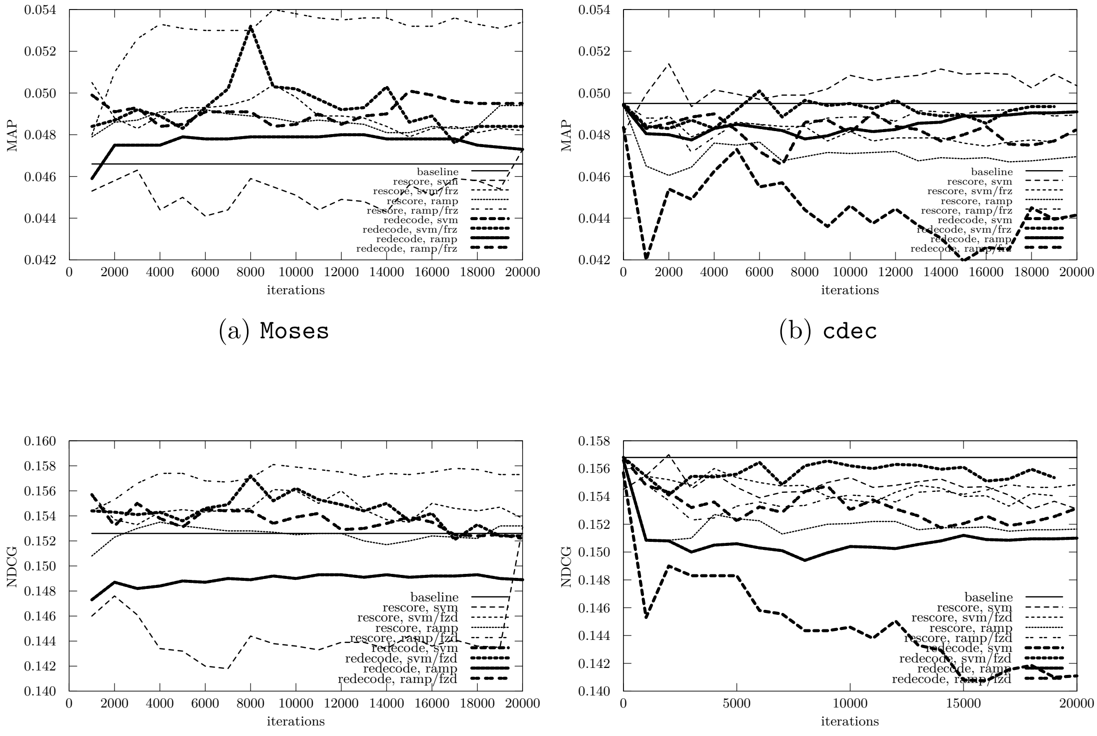
Retrieval experiments were conducted on the BoostCLIR[1] data set, a corpus consisting of Japanese (JP) & English (EN) patent abstracts (Sokolov et al., 2013). We took NTCIR-7 data (Fujii et al., 2008) (1.8M parallel sentences) from the years 1993-2000 for SMT training and the NTCIR-8 test collection (2k sentences) for parameter tuning. The data were extracted from patent descriptions published by the Japanese Patent and the US Patent & Trademark Offices as in Utiyama and Isahara (2007). A 5-gram language model on the English side of the training data was trained using the KenLM toolkit (Heafield, 2011). Additionally to a dozen of vanilla dense SMT features, both decoders included lexicalized sparse features based on word alignments, indicating source word deletions, target word insertions, and word-to-word mappings. The code for these lexicalized sparse features in `cdec` has been added to its main repository.[2] Both baseline systems were tuned with their respective MIRA (Chiang et al., 2008) implementations. On held-out parallel test data from the NTCIR, `Moses` and `cdec` achieved 0.2640 and 0.2829 BLEU (Papineni et al., 2002), respectively.

BoostCLIR contains automatically induced relevance judgments for patent abstracts. English patents are regarded as relevant to the Japanese query patent, if they are cited by either the applicant or the patent examiner, following a method by Graf and Azzopardi (2008). We assigned three relevance levels to three categories of relationships. Relevance level (2) for examiner citations, level (1) for applicants' own citations, and level (0) otherwise. As in Guo and Gomes (2009) we did not regard family patents as relevant. In the original BoostCLIR corpus, family patents are assigned a relevance level (3) and constitute almost always a literal translation to the corresponding Japanese abstract. In order to create a more realistic setting where an SMT model, optimized for IR, produces non-standard lexical output, we chose to exclude relevance level (3) judgments and documents from the data. For more details on the creation of BoostCLIR, see Sokolov et al. (2013).

A patent abstract contains about five sentences on average. At test time, we split the abstracts into single sentences, translate them using the retrieval-optimized sys-

[1] `www.cl.uni-heidelberg.de/statnlpgroup/boostclir`. last access: April 26, 2015

[2] `https://github.com/redpony/cdec/blob/master/decoder/ff_lexical.h`. last access: April 26, 2015



(a) `Moses`

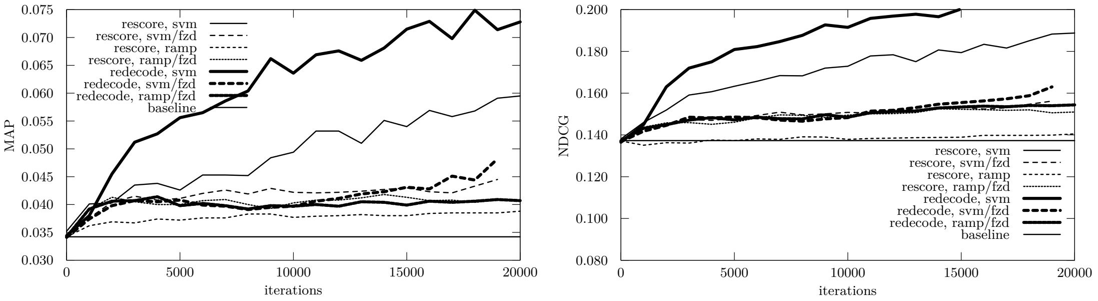
(b) `cdec`

(c) `Moses`

(d) `cdec`

Figure 3.1: MAP and NDCG scores on development data for tuning phrase-based, (a) and (c), and hierarchical, (b) and (d), SMT systems using a retrieval-based objective function. For `Moses` the "redecode, svm" curve is below the visible part of the plots.

tems, and concatenate the translations back into a single query before running monolingual $BM25$-based retrieval (Section 1.5.1.3). The data was split into two training subsets of 200 and 1,000 queries, yielding respectively, $\simeq$ 1k and $\simeq$ 5k sentences. We furthermore create development and test subsets of 400 queries each, all sampled without replacement. Oracle tuning and the training to determine the best learning configuration (see below) were done on the smaller training set and evaluated on the development set. We ran training for 20,000 iterations starting from the MIRA weights found during the SMT baseline tuning on the NTCIR-8 test collection of the respective decoders. The learning rate for all experiments was set to $\alpha = 0.001$.



(a) MAP

(b) NDCG

Figure 3.2: Evaluation on a subset of the training data for `cdec`.

3.2.2 Results & Discussion

Figure 3.1 shows MAP and NDCG scores for both decoders on the development set, evaluated every 1,000 iterations during training. We also include the retrieval results of the baseline MT systems as a straight line. An implementation of the re-ranking approach (Nikoulina et al., 2012) with our set of features scored about $\simeq 0.002$ MAP above baseline. Small but stable improvements are gained only for the phrase-based system. We plot results for both updates (3.10) and (3.11). We see that ramp loss updates generally perform better than SVM updates for `Moses`. This is due to the ability to trade off the capabilities of the model against the best possible approximate performance on the retrieval task in the ramp loss setting. The SVM update is forced to perform “bold updates” towards the oracle which can result in updates that overfit to particular oracles (Liang et al., 2006). This is supported by Figure 3.2, showing overfitting learning curves for `cdec`, most prominently for the SVM updates. Furthermore, we find it to be beneficial to constrain parameter updates during training by freezing the dense features after MIRA training on parallel data. We only tune parameters of sparse lexicalized features that promote or demote the insertion, deletion, and translation of particular words. Additionally, we test two decoding evaluation setups of search space *rescoring* and *redecoding*. The former reuses hypergraphs/lattices produced with the MIRA-tuned weights and applies new weights to find an alternative, CLIR-optimized, derivation. The latter runs the decoder directly with the new weights, which directly affects the beam search-based (`Moses`) or cube pruning (`cdec`) of search spaces. Both constraints (freezing and rescoring) show that the farther the

configuration	Moses		cdec	
	MAP	**NDCG**	**MAP**	**NDCG**
baseline	0.0438	0.1498	0.0515	0.1600
rescore	$^{0.03}$**0.0498**	$^{0.02}$**0.1575**	$^{0.11}$0.0473	$^{0.08}$0.1548
redecode	$^{0.28}$0.0463	$^{0.26}$0.1532	$^{0.23}$0.0487	$^{0.27}$0.1571

Table 3.2: Test performance of the chosen learning configurations for Moses (rescore: ramp/frz@9k, redecode: svm/frz@8k) and cdec (rescore: svm/frz@2k, redecode: svm/frz@6k). Superscripts denote p-values obtained by a paired randomization test with respect to the baseline.

configuration	BLEU NTCIR-8	Oracle MAP	Oracle NDCG
poplimit=200 (default)	0.2879	0.1486	0.3301
poplimit=400	0.2897	0.1638	0.3680
poplimit=800	0.2917	0.2087	0.4152
unpruned, no LM	0.2237	0.5060	0.7372

Table 3.3: Oracle performance on small training set for cdec using larger search spaces. "unpruned, no LM" refers to a system without a language model.

setup strays away from the original MIRA model, the more difficult becomes generalization to unseen data. This suggests that it is crucial to find the optimal combination of translation- and retrieval-specific information for both inference and learning.

Table 3.2 shows test results for models trained on the bigger training set using the best settings found on the development set (see caption). For the hierarchical system, improving over the significantly (at level $p = 0.01$) stronger baseline proves to be difficult. One reason could be a relatively harsh pruning strategy in cdec, governed by the language model, which produces lexically less diverse search spaces. In fact, the MIRA baseline for cdec assigns the largest weight to the language model. This explanation is supported by weaker oracles (Table 3.1) and fewer active sparse features in the learned models when compared to Moses (17,000 vs. 23,000 on the small training set).

The gap between oracle performance of Moses and cdec, suggests that the hierarchical system is unable to produce derivations that match relevant documents and are radically different from fluent translations. We thus experimented with larger cube pruning poplimits than default 200 to obtain better oracles. The cube pruning poplimit controls the number of partial derivations kept at every node created in the hypergraph, while applying the language model rescoring (see Chapter 5 for more information on hierarchical MT decoding). Table 3.3 shows translation scores on the development set for `cdec` larger poplimit settings. We also show retrieval results for oracles produced using these `cdec` configurations. Oracle configurations were found as described in Section 3.1.4. While BLEU scores only increase moderately, retrieval efficacy of IR oracles improves significantly. Especially in unpruned search spaces, oracles found with respect to relevant documents score dramatically well. However, with larger poplimits, the size of search spaces also increased significantly, which rendered decoding and training time, as well as memory requirements, too difficult to handle. Engineering an optimized training algorithm for these cases is subject to future work.

3.3 Conclusion and Outlook

We presented an approach for tuning an SMT system for cross-lingual retrieval. Our approach is efficient, since it uses a decomposable proxy for retrieval quality that can be computed directly on the translation hypergraph or lattice in training, avoiding costly intermediate retrieval steps in similar approaches (Nikoulina et al., 2012). Furthermore, it is effective since optimal weights of retrieval-governing sparse features are accessible to the decoder, which combines this information with translation-specific dense features for improved query translation in a cross-lingual retrieval setup.

Experiments on search space sizes and oracle performance illustrate the lexical diversity present in the search space of current SMT models. If the correct choice of a lexical translation would be known in advance (as is the case in oracles), *reaching* the best matching translation derivation with respect to the relevant documents is very likely. Nevertheless, the presented approach of optimizing translation output towards derivations yielded only modest improvements for only one of two evaluated MT frameworks during test time. We conjecture, based on training and development set performances, that the ability to generalize well to unseen data, at least for manageable training data sizes, is severely limited by the CLIR pipeline approach used in the experiments: the disambiguation to a single first-best translation string, which is subsequently used in a monolingual $BM25$-based retrieval setup, is not sufficient to exploit the fact that the translation model has enough expressive freedom to generate derivations that match the relevant documents very well. For example, promoting certain lexical translations for training query A may not be, or even be counterproductive for query B at test time.

In Chapter 5, we will encounter these shortcomings by integrating the retrieval function into the SMT decoder. By modeling a retrieval function within the SMT decoding process using efficient decomposable retrieval features, we are able to present all translation alternatives in the search space to the ranking function. We show that such a model significantly outperforms a CLIR pipeline approach of direct translation, as well as a Probabilistic Structured Query approach. A combined model of translation and retrieval furthermore enables us to use learning-to-rank techniques to directly optimize ranking in CLIR. For effective training and thorough evaluation of the model presented in Chapter 5, large-scale relevance annotated data is crucial. We thus present a method for automatic extraction of relevance judgments from Wikipedia in the following chapter.

Chapter 4

Automatic Extraction of Relevance Annotations from Wikipedia

Abstract

Training of ranking models, be them translation-based or built from structural knowledge about documents, requires large-scale annotated training data. Annotations in (Cross-Language) Information Retrieval are usually given by relevance judgments. These judgments, typically created by human annotation, are costly to produce. Hence, an automatic extraction of relevance judgments, derived from reasonable assumptions about the link structure in large document collections such as Wikipedia, can provide a valuable alternative for enabling cost-efficient training of ranking models. The following chapter presents a method to automatically extract relevance judgments from multilingual Wikipedia databases by defining a notion of relevance, derived from the rich cross-lingual link structure present in the online encyclopedia. The resulting data set provides a large-scale alternative to existing data sets for specialized retrieval tasks such as patent prior art search. It has been recently used in the work of Schamoni et al. (2014), the most recent contribution in the line of work regarding large-scale training of ranking models (Bai et al., 2010; Guo and Gomes, 2009; Sokolov et al., 2013). Furthermore, Chapter 5 of this thesis will carry out discriminative training of a joint model of context-sensitive translation and retrieval on this data set using the pairwise learning-to-rank framework. The data set is publicly available under a Creative Commons BY-SA 4.0 Unported License.[1]

Author's Contributions

- Definition of the CLIR use case of cross-lingual Wikipedia article retrieval.
- Deriving a notion of relevance based on the rich link structure of Wikipedia.
- Data analysis and automatic extraction of relevance judgments to obtain a data set release, WikiCLIR.

[1] `http://www.cl.uni-heidelberg.de/wikiclir/`. last access: April 26, 2015

4.1 Large-Scale Training Data for CLIR

Learning-to-rank approaches (Chapter 5) or training of sparse, lexicalized ranking models (Bai et al., 2010; Guo and Gomes, 2009; Sokolov et al., 2013) require large amounts of annotated training data, which are, in the case of Cross-Language Information Retrieval (CLIR), not readily available. Common retrieval data sets for the Text REtrieval Conference (TREC)[2] or the LETOR data set (Liu et al., 2007) are either monolingual or provide only small amounts of test queries, which renders large-scale machine learning impossible. Furthermore, those data sets are often preprocessed into feature matrices and do not provide raw texts of documents and queries. For the translation-based retrieval models in this thesis, however, such textual data is inherently necessary. While click-through data from large web search engines would constitute a viable option, such data is rarely made publicly available by search engine providers. A possible solution to the lack of training data is the exploitation of existing structural information in special domains that often already encode relevance information and allow the automatic extraction of relevance judgments, based on reasonable assumptions about relevance.

For the task of patent prior art search, the patent citation graph defined by citations within patent applications has been shown to be a suitable resource for automatic extraction of relevance judgments (Graf and Azzopardi, 2008; Guo and Gomes, 2009). Sokolov et al. (2013) present BoostCLIR[3], a corpus of Japanese-English patent abstracts that applies a method proposed by Graf and Azzopardi (2008) to extract relevance judgments from the citation graph. BoostCLIR is also used in Chapter 3 to optimize query translations from a Statistical Machine Translation model for the task of Cross-Language Information Retrieval.

In this chapter, we describe a method to exploit the rich (cross-lingual) link structure in the online encyclopedia Wikipedia. While previous work has used this resource for retrieval before (Bai et al., 2010), our data set differs in the definition of the ranking task, and the notion of relevance assumed. Furthermore, the extracted queries constitute fluent natural language queries, that allow the evaluation of context-sensitive translation-based CLIR models.

[2] `http://trec.nist.gov/`. last access: April 26, 2015

[3] `http://www.cl.uni-heidelberg.de/boostclir/`. last access: April 26, 2015

4.2 Cross-lingual Encyclopedic Article Retrieval

Consider a cross-lingual retrieval scenario in which a user intends to write an article for Wikipedia in language A, while at the same time relevant articles in the Wikipedia of language B may already exist. Authors on Wikipedia naturally want to avoid orphan articles and are encouraged to cite their sources. Based on this assumption, we define relevance between Wikipedia articles as follows:

Definition 2 *A Wikipedia article a_i is relevant to Wikipedia article a_j, if there exists a lexical semantic relation $S(a_i, a_j)$ between them, e.g synonymy, antonymy, hypernymy, hyponymy, or metonymy.*

Encyclopedic articles generally try to exhaustively refer to relevant sub- or super concepts, or use instances of the described concept as an example. Besides references within the text, hyperlinks between Wikipedia articles may indicate such semantic relations implicitly. Furthermore, cross-lingual synonymy is encoded with inter-language links that link the same concept in two languages. In other words, following Bai et al. (2010), the set of relevant articles can be derived from the Wikipedia link structure between existing articles. The set of links within an article defines the set of relevant concepts the author had in mind while writing the article.

Consequently, we extract cross-lingual relevance judgments for synonymic relations between source language queries and target language documents (*cross-lingual mates*) via the graph of inter-language links and assign a relevance level of (3). Relevance judgments that encode other semantic relations, such as hyper- and hyponymy, are extracted via the set of intra-Wikipedia links present in the cross-lingual mate and assigned a relevance level of (2). Instead of using all outgoing links from the mate (Bai et al., 2010), we enforce a stricter relevance constraint by taking only *bi-directional links* into account, i.e. articles that link to each other. The fact that the authors of both articles independently encoded the same relation with their link decision, provides a stronger signal for relevance. In terms of data size, this additional constraint significantly reduces the number of relevant documents (Table 4.3). Table 4.1 shows examples of bi-directional links in the English Wikipedia. The Wikipedia articles on *climbing* and *rock climbing* both link to each other, thus establishing a semantic relation of hyponymy, and hypernymy respectively. We describe automatic extraction and data pre-processing steps for a German-English data set in the following, but the proposed method is not limited to a specific language pair.

article title	bi-directional links
Computer expo	SMAU, COMDEX, CeBIT, Trade fair, Computex Taipei, LinuxTag
Climbing	SportAccord, Rock climbing, Tree climbing, Climbing protection , [...]
Formal language	Theorem, Context-free language, Axiom, [...]
Applied linguistics	Theoretical linguistics, Translation, Lexicography, Linguistics, [...]

Table 4.1: Examples of bi-directional links in the English Wikipedia (11/4/2013).

Consider the case where the German Wikipedia article on *geological sea stacks* does not yet exist. A native speaker of German with profound knowledge in geology intends to write it, naming it "*Brandungspfeiler*", while seeking to align its structure with the English counterpart. The task of a cross-lingual retrieval engine is to return a list of relevant English Wikipedia articles (Definition 2) that may describe the very same concept (*Stack (geology)*), or related concepts, e.g. particular instances of it (*Bako National Park*, *Lange Anna*). The information need may be paraphrased as a high-level definition of the topic. Since typically the first sentence of any Wikipedia article is such a well-formed definition, this allows us to extract a large set of sentence-long natural language queries from Wikipedia articles, such as:

> *Brandungspfeiler sind vor einer Kliffküste aufragende Felsentürme und vergleichbare Formationen, die durch Brandungserosion gebildet werden.*[4]

4.3 Dataset Creation

The data set described here is made publicly available[5] under a Creative Commons BY-NC-SA 3.0 Unported License[6]. It is built from raw XML dumps of the German and English Wikipedia dated November 22nd[7] and 4th[8] 2013, respectively. We selected German as the query language, because the English Wikipedia contains three times more articles and hence provides a much larger and diverse document collection.

[4] `http://de.wikipedia.org/wiki/Brandungspfeiler`. last access: April 26, 2015

[5] `http://www.cl.uni-heidelberg.de/wikiclir/`. last access: April 26, 2015

[6] `http://creativecommons.org/licenses/by-nc-sa/3.0/`. last access: April 26, 2015

[7] `http://dumps.wikimedia.org/dewiki/20131122/dewiki-20131122-pages-articles.xml.bz2`. last access: April 26, 2015

[8] `http://dumps.wikimedia.org/enwiki/20131104/enwiki-20131104-pages-articles.xml.bz2`. last access: April 26, 2015

	#articles	#disamb	#empty	#other	#redirect	#stub	$\sum$
DE	1,476,474	191,825	137	475,058	1,111,557	4	3,255,051
EN	4,442,789	128,446	674	3,159,844	6,230,594	1,793,094	13,962,347

Table 4.2: Page type counts in Wikipedia XML dumps as classified by Cloud9: proper articles, disambiguation pages, empty pages, other/wikipedia metapages, redirection pages, and incomplete stub articles.

Page Types. XML parsing of over 17M pages, Wikipedia markup removal, and link extraction from article texts was carried out using the Cloud9 toolkit[9], which integrates the Bliki Parsing Engine[10]. Table 4.2 shows the distribution of page types in the German and English Wikipedia as classified by the Cloud9 toolkit. The toolkit classifies page types based on its markup structure and the page type name given in the respective language. Pages classified as *Articles* are Wikipedia articles with full encyclopedic coverage. *Disambiguation* pages list concepts with the same (or similar) surface form, *other* refers to Wikipedia meta-pages such as help pages. *Redirection* pages point to an article via an alternative name, and articles are marked as *stub* if they are too short to contain sufficient encyclopedic coverage of the topic[11]. As the English Wikipedia contains over 1.7M stub articles and those are densely connected to other English and German articles via inter-language links, we decided to include them into the set of proper articles. Thus, the set of content pages we retained for document and query candidates consisted of *article* and *stub* pages. *Disambiguation* pages were disregarded, since cross-language links already disambiguate correctly.

Link Resolution. Wikipedia pages often contain hundreds of outgoing links to other pages, especially if a page only lists instances of a certain concept (e.g. the "list of X"-articles). In the German Wikipedia, an article exhibits 48.76 outgoing links on average (maximum at 10,500), and 26.93 in the English Wikipedia (maximum at 9,400), respectively. However, not all links point to other articles, but rather to Wikipedia meta-pages or topics not included in the XML dump. Thus all intra-language links are resolved by checking for the existence of the target page in the collection of *article*, *redirection* and *stub* pages. Redirection pages were further used to resolve links via at most two hops: If an article is connected to another article via a redirection page, the

[9] `lintool.github.io/Cloud9/index.html`. last access: April 26, 2015

[10] `https://code.google.com/p/gwtwiki/`. last access: April 26, 2015

[11] `http://en.wikipedia.org/wiki/Wikipedia:Stub`. last access: April 26, 2015

	DE	EN
# links resolved (direct)	43,378,312	124,056,598
# links resolved (redirect)	5,073,256	29,248,611
# links unresolved	4,998,169	10,830,359
avg. # inlinks/article	26.47	29.86
avg. # outlinks/article	27.15	31.23
avg. # bi-directional links/article	5.29	5.21
avg. # links to or from same category	6.11	5.06

Table 4.3: Statistics on (intra-)Wikipedia links after resolving link targets.

link target in the source page is replaced with the target of the redirection page. We only keep links to pages that are present in the set of articles and stubs, which results in the statistics given in Table 4.3. The English and German Wikipedia are very similar with respect to the average number of incoming, outgoing, and bi-directional links per article. The constraint of only using bi-directional links reduces the number of possible relevant articles to an amount justifiable to be read by a Wikipedia author in the scenario described above. Since February 2013, the Wikimedia Foundation started to remove inter-language links from the Wikipedia page markup, and now maintains *Wikidata*[12], a database where structural information about pages across languages is stored. Inter-language links from Wikidata were matched with page IDs from the XML dumps to resolve inter-language links. After filtering German pages without an English mate, the repository of articles and stubs was reduced to 755,400 German articles constituting the set of query candidates.

Query and Document Extraction. The first sentences of German Wikipedia articles were used as queries, since they generally convey a high-level description of an article's concept and can be viewed as a paraphrase of the user's information need. Sentence extraction was carried out with the NLTK toolkit[13]. The first sentence of any German article was classified as a query set if (1) the article did not describe a calendar day, month or year, and (2) was not a "list of X"-article, e.g. *Liste von Autoren.* Furthermore, the sentence itself was required to contain no asterisk (3), which usually indicates biographic descriptions of persons, and (4) the length of the extracted sentence was between 8 and 80 words. This yielded a final set of 245,294 German queries.

[12] `www.wikidata.org/`. last access: April 26, 2015

[13] `www.nltk.org/`. last access: April 26, 2015

WikiCLIR	# queries	# documents	# $\frac{\text{documents}}{\text{query}}$	# $\frac{\text{words}}{\text{query}}$
train	225,294	1,226,741	13.04	25.80
dev	10,000	113,553	12.97	25.75
test	10,000	115,131	13.22	25.73

Table 4.4: Statistics of WikiCLIR (German-English) data splits.

In a final preprocessing step, occurrences of the article's title words were removed from the German query sentence to avoid rendering the retrieval task too easy for CLIR models with a "self-translation" component (see Sections 2.2.1, 2.4.1, and 5.2.4). Due to the large number of articles for named entities, we experienced a ceiling effect on retrieval performance if literal keyword matching without translation was possible. We thus remove the name of the article from the final query string. For the previous example, this yields:

> *sind vor einer Kliffküste aufragende Felsentürme und vergleichbare Formationen, die durch Brandungserosion gebildet werden.*

The final query strings contain about 26 words on average (Table 4.4). Due to the large variance of article lengths, English documents were stripped to the first 200 words, not considering stopwords. This reduces the size of feature spaces in sparse ranking models and was found to be crucial to enable efficient training (Schamoni et al., 2014).

Data Splits. The final data set, WikiCLIR, is generated by collecting all English documents that are judged relevant to any of the 245,294 German queries. Sets for training, development, and testing were created by sampling and splitting on the query level. The document collections for each set contain only relevant documents with respect to the queries in the set. Statistics are given in Table 4.4.

4.4 Baseline Results for SMT-based CLIR Models

WikiCLIR is a data set suitable for large-scale training of retrieval models for CLIR. It has been used in Schamoni et al. (2014) showing that the combination of different types of retrieval models can contribute orthogonal information to a CLIR system that significantly outperforms single models. In this section, we only report our baseline results for SMT-based models that do not require any learning. These results establish a baseline for the SMT-based CLIR models trained on relevance judgments in Chapter 5. We use two types of CLIR models that are based on a hierarchical phrase-based

	dev			test		
WikiCLIR	MAP	NDCG	PRES	MAP	NDCG	PRES
DT[1]	.3632	.5656	.7178	.3678	.5691	^{2}.7219
PSQ[2]	.3588	.5633	.7125	.3642	.5671	.7165
DT+PSQ				^{1,2}.3742	^{1,2}.5777	^{1,2}.7306

Table 4.5: Results for SMT-based CLIR models on WikiCLIR. Significant differences (Smucker et al., 2007) at $p = 0.01$ are indicated with superscripts.

SMT system: (1) a *Direct Translation* (DT) approach that translates the query and performs monolingual BM25-based retrieval, and (2) a *Probabilistic Structured Query* (PSQ) approach as presented in Section 2.4.1. The advantage of PSQ over DT is the ability to carry over weighted translation alternatives to the retrieval process to increase the probability of matching a document term.

Table 4.5 shows MAP, NDCG and PRES scores for both types of models on the development and test data of WikiCLIR. The SMT system is a German-English hierarchical phrase-based system using `cdec`. The parallel training data (over 104M words) consisted of the Europarl[14] corpus in version 7, the News Commentary corpus, and the Common Crawl corpus (Smith et al., 2013). Word alignments were created with `fast_align` (Dyer et al., 2013). The 4-gram language model was trained with the KenLM toolkit (Heafield, 2011) on the English side of the training data and the English Wikipedia documents from WikiCLIR. Feature weights were optimized using MIRA (Chiang et al., 2008) on the WMT2011 news test set (3003 sentences). The parameters for the PSQ model were found on the WikiCLIR development set: size of n-best lists: 1000; interpolation parameter λ=0.4, lower threshold L=0, and cumulative threshold C=1.

We can observe on both development and test sets that DT and PSQ models score very similar, with only PRES on the test set being significantly better for DT. A model combination learned on the development set (Schamoni et al., 2014), significantly improves performance on test, which suggests that both models produce sufficiently distinct rankings despite using the same underlying SMT system. Results for the combination of both models were produced by the first author. Numbers for the development set were not available anymore.

[14] `http://www.statmt.org/europarl/`. last access: April 26, 2015

Chapter 5

Document Ranking: Bag-of-Words Forced Decoding for CLIR

Abstract

Current approaches to Cross-Language Information Retrieval (CLIR) rely on standard retrieval models into which query translations by Statistical Machine Translation (SMT) are integrated at varying degree. While Direct Translation (DT) approaches resort to standard monolingual retrieval using the first-best query translation string, existing work such as Ture et al. (2012a) use multiple decoder derivations to produce probabilistically weighted translation options. However, the SMT system producing such alternatives is not optimized for retrieval. In this chapter, we present an attempt to turn this situation on its head: Instead of the retrieval aspect, we emphasize the translation component in CLIR and how it can be used for retrieval itself. We perform search by using an SMT decoder in forced decoding mode to produce a Bag-of-Words representation of the target documents to be ranked. The SMT model is extended by retrieval-specific features that are optimized jointly with standard translation features for a ranking objective. We find significant gains over the state-of-the-art in a large-scale evaluation on cross-lingual search in the domains of patents and Wikipedia.

This work was published in the proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technologies (Hieber and Riezler, 2015).

Author's Contributions

- A new model of joint translation and retrieval that uses an IR feature enriched decoder score for ranking documents.
- Description of an extended dynamic programming procedure to efficiently decode with respect to single documents.
- Defining retrieval features that decompose over partial hypothesis and generalization to unseen data through a default weight approach.
- Large-scale training of forced decoding model on relevance judgments using established learning-to-rank techniques.
- A C++ implementation is available at `https://github.com/fhieber/cclir`.

5.1 Introduction & Related Work

Approaches to CLIR have been plentiful and diverse. While simple word translation probabilities are easily integrated into term-based retrieval models (Berger and Lafferty, 1999; Xu et al., 2001), state-of-the-art SMT systems (Koehn, 2010; Chiang, 2007) are complex statistical models on their own. The use of established translation models for context-aware translation of query strings, effectively reducing the problem of CLIR to a pipeline of translation and monolingual retrieval, has been shown to work well in the past (Chin et al., 2008). Only recently, approaches have been presented to include (weighted) translation alternatives into the query structure to allow a more generalized term matching (Ture et al., 2012a,b). However, this integration of SMT remains agnostic about its use for CLIR and is instead optimized to match fluent, human reference translations. In contrast, retrieval systems often use Bag-of-Words representations, stopword filtering, and stemming techniques during document scoring, and queries are rarely fluent, grammatical natural language queries (Downey et al., 2008). Thus, most of a translation's structural information is lost during retrieval, and lexical choices may not be optimal for the retrieval task. Furthermore, the nature of modeling translation and retrieval separately requires that a single query translation is selected, which is usually done by choosing the most probable SMT output.

Attempts to inform the SMT system about its use for retrieval by optimizing its parameters towards a retrieval objective have been presented in the form of re-ranking (Nikoulina et al., 2012) or ranking (Chapter 3). In this chapter, we take this idea a step further and directly integrate the task of *ranking documents* with respect to the query into the process of *translation decoding*. We make the full expressiveness of the translation search space available to the retrieval model, without enumerating all possible translation alternatives. This is done by augmenting the linear model of the SMT system with features that relate partial translation hypotheses with documents in the retrieval collection (Section 5.2.1). These retrieval-specific features decompose over partial translation hypotheses and thus allow efficient decoding using standard dynamic programming techniques (Section 5.2.2). Decoding is forced to produce a Bag-of-Words representation of each target document to be ranked. Furthermore, we apply learning-to-rank to jointly optimize translation and retrieval for the objective of retrieving relevant documents (Section 5.3).

One of the key features of our approach is the use of context-sensitive information such as the language model and reordering information. We show that the use of such a translation-benign search space is crucial to outperform state-of-the-art CLIR approaches. Our experimental evaluation of retrieval performance is done on Wikipedia cross-lingual article retrieval, as described in Chapter 4, and patent prior art search (Fujii et al., 2009; Guo and Gomes, 2009; Sokolov et al., 2013; Schamoni et al., 2014), as used in Chapter 3. On both data sets, we show substantial improvements over the CLIR baselines of Direct Translation and Probabilistic Structured Queries, with and without further parameter tuning using learning-to-rank techniques. From the results we conclude, that, in spite of algorithmic complexity, it is central to model translation and retrieval jointly to create more powerful CLIR models.

Related Work. The framework of translation-model based retrieval has been introduced by Berger and Lafferty (1999). An extension to the cross-lingual case using context-free lexical translation tables has been given by Xu et al. (2001) (Chapter 2). While the industry standard to CLIR is a pipeline of query translation using SMT and monolingual retrieval (Chin et al., 2008), recent approaches include (weighted) SMT translation alternatives into the query structure to allow a more generalized term matching (Ture et al., 2012a,b) (Section 2.4.1). Less work has been devoted to optimizing SMT towards a retrieval objective, for example in a re-ranking framework (Nikoulina et al., 2012), or by integrating a decomposable proxy for retrieval quality of query translations into discriminative ranking (Chapter 3).

Most similar to our approach is the recent work of Dong et al. (2014) who use the `Moses` translation option lattices for translation retrieval, i.e mining comparable data. Their query lattices given by the translation options encode exponentially many queries and are used to retrieve the most probable translation candidate from a set of candidates. The approach is evaluated in the context of a parallel corpus mining system. We present a model that not only uses the full search space, including the language model and reordering information, but also evaluate the model specifically for the task of retrieval, rather than mate-finding only. We show that a forced decoding model using Bag-of-Words representations for documents and retrieval features that are decomposable over query terms significantly outperforms state-of-the-art CLIR baselines such as Direct Translation (Chin et al., 2008) or Probabilistic Structured Queries obtained from n-best list query translations (Darwish and Oard, 2003; Ture et al., 2012b). Additionally, we find that the use of context-sensitive translation information such as language models or reordering information, greatly improves retrieval

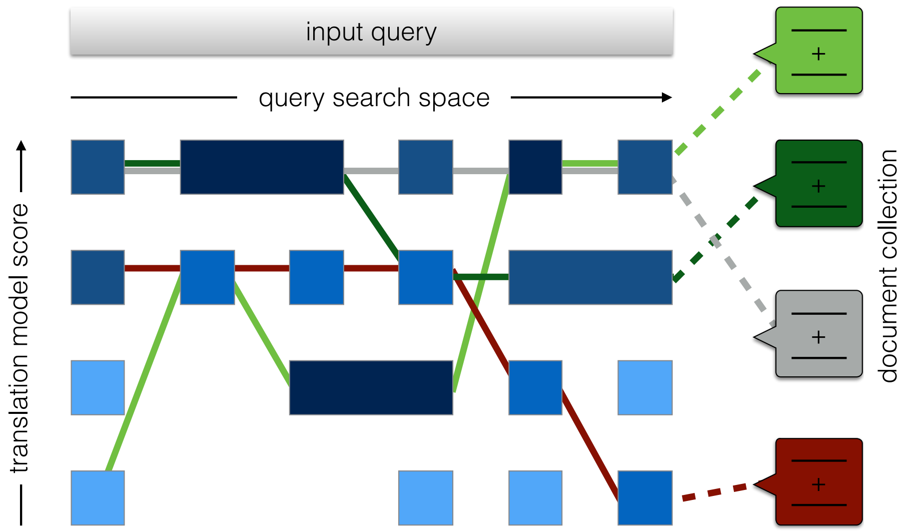


Figure 5.1: Illustration of a Bag-of-Words forced decoding model for retrieval where different documents yield different derivations, thus producing a ranking over documents directly.

quality in these types of models. We furthermore show how to directly optimize the retrieval objective using large-scale retrieval data sets with automatically induced relevance judgments.

5.2 A Bag-of-Words Forced Decoding Model

The CLIR model described in the following can be seen as an instance of *forced decoding* for statistical machine translation systems: In forced decoding, the task of the SMT decoder is to generate a set of derivations that are compatible to a known reference translation and to return their model scores. Derivations that are *incompatible* with the reference, that is, they produce different surface strings, receive score 0. If an input sentence can not generate any derivation that matches the reference translation, the reference is called *non-reachable*.

In order to jointly model translation and retrieval for the task of Cross-Language Information Retrieval, we draw from the theoretical framework of (monolingual) language-model based retrieval models as introduced in Section 1.5.1.4. A user query is seen as a distorted version of the relevant documents. The common way of using such query likelihood models (Berger and Lafferty, 1999) in CLIR is to use lexical translation tables and model foreign documents in uni-gram language models (Chapter 2).

Here, we seek to use a full SMT system for query translation and directly obtain a retrieval score during the decoding process. We thus force the decoder to obtain a query derivation that is (lexically) as close as possible to the respective document. Within the range of possible translation alternatives from the SMT model, the decoder finds the best possible translation alternative for every document (Figure 5.1). In contrast to regular forced decoding, we relax the harsh constraint of exact reachability, by disregarding word order information in the documents. We use a weighted linear combination of retrieval features computed on the Bag-of-Words representations of documents that decompose over derivation terms and reward matches between query translation and document. The use of two types of feature spaces, namely translation features and retrieval features, allows the model to balance between the task of producing an accurate translation, and the task of query term matching.

Figure 5.1 illustrates the model graphically. The query search space for the input query is given by the translation model, which would, in absence of any retrieval features, produce the most likely query translation under the translation model features (top-most blue combination of phrasal segments). Our joint model of translation and retrieval, however, may produce different derivations (and scores) for each of the documents in the collection, depending on their lexical content. For example, the red document produces a derivation that strongly deviates from the most likely translation derivation (dark red path). Even though this derivation matches words in the document, it receives a penalty from the translation model for not being an adequate translation. The green documents in contrast, produce derivations that agree in terms of lexical matches and translation adequacy, thus producing a higher model score overall.

This architecture allows the induction of a ranking over documents with respect to the SMT decoder score. Query expansion is naturally integrated by allowing the model to produce an optimal lexical choice for each document candidate. At the same time, the joint definition of retrieval and translation allows training such a model with respect to relevance judgments on retrieval data, and thus optimizing translation for retrieval directly.

5.2.1 Model Definition

SMT systems use a Viterbi approximation to find the output hypothesis q_e^*

$$q_e^* = \arg\max_{q_e} \max_{h \in \mathcal{E}_{q_f}} P(h, q_e | q_f). \tag{5.1}$$

over the search space of hypotheses or derivations $h \in \mathcal{E}_{q_f}$ for a given input q_f. The probability of a translation output q_e under derivation h given q_f is usually modeled in a log-linear model

$$P(h, q_e | q_f; \mathbf{w}_{smt}) = \frac{e^{\sigma_{smt}(h, q_e, q_f)}}{\sum_{q_e, h} e^{\sigma_{smt}(h, q_e, q_f)}},$$

where $\sigma(h, q_e, q_f)$ is a learned linear combination of input-output features, that is, the dot product between parameter column vector $\mathbf{w}_{smt}$ and feature column vector given by feature map $\mathbf{\Phi}_{smt}$,

$$\sigma_{smt}(h, q_e, q_f) = \mathbf{w}_{smt}^T \mathbf{\Phi}_{smt}(h, q_e, q_f). \tag{5.2}$$

Bag-of-Words Forced Decoding. In CLIR, we seek to choose a derivation that is *both* an accurate translation of the input according to the translation model, and a well-formed discriminative query that matches relevant documents with high probability. We combine both objectives by directly modeling the probability of a document d_e in target language e given a query q_f in source language f, factorized as follows:

$$P(d_e | q_f) = \sum_{h \in \mathcal{E}_{q_f}} \underbrace{P(h | q_f)}_{\text{translation}} \times \underbrace{P(d_e | h, q_f)}_{\text{retrieval}}.$$

Applying the same Viterbi approximation during inference as in (5.1), we choose the retrieval score of d_e to be the score of the highest scoring hypothesis h,

$$score(q_f, d_e) = \max_{h \in \mathcal{E}_{q_f}} P(h | q_f) \times P(d_e | h, q_f), \tag{5.3}$$

where the product between both models can be interpreted as a conjunctive operation similar to a product of experts (Hinton, 2002): A high score is achieved if both experts, namely translation and retrieval models, assign high scores to a hypothesis. That is, the model attempts to produce a well-formed translation, but at the same time chooses lexical items present in the Bag-of-Words representation of the document. Similarly, we can interpret the inclusion of the retrieval component as a constraint to *force* the decoder to retrieve d_e with high probability. We will henceforth call our approach Bag-of-Words Forced Decoding (BOW-FD).

The translation term $P(h|q_f)$ is modeled as in (5.2) for standard hierarchical phrase-based SMT (Chiang, 2007) and left unchanged in our joint model. The retrieval term $P(d_e|h, q_f)$ is modeled in a similar form

$$\sigma_{ir}(h, d_e) = \mathbf{w}_{ir}^T \mathbf{\Phi}_{ir}(h, d_e),$$

where IR features do not depend on q_f and decompose over derivation terms. This allows a Bag-of-Words vector representation of documents, and retrieval features are local to single edges in the search space for efficient Viterbi inference. The joint scoring model is defined as follows:

$$score(q_f, d_e; \mathbf{w}) = \max_{h \in \mathcal{E}_{q_f}} exp(\sigma_{smt}(h, q_e, q_f) + \sigma_{ir}(h, d_e)),$$

where the weight vector is defined by the vector concatenation $\mathbf{w} = \mathbf{w}_{smt} \| \mathbf{w}_{ir}$, and q_e refers to the yield of derivation h.

5.2.2 Dynamic Programming on Hypergraphs

Decoding in a hierarchical phrase-based SMT (Chiang, 2007) is usually understood as a two-step process: Initially, an input sentence is parsed using a Weighted Synchronous Context-Free Grammar (WSCFG) in a bottom-up manner to construct an initial hypergraph $\mathcal{H}$ that compactly encodes the full search space ("translation forest") (Gallo et al., 1993; Klein and Manning, 2001; Huang and Chiang, 2005; Dyer et al., 2010).

Definition 3 *An ordered, directed hypergraph $\mathcal{H}$ is a tuple $\langle V, E, g, \mathcal{W} \rangle$, consisting of a finite set of nodes V, the goal node g, a finite set of hyperedges E, and weight function $\mathcal{W} : E \mapsto \mathbb{R}$ assigning real-valued weights to $e \in E$. A hyperedge $e \in E$ is a tuple $e = \langle h(e), t(e) \rangle$, where $h(e)$ denotes the head node and $t(e)$ the vector of tail nodes. We further define $|t(e)|$ as the arity of e and $\text{IN}(v) = \{e \in E | h(e) = v\}$ as the set of incoming edges for node $v \in V$.*

Algorithm 2 Inside algorithm over hypergraph with real valued weights (Dyer, 2010).

```
Require: search space $\mathcal{H}$, semiring $K$, weight function $\mathcal{W}$, beam width $b$
procedure Inside($\mathcal{H}$, $\mathcal{W}$, $b$)
    for all nodes $v$ in topological order in $\mathcal{H}$ do
        if $\text{IN}(v) = \emptyset$ then
            $S[v] \leftarrow \bar{1}$
        else
            $S[v] \leftarrow \bar{0}$
            for all edges $e_i : \text{IN}(v)$ do
                if $i < b$ then
                    break
                    $s \leftarrow \mathcal{W}(e_i)$
                    for all nodes $u_j : t(e_i)$ do
                        $s \leftarrow s \bigotimes S[u_j]$
                    end for
                    $S[v] \leftarrow S[v] \bigoplus s$
                end if
            end for
        end if
    end for
    return $S[g]$
end procedure
```

Language models are typically added in a second rescoring phase that is carried out by approximate solutions, such as cube pruning (Chiang, 2007; Huang and Chiang, 2007), limiting the number of derivations created at each node through the poplimit parameter. A translation hypothesis $h \in \mathcal{E}$ corresponds to a sequence of nodes $S \subseteq V$ connected via hyperedges e ending in goal node g. Each hyperedge e is associated with a synchronous translation rule, $r(e)$, and corresponding feature values $\mathbf{\Phi}(r(e))$. The weight of hyperedge e is defined as $\mathcal{W}(e; \mathbf{w}) = \mathbf{w}^T \mathbf{\Phi}(r(e))$.

The quantity in (5.1) is efficiently computed using dynamic programming under the proper semiring. A commutative semiring K is a tuple $\langle \mathbb{K}, \bigoplus, \bigotimes, \bar{0}, \bar{1} \rangle$, of a set $\mathbb{K}$, an associative and commutative addition operator $\bigoplus$, an associative multiplication operator $\bigotimes$, and their "neutral" elements $\bar{0}$ and $\bar{1}$, respectively (Dyer, 2010). The Inside algorithm (Algorithm 2) over the topologically sorted, acyclic hypergraph $\mathcal{H}$ under the tropical $\langle \mathbb{R}, \max, \times, -\infty, 0 \rangle$ semiring (Goodman, 1999; Mohri, 2009) computes the inside score α of the Viterbi hypothesis, i.e. the weight of its sequence of nodes ending in goal node g:

$$\arg\max_{h \in \mathcal{E}_q} P(h|q) \equiv \alpha(g) = \bigoplus_{h \in \mathcal{H}_q} \bigotimes_{e \in h} \mathcal{W}(e; \mathbf{w}_{smt}), \tag{5.4}$$

where

$$\mathcal{W}(e; \mathbf{w}_{smt}) = \mathbf{w}_{smt}^T \mathbf{\Phi}_{smt}(r(e))$$

assigns weights given parameters and features of the translation model.

For Bag-of-Words forced decoding (5.3), we extend $\mathcal{W}$ with another set of parameters $\mathbf{w}_{ir}$ for local IR features $\mathbf{\Phi}_{ir}$:

$$\arg\max_{h \in \mathcal{E}_q} P(h|q, d) \equiv \alpha(g) = \bigoplus_{h \in \mathcal{H}_q} \bigotimes_{e \in h} \mathcal{W}'(e, d; \mathbf{w}_{smt}, \mathbf{w}_{ir}), \tag{5.5}$$

with

$$\mathcal{W}'(e, d; \mathbf{w}_{smt}, \mathbf{w}_{ir}) = \mathbf{w}_{smt}^T \mathbf{\Phi}_{smt}(r(e)) + \mathbf{w}_{ir}^T \mathbf{\Phi}_{ir}(r(e), d).$$

Note that $\mathbf{\Phi}_{ir}$ depends on both translation rule $r(e)$ and document d, while $\mathbf{\Phi}_{smt}$ solely depends on source and target side of $r(e)$.

5.2.3 Decomposable Retrieval Features

In order to induce a ranking over documents, we use sparse, lexicalized IR features that relate derivations h to document d using $bm25$ term weights (see Section 1.5.1.3):

$$bm25(t, d) = rsj(t, \mathbf{C}) \cdot tf_{bm25}(t, d),$$

consisting of the Robertson/Sparck-Jones (rsj) weight, a constant term weight approximated on document frequencies in collection $\mathbf{C}$, and a term frequency weight that scales down very frequent document terms. Okapi $BM25$ parameters are set to $k_1 = 1.2$ and $b = 0.75$. We fire the $bm25$ term weight for each derivation term $t \in h$ with respect to document d in collection $\mathbf{C}$. The sum of feature values for all derivation terms $t_i \in h$ equals the regular $BM25$ score $BM25(h, d) = \sum_{t \in h} bm25(t, d)$. Weights $\mathbf{w}_{ir}$ for this type of features are interpretable as additional, general term weights.

We also experimented with a type of IR feature that excludes the $rsj(t, \mathbf{C})$ term. The feature values then solely consist of term frequency information in the document, and weights learned for such features should "recover" the Robertson/Sparck-Jones weight during learning of the model. However, we found in experiments that the inclusion of collection-specific information through the rsj-term was crucial for performance of BOW-FD.

Besides standard features for the SMT model, we considered another group of translation features. We fire lexicalized sparse features such as used in Chapter 3 and

Green et al. (2014) that indicate the translation, deletion, or insertion of terms in the hypothesis. By giving the model more degrees of freedom in adjusting lexical choice, we seek to learn dropping of common words that do not contribute to the retrieval score.

5.2.4 Default Retrieval Weights & Self-Translation

To enforce a ranking over documents, we define an *IR default weight* v, $\mathbf{w}_{ir} = \mathbf{1}v$. Intuitively, v controls the model's disposition to diverge from the SMT Viterbi path. If IR features fire in other regions of the search space than the SMT Viterbi path, this weight compensates for the loss incurred by not producing the Viterbi hypothesis. Furthermore, the default weight allows the model to generalize to unseen data: If an unknown query word, for example a named entity, causes an IR feature to fire at test time, the decoder will simply *pass through* the source word to any derivation, and the IR feature can contribute to the retrieval score with $v > 0$. This resembles the previously introduced concept of "self-translation" in Chapter 2, where words unknown to the translation model contribute to retrieval.

5.2.5 Multi-Sentence Queries

Unlike retrieval tasks such as mate finding (Chapter 2) or Wikipedia article retrieval (Chapter 4), where we regard queries as single sentences, specialized retrieval tasks such as patent prior art search (Chapter 3) may exhibit long, coherent search queries that contain multiple sentences. Multiple sentences need to be decoded separately, each producing a ranking over documents in collection $\mathbf{C}$.

To obtain a final ranking, we need to combine the sentence-wise rankings of a multi-sentence query $q = (s_1, \ldots, s_m)$. We model this task from a product of experts perspective (Hinton, 2002), where documents receive only high scores if each of the experts (sentences) agree on it. We multiply scores the $score(\cdot, d)$ of document d in all m sentence rankings and re-sort the final output. If d is not in the top-k ranking of a sentence, we take the minimum score of that top-k ranking as a smoothing value to prevent the product to become zero.

5.2.6 Implementation Details & Complexity Analysis

We implemented the BOW-FD model on top of the hierarchical phrase-based decoder `cdec` (Dyer et al., 2010), but there are no limitations for applying this approach to

Algorithm 3 BOW Forced Decoding for Retrieval.

```
Require: input query $q$, document collection $\mathbf{C}$, parameter weights $\mathbf{w}_{smt}, \mathbf{w}_{ir}$, size of ranking returned
    $k$, beam width $b$
 1:
 2: procedure RANK($q, \mathbf{C}, \mathbf{w}_{smt}, \mathbf{w}_{ir}, k, b$)
 3:     $\mathcal{H}_q, s_{smt} \leftarrow$ DECODE($q, \mathbf{w}_{smt}$)      ▷ construct +LM search space (default cdec)
 4:     $\mathbf{w}^a_{ir} \leftarrow$ MARKIREDGES($\mathcal{H}_q, \mathbf{w}_{ir}$)      ▷ select IR weights active for $\mathcal{H}_q$
 5:     return SCORE($\mathbf{w}^a_{ir}, \mathcal{H}_q, \mathbf{C}, s_{smt}, b$)
 6: end procedure
 7:
 8: procedure SCORE($\mathbf{w}_{ir}, \mathcal{H}, \mathbf{C}, s_{smt}, b$)
 9:     $S \leftarrow$ PRIORITYQUEUE
10:     for all $d \in \mathbf{C}$ do in parallel
11:         $\tilde{\mathbf{d}} \leftarrow \mathbf{\Phi}_{ir}(d) \odot \mathbf{w}_{ir}$      ▷ element-wise multiplication to factor in IR weights
12:         if $\tilde{\mathbf{d}}$ not empty then
13:             $s \leftarrow$ INSIDE($\mathcal{H}, \mathcal{W}', b$)      ▷ yields Inside score w.r.t. $\mathcal{W}'$ (Algorithm 2)
14:         else
15:             $s \leftarrow s_{smt}$
16:         end if
17:         PUSH($S, (d, s)$)
18:     end for
19:     return POP-K($S, k$)      ▷ return top-$k$ elements from the queue
20: end procedure
```

phrase-based systems (Koehn et al., 2007). Inference in the BOW-FD model involves the execution of Algorithm 2 for every document candidate $d \in \mathbf{C}$. We present two approaches of document filtering and approximate beam search decoding to minimize runtime. We furthermore analyze algorithmic complexity of the implemented approach.

5.2.6.1 Document Pre-Filtering

Procedurally, we compute the overlap of IR feature activations between edges in the search space and document candidates. This allows us to decide whether we need to execute the Inside algorithm or can assign a lower bound score.

High-level pseudo-code of the implementation is given in Algorithm 3. Inputs to the algorithm are query q, document collection $\mathbf{C}$, model parameters $\mathbf{w}_{smt}$ and $\mathbf{w}_{ir}$, the number of documents to return k, and beam width b. Documents $d \in \mathbf{C}$ are mapped to the IR feature space and represented as Bag-of-Words vectors (Section 5.2.3). The standard `cdec` algorithm constructs the +LM translation forest $\mathcal{H}$ in procedure DECODE (line 3) and also returns the score of the Viterbi derivation with respect to $\mathbf{w}_{smt}$, s_{smt}. Edge weights of the forest are set to $\mathcal{W}(e; \mathbf{w}_{smt}) = \mathbf{w}^T_{smt}\mathbf{\Phi}_{smt}(r(e))$.

We first observe that a naive approach of computing the inside score using the extended weight function $\mathcal{W}'$ (Equation 5.5) would involve the calculation of redundant products within the dot product between IR feature weights and corresponding model parameters at every edge in the hypergraph. For each document candidate, however, we can pre-compute the element-wise products between the document feature values and corresponding model parameters before running the Inside algorithm (line 11):

$$\tilde{\mathbf{d}} \leftarrow \mathbf{\Phi}_{ir}(d) \odot \mathbf{w}_{ir}.$$

This yields a "weighted" representation, $\tilde{\mathbf{d}}$, of document d, such that the IR term within weight function $\mathcal{W}'$ can be computed as a sum over those weighted features in $\tilde{\mathbf{d}}$ that are present in the yield of translation rule $r(e)$:

$$\mathcal{W}'(e, d; \mathbf{w}_{smt}, \mathbf{w}_{ir}) = \mathcal{W}(e; \mathbf{w}_{smt}) + \sum_{x \in \gamma(r(e))} \tilde{\mathbf{d}}(x).$$

$\gamma(r(e))$ denotes the set of active IR features ("yield") of translation rule $r(e)$ at hyperedge e. We compute such activation indicators as a side product in procedure MarkIREdges (line 4). The weighted representations of documents are implemented as hash maps, allowing lookups in constant time. The procedure MarkIREdges also produces a smaller set of model parameters, $\mathbf{w}^a_{ir}$, possibly active in the query search space hg_q. This reduced weight vector drastically reduces the size of weighted document representations $\tilde{\mathbf{d}}$.

More importantly, with such precomputed weighted representations of each document candidate, we can skip the execution of the Inside algorithm, if document and query search space do not share any IR features (line 12). Such candidates are assigned the SMT Viterbi score, s_{smt}, which constitutes a lower bound on the ranking score returned by the BOW-FD model. Our pre-filtering approach is similar to the coarse query approach in Dong et al. (2014), who score only documents that contain at least one term in the query lattice.

Scoring of documents can be embarrassingly parallelized since workers require only read access to the constant hypergraph object with SMT edge weights. The priority queue for retrieval scores is implemented as a min-heap of size k.

5.2.6.2 Approximate Decoding with a Beam

We further reduce runtime of the inference procedure by using approximate decoding. The Inside algorithm (Algorithm 2) visits every incoming hyperedge at each node and evaluates the joint edge weight as described above. We experimented with a beam search approach to limit the number of weight evaluations in Equation 5.5 for incoming edges at each node (parameter b in Algorithm 2). The max operation of the tropical semiring is discontinued once the number of considered incoming edges at a node exceeds the size of the beam (line 9).

5.2.6.3 Complexity Analysis

The complexity of constructing the +LM translation forest is common to BOW-FD and other SMT-based models, such as DT or PSQ, and thus not included in the following analysis.

For a single query q, forced decoding requires a single pass over the topologically sorted search space to find IR feature activations along hyperedges, yielding a complexity of $O(|V| + |E|)$. The dynamic programming procedure (Algorithm 2) for scoring a document requires another pass over the forest and evaluates the extended edge weight (5.5) for every edge $e \in E$. Note that the dot product for translation features is already precomputed by `cdec`. The retrieval part depends on the number of active IR features, $\omega := |\mathbf{\Phi}_{ir}(r(e), d)|$. Overall complexity for a single query and all documents $d \in \mathbf{C}$ is thus

$$O\Big(|V| + |E| + (|V| + |E| \cdot \omega) \cdot |\mathbf{C}|\Big). \tag{5.6}$$

As noted above, we reduce the quantity $|\mathbf{C}|$ by checking if a document candidate shares any IR features with the search space and avoid superfluous executions of the Inside algorithm. In our experiments on Wikipedia data, we found that this check reduced $|\mathbf{C}|$ to about 64% of its original size on average.

5.3 Learning to Decode for Retrieval

We now turn to the problem of learning parameter weights for the BOW-FD model. The objective function is no longer a translation measure such as BLEU (Papineni et al., 2002), but a measure of retrieval quality. This can be framed as a *learning-to-rank* problem if supervision in form of relevance judgments is available. Common retrieval performance measures are Mean Average Precision (MAP) (Baeza-Yates and Ribeiro-Neto, 1999), Normalized Discounted Cumulative Gain (NDCG) (Järvelin and Kekäläinen, 2002), and the recall-oriented Patent Retrieval Evaluation Score (PRES)

(Magdy and Jones, 2010). All of these measures are non-convex and discontinuous in their scores (Chaudhuri and Tewari, 2014), making them hard to optimize directly. Common learning-to-rank methods define surrogate loss functions, usually grouped into *point-wise*, *pair-wise*, and *list-wise* approaches. In this work, we chose to follow a pair-wise approach, where the problem of optimizing a ranked list is reduced to a binary classification problem of correctly ordering pairs of documents. Instead of optimizing a fully ordered list, as in list-wise approaches, we will sample preference pairs from the training data to be able to learn on large amounts of queries.

5.3.1 Pair-wise Learning-to-Rank

The objective is to prefer a relevant document d^+ over an irrelevant one d^- by assigning a higher score to d^+ than to d^-,

$$score(q, d^+; \mathbf{w}) > score(q, d^-; \mathbf{w}).$$

We sample a set of preference pairs

$$\mathcal{P} = \{(d^+, d^-) | rl(d^+, q) > rl(d^-, q)\}$$

from relevance-annotated data, where $rl(d, q)$ indicates the relevance level of a document given query. Furthermore, we require the difference of scores to satisfy a certain margin:

$$score(q, d^+; \mathbf{w}) > score(q, d^-; \mathbf{w}) + \Delta,$$

where the margin is defined as

$$\Delta = rl(d^+, q) - rl(d^-, q).$$

Our final objective is a margin-rescaled hinge-loss

$$L(\mathcal{P}) = \sum_{d^+, d^- \in \mathcal{P}} \left[score(q, d^-; \mathbf{w}) - score(q, d^+; \mathbf{w}) + \Delta \right]_+,$$

with $[\cdot]_+ = max(0, \cdot)$.

Algorithm 4 Pairwise Learning-to-Rank for BOW-FD using *Adadelta* (Zeiler, 2012) and Iterative Parameter Mixing (McDonald et al., 2010)

```
Require: Sampled preference pairs $\mathcal{P}$, number of epochs $E$, number of shards $j$, decay rate $\rho$,
    constant $\epsilon$
procedure IterativeParameterMixing($\mathcal{P}, E, \rho, \epsilon$)
    initialize parameters $\mathbf{w}_1$
    for $e = 1 : E$ do
        Shard $\mathcal{P}$ into $j$ pieces $\mathcal{P} = \{\mathcal{P}_1, \dots, \mathcal{P}_j\}$
        for all $\mathcal{P}_i : \mathcal{P}$ do in parallel
            $\mathbf{w}_i \leftarrow$ SingleEpochAdadelta($\mathcal{P}_i, \mathbf{w}_e, \rho, \epsilon$)
        end for
        $\mathbf{w}_{e+1} = \frac{1}{|\mathcal{P}|} \sum_{i=1}^{j} \mathbf{w}_i |\mathcal{P}_i|)$
    end for
    return $\mathbf{w}_e$
end procedure

procedure SingleEpochAdadelta($\mathcal{P}, \mathbf{w}_0, \rho, \epsilon$)
    $E[g^2]_0 = 0$; $E[\delta^2]_0 = 0$                    ▷ initialize Adadelta accumulation variables
    $t = 0$
    for all $p = (q, d^+, d^-) \in \mathcal{P}$ do
        $s^+ = score(q, d^+; \mathbf{w}_t)$; $s^- = score(q, d^-; \mathbf{w}_t)$
        if $|s^+ - s^-| < \Delta$ then
            $\mathbf{g}_t \leftarrow \nabla_{\mathbf{w}} L(p)$
            $E[g^2]_t = \rho E[g^2]_{t-1} + (1-\rho)\mathbf{g}_i^2$            ▷ Accumulate gradient values
            $\boldsymbol{\delta}_t = -\frac{\sqrt{E[\delta^2]_{t-1} + \epsilon}}{\sqrt{E[g^2]_t + \epsilon}} \mathbf{g}_t$                        ▷ Compute update
            $E[u^2]_t = \rho E[\delta^2]_{t-1} + (1-\rho)\boldsymbol{\delta}_t^2$              ▷ Accumulate updates
            $\mathbf{w}_{t+1} = \mathbf{w}_t + \boldsymbol{\delta}_t$
            $t = t + 1$
        end if
    end for
    return $\mathbf{w}_t$
end procedure
```

5.3.2 Learning Algorithm

Algorithm 4 shows the learning algorithm for BOW-FD. We perform stochastic (sub)gradient descent optimization using the *Adadelta* (Zeiler, 2012) update rule. In contrast to other per-dimensional learning rate methods such as *Adagrad* (Duchi et al., 2011), *Adadelta* does not require manual tuning of a global learning rate and requires only two hyperparameters: the sliding window decay rate $\rho = 0.95$ and a constant $\epsilon = 10^{-6}$. Both parameters were set to the default values given in the original paper, as changing these values was shown to have only minor effect on learning performance. *Adadelta* dynamically adapts the step size for gradient dimensions by using a sliding window approximation over past gradient values, and corrects for different units in the parameter updates using a first-order approximation of the Hessian. We also experimented with the `Vowpal Wabbit` toolkit (Agarwal et al., 2014) for parameter updates, but *Adadelta* yielded more stable results with less technical overhead.

We furthermore use the distributed learning technique of *Iterative Parameter Mixing* (McDonald et al., 2010), where multiple models on several shards of the training data are trained in parallel, and parameters are averaged after each epoch. Training of the models is carried out on a SunGridEngine cluster with 20 nodes. We perform incremental optimization using a *cyclic order* of the data sequence (Bertsekas, 2011), that is, the learner steps through a fixed sequence of pairs, query by query, and relevant document by relevant document, without randomization after epochs. This allows us to cache consecutive query search spaces and feature vectors for relevant documents. We sample preference pairs as follows: For each query in the training set, and for each of maximally s^- relevant documents, we sample s^- irrelevant documents, requiring at least a margin of 2. Regularization is done by early stopping where the best iteration is found on a held-out development set.

5.4 Evaluation on Patent Prior Art Search and Wikipedia Article Retrieval

5.4.1 Data & Systems

We conducted experiments on two large-scale CLIR tasks, namely German-English Wikipedia cross-lingual article retrieval on the WikiCLIR data set (Chapter 4), and patent prior art search with Japanese-English patent abstracts on the BoostCLIR data set[1] (Sokolov et al., 2013). In contrast to Chapter 3, we include relevance level (3) judgments for family patent abstracts. Family patent abstracts are almost always translations of the query abstract, and thus provide a special type of relevant document, the “cross-lingual mate”.

We present results for BOW-FD using a default weight v optimized on the respective development sets, and for models with parameters trained using pairwise learning-to-rank. We compute MAP (Baeza-Yates and Ribeiro-Neto, 1999), NDCG (Järvelin and Kekäläinen, 2002), and PRES (Magdy and Jones, 2010) scores on the top 1,000 returned documents to provide an extensive evaluation across precision- and

[1] `http://www.cl.uni-heidelberg.de/statnlpgroup/boostclir/`. last access: April 26, 2015

recall-oriented measures. Differences in evaluation scores between two systems were tested for statistical significance using paired randomization tests (Smucker et al., 2007). Significance levels are either indicated as superscripts, or provided in the captions of the respective tables.

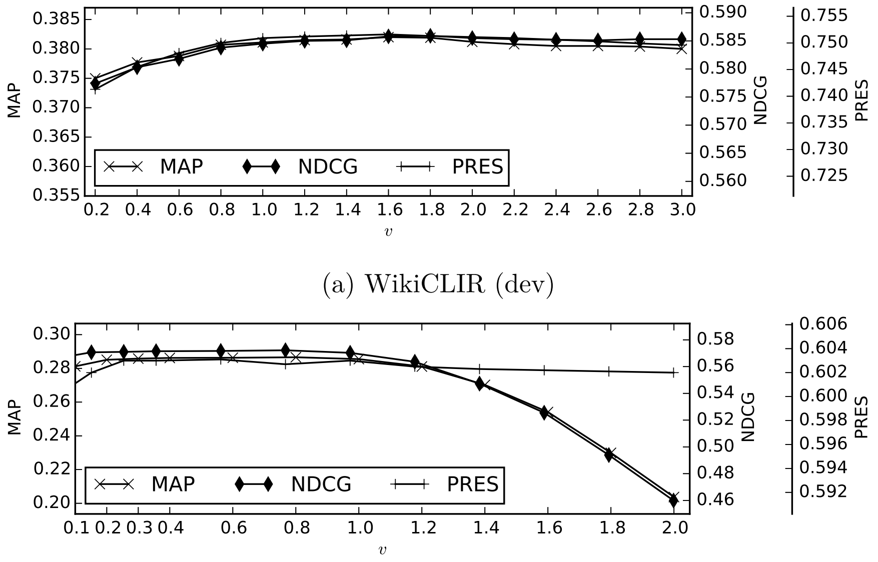
We compared retrieval performance of BOW-FD against the state-of-the-art SMT-based CLIR baselines of Direct Translation (DT) and cross-lingual Probabilistic Structured Queries (PSQ) (Ture et al., 2012a,b). Baseline SMT systems and BOW-FD share the hierarchical phrase-based SMT systems built with `cdec` (Dyer et al., 2010).

For German-English cross-lingual article retrieval on Wikipedia, we use a system previously built for the experiments in Schamoni et al. (2014) from parallel training data (over 104M words) consisting of the Europarl corpus (Koehn, 2005) in version 7, the News Commentary corpus, and the Common Crawl corpus (Smith et al., 2013). Word alignments were created with `fast_align` (Dyer et al., 2013). The 4-gram language model was trained with the KenLM toolkit (Heafield, 2011) on the English side of the training data and the English Wikipedia articles. Language model scores are added to the search spaces using the cube pruning algorithm (Huang and Chiang, 2007) with $\textit{poplimit} = 200$. SMT model parameters were optimized using MIRA (Chiang et al., 2008) on the WMT'11 News test set (3,003 sentences). The parameters for the baseline PSQ model were found on the WikiCLIR development set (Chapter 4) consisting of 10,000 German queries using 1,000-best lists: interpolation parameter $\lambda = 0.4$, lower threshold $L = 0$, and cumulative threshold $C = 1$.

For the task of Japanese-English patent prior-art search, we use a system previously trained for experiments in Sokolov et al. (2013) and Schamoni et al. (2014). Its SMT features were trained on 1.8M parallel sentences of NTCIR-7 data (Fujii et al., 2008) and weights were tuned on the NTCIR-8 test collection (2,000 sentences) using MIRA (Chiang et al., 2008). A 5-gram language model on the English side of the training data was trained with the KenLM toolkit (Heafield, 2011). The system uses a cube pruning poplimit of 30. Parameters for the baseline PSQ model were found on the BoostCLIR development set of 2,000 patent abstract queries and set to n-best list size $= 1000$, $\lambda = 1.0$, $L = 0.005$, $C = 0.95$.

5.4.2 Experiments & Results

Default Weight Grid Search. We first found a default weight v using grid search within $v = [0, 3]$ and $v = [0, 2]$ on the development sets for WikiCLIR and BoostCLIR, respectively. v controls the balance between retrieval and translation features and with



(a) WikiCLIR (dev)

(b) BoostCLIR (dev)

Figure 5.2: Retrieval performance as a function of default weight $v : \mathbf{w}_{ir} = \mathbf{1}v$.

larger v, the model is more likely to produce query derivations diverging from the SMT 1-best translation. For WikiCLIR, we sampled 1,000 out of 10,000 queries to reduce the time of the grid search. For BoostCLIR, we used the full development set of 2,000 queries with 8,381 sentences. We combine rankings for single-sentence queries from multi-sentence patent abstracts using the product method as described in Section 5.2.5. Well-performing values were found at $v = 1.6$ for WikiCLIR, and $v = 0.8$ for BoostCLIR, respectively. Figure 5.2 shows results of the grid search on both data sets. Scores for Wikipedia remain relatively stable for $v > 1.0$. The SMT feature with the largest weight for the German-English system is the language model (0.69), suggesting that a much larger default weight is required in order to diverge from the SMT Viterbi derivation. For the patent retrieval task, larger default weights decrease MAP and NDCG scores significantly. A closer analysis on the stability of PRES for BOW-FD is given below.

Default Weight Test Results. Table 5.1 shows test set performance of DT and PSQ baselines versus BOW-FD on both data sets. Scores for DT and PSQ are as reported in Schamoni et al. (2014). We observe that BOW-FD significantly outperforms both baselines by over 2 points on WikiCLIR and BoostCLIR under all three evaluation

	WikiCLIR			**BoostCLIR**		
	MAP	NDCG	PRES	MAP	NDCG	PRES
DT	.3678	.5691	.7219	.2554	.5397	.5680
PSQ	.3642	.5671	.7165	.2659	.5508	.5851
BOW-FD	*.3880	*.5911	*.7417	*.2825	*.5721	*.6072
BOW-FD+LTR	†.3913	†.5962	†**.7543**	†.2870	†.5807	†**.6260**
BOW-FD+LEX+LTR	†**.3919**	†**.5963**	†.7528	†**.2883**	†**.5819**	†.6251

Table 5.1: Retrieval results of baseline systems and BOW-FD with default weight $v = 1.6$ for WikiCLIR and $v = 0.8$ for BoostCLIR, respectively. Baseline and BOW-FD models use the same SMT system. Significant differences at $p = 10^{-4}$ with respect to baselines are indicated with *. Significant differences at $p = 10^{-6}$ of learning-to-rank-based models (LTR) with respect to BOW-FD are indicated with †.

measures (at $p = 10^{-4}$). While the German-English SMT system uses a cube pruning poplimit of 200 for the WikiCLIR experiments, the Japanese-English SMT system uses a poplimit of 30. This may reduce the diversity of the search space considerably. In order to compare to the scores given in Schamoni et al. (2014), we nevertheless report BOW-FD results with poplimit 30 in the table. Increasing the poplimit from 30 to 200 for the Japanese-English system yielded another significant gain of BOW-FD over both baselines (MAP=0.2893, NDCG=0.5807, PRES=0.6172).

Cube Pruning Poplimit Effect. Figure 5.3 shows evaluation scores on the Wikipedia development set as a function of the cube pruning poplimit $[100, 800]$. The solid line shows the default weight optimized BOW-FD system if the size of the search space is increased. We can observe a ceiling effect for precision-oriented metrics, MAP and NDCG, whereas PRES slightly benefits from larger search spaces in which the model can reach more query term translation alternatives. We carried out the same experiment with a German-English system, using only a bi-gram language model. The smaller history of the language model causes the +LM forest to be closer to the -LM forest, due to fewer state splits during cube pruning. Here, we see that precision-oriented metrics actually decrease with higher poplimits, indicating that the use of a language model is key for precision in CLIR. Again, we see the reversed effect for PRES, suggesting that BOW-FD produces increased recall with higher poplimits. For an in-depth explanation of this behavior, see Section 5.4.3.

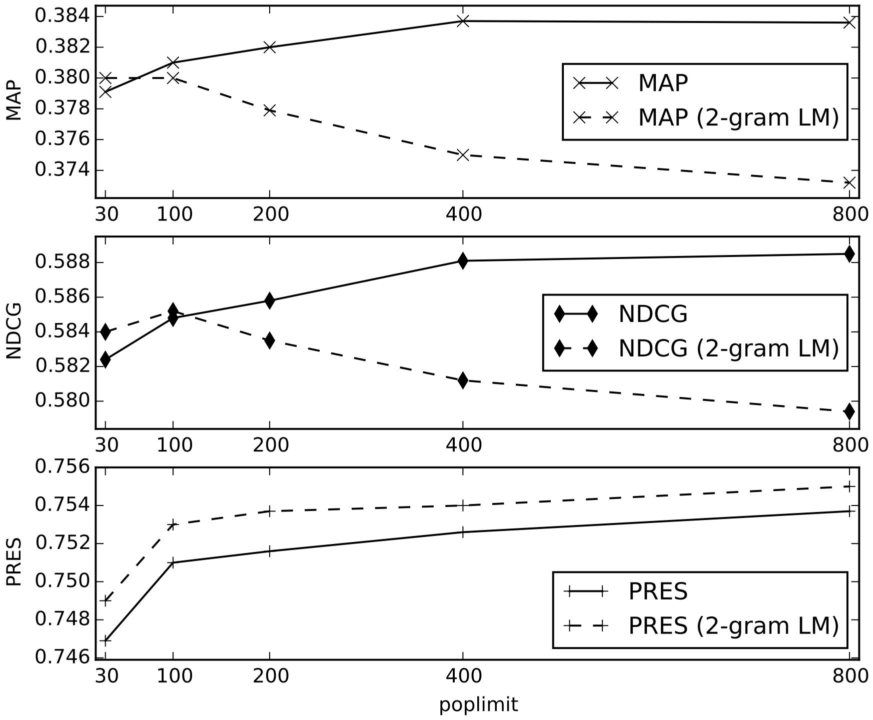
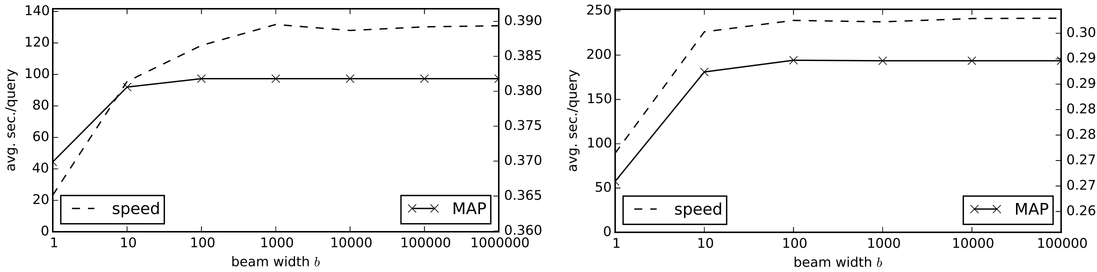


Figure 5.3: BOW-FD retrieval performance as a function of the cube pruning poplimit for the German-English SMT system on the WikiCLIR development set.

Approximate decoding results. We also evaluated the technique of approximate decoding as described in Section 5.2.6.1. The goal of this experiment is to illustrate the trade-off between speed and quality in the BOW-FD model. Figure 5.4 shows MAP scores and average time taken per query if the beam size parameter b is varied. With larger beam settings, the execution of the Inside procedure (Algorithm 2) becomes slower. We see that for both systems, MAP stabilizes at around $b = 100$. While the average time taken per query continues to increase slightly for larger b, the results show that the average density of the hypergraphs (number of incoming edges per node) rarely exceeds the low hundreds. We thus conclude that the use of a beam has only limited effects on runtime. Note that the absolute speed values should be viewed with a grain of salt. All beam width experiments were carried out single-threaded. As shown in Algorithm 3, we can score multiple documents simultaneously and process a single query much faster on average in a multi-threaded environment.



(a) WikiCLIR (dev)

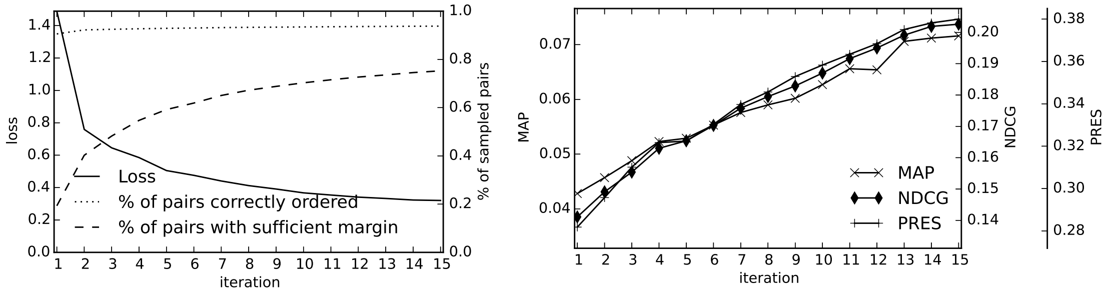
(b) BoostCLIR (dev)

Figure 5.4: Speed and retrieval performance as a function of beam width b (Algorithm 2).

Learning-to-rank results. We first conducted an experiment to establish the correctness of Algorithm 4, initializing all model weights to 0. Figure 5.5 shows learning curves over 15 iterations on a reduced training set of WikiCLIR. Figure 5.5a indicates a state close to convergence, as shown by the hinge loss curve and the number of correctly ordered, and sufficiently separated pairs. Training performance continues to improve steadily (Figure 5.5b). However, overall retrieval scores are significantly lower than DT and PSQ baselines, or BOW-FD models with pre-trained SMT weights and grid search optimized IR default weights. The fact that training for 15 iterations on this reduced training set took over a week, led us to conduct learning-to-rank experiments with MIRA-initialized SMT weights and IR default weights.

We ran Algorithm 4 on both data sets for up to 10 iterations, choosing the final model by evaluating each epoch on the respective development sets. Model parameters were initialized from grid search-optimized IR default weights and pre-trained SMT weights by MIRA. We found significant improvements over grid search optimized BOW-FD models in precision-oriented metrics, MAP and NDCG, when freezing dense SMT weights. Freezing dense SMT weights allows the model to maintain translation-benign search spaces, while preferred lexical choice for retrieval is learned through sparse lexical alignment features.

Table 5.1 shows that BOW-FD+LTR, with and without lexical alignment features, significantly outperforms BOW-FD on both data sets, with the largest improvements for PRES. Differences between models with and without lexical alignment features are not statistically significant. We conjecture that LTR models mostly optimize recall rather than precision, i.e. they return more relevant documents. This is supported by the fact that BOW-FD+LTR retrieves 70.1% of the relevant documents in



(a) loss

(b) performance (train)

Figure 5.5: Learning a BOW-FD model from zero-initialized weights is slow.

	WikiCLIR			**BoostCLIR**		
	MAP	NDCG	PRES	MAP	NDCG	PRES
DT	.3347$^{(-.03)}$	.5368$^{(-.03)}$	.6970$^{(-.03)}$	.2315$^{(-.02)}$	.5105$^{(-.03)}$	.5420$^{(-.03)}$
PSQ	.3464$^{(-.02)}$	.5483$^{(-.02)}$	.7006$^{(-.02)}$	.2460$^{(-.02)}$	.5290$^{(-.02)}$	.5672$^{(-.02)}$
BOW-FD	.3218$^{(-.07)}$	.5315$^{(-.06)}$	.7220$^{(-.02)}$	.1651$^{(-.12)}$	.4185$^{(-.15)}$	.4959$^{(-.11)}$

Table 5.2: SMT-based CLIR models without a language model. Numbers in superscripts denote the absolute loss with respect to equivalent systems in Table 5.1.

the Wikipedia test set, compared to 68.0% by BOW-FD, while the Mean Reciprocal Rank (MRR) hardly differs (0.7344 vs. 0.7332). An experiment with no pre-trained SMT weights or default IR weights performed worse, indicating the importance of translation-benign search spaces and IR default weights for generalization to unseen terms.

5.4.3 Importance of Language Model for Retrieval

Liu et al. (2012) and Dong et al. (2014) claim that computationally expensive SMT feature functions such as language models have only minor impact on CLIR performance of SMT-based models. We found that such context-sensitive information present in single 1-best query translations (DT), weighted translation alternatives from the n-best list (PSQ), and forced decoding in a “translation-benign” search space (BOW-FD) is crucial for retrieval performance in our experiments. In order to investigate the question of the importance of context-sensitive information, such as language model scores for retrieval, we conducted an experiment in which the language model information is removed from all three SMT-based models. For the PSQ models, we also set the parameter λ to 1.0 to disable interpolation with the context-free lexical translation

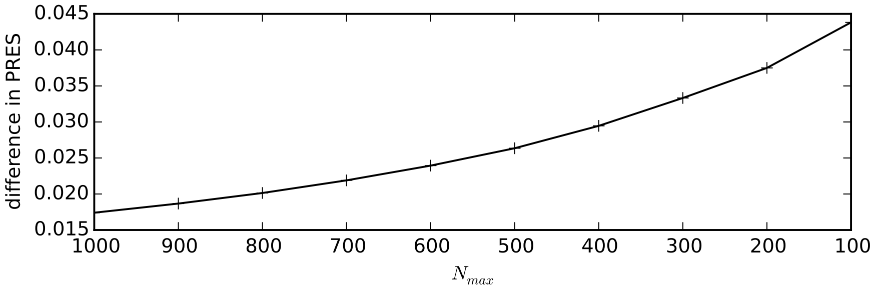


Figure 5.6: Difference in PRES scores on the WikiCLIR development set as a function of PRES's N_{max} parameter between BOW-FD +LM and -LM systems.

table (Ture et al., 2012a).

Table 5.2 shows that retrieval performance drops significantly for all models. The drop in performance for both baseline models is comparable on both data sets. Removing the language model for BOW-FD hurts performance the most (with an average drop of 6 points in MAP and NDCG scores for WikiCLIR, and over 11 points in all measures for BoostCLIR). However, scores for recall-oriented PRES on WikiCLIR remains relatively stable for BOW-FD with and without a language model. A closer analysis on the rankings for BOW-FD on WikiCLIR shows that the -LM model returns 1,589 (out of 86,994) relevant documents less than the +LM model. However, only 2 documents with relevance level 3, these are directly linked cross-lingual "mates", were no longer retrieved, suggesting that excluding the language model from the system mostly affects the retrieval of "non-mates", i.e. documents that are linked by, or link to the cross-lingual mate. We explain this behavior as follows: Cross-lingual mates are likely to contain words that are close to an adequate query translation, since they constitute the beginning of a Wikipedia article with the same topic as the query. Derivations generated for these documents are such that both translation model features (with or without the LM) *and* retrieval features agree on a path close to the SMT Viterbi translation. In contrast, other relevant documents require more non-standard lexical choices, which are harder to achieve in a +LM search space, since the strong weight on the language model, plus a language model-driven pruning technique, strongly favor lexical choices that agree with the language model's concept of fluency. In a -LM search space, disfluent derivations are easily reached by IR feature activations, whose default weight is much larger in relation to the remaining SMT features. The use of "glue rules", allowing left-to-right concatenation of partial translations, along with loosely extracted synchronous grammar rules, give hierarchical MT models large degrees of freedom in producing very disfluent translations in the -LM space. If a language model is not ensuring a more or less “translation-benign” search space, the “reachability” of terms in irrelevant documents is increased, causing them to interfere with the ranking of relevant documents that may be closer translations of the query. This behavior immediately affects precision-oriented scores such as MAP and NDCG, while PRES is only affected if its recall cutoff parameter, N_{max}, is lowered, as shown in Figure 5.6.

The major drop in performance for patent data may be explained with the way multiple sentence queries are evaluated: A language model limits diversity of translation options for multiple sentences. Without a language model, the sets of documents retrieved by each sentence are almost disjoint, i.e. the sentences do not agree on a common set of documents.

5.5 Discussion & Future Work

In this chapter, we presented an approach that switches the retrieval focus in state-of-the-art CLIR to a translation focus by forcing a standard SMT decoder to produce a Bag-of-Words representation of the document repository. This is done by joint optimization of a linear model including both translation and retrieval features under a ranking objective. Highly weighted term-match features are then used to find a decoding path that gives highest score to the document that is optimal with respect to both relevance and translational adequacy. We showed in a large-scale evaluation on cross-lingual retrieval tasks in the domains of patents and Wikipedia pages that our approach significantly outperforms Direct Translation and Probabilistic Structured Query approaches under a variety of evaluation metrics. Furthermore, we investigated the role of context-sensitive information such as language model scores in retrieval. In contrast to previous claims about the minor impact of language models in retrieval performance in SMT-based CLIR, we found significant drops MAP, NDCG, and PRES scores across all models when removing language model information. This confirms the dual role of the language model to ensure fluency, and to select the proper translation terms in the context of the neighboring target terms. The latter role of the language model makes it an indispensable ingredient of any SMT-based CLIR approach.

Open questions in our work regard further improvements in efficiency of retrieval. So far we could achieve substantial reductions in retrieval complexity by pre-filtering the document collection based on coarse term matches. The inherent complexity of SMT decoding is less of a problem in offline applications such as translation retrieval (Dong et al., 2014), but it becomes prohibitive in online applications such as cross-lingual web search. In future work, we would like to address efficiency, e.g. by investigating the possibility of incorporating an inverted index into online applications of forced decoding. Furthermore, one could explore other approaches to approximate decoding, such as the use of future cost estimates to design an admissible heuristic, allowing A^*-like search for the best derivation. Another direction of research could be a more intelligent way of evaluating edges within the dynamic programming procedure: If the number of considered edges is limited by a beam size, one can imagine ways to sort the order of edge evaluations according to some retrieval heuristic, thus guaranteeing to evaluate only the most retrieval-informative edges.

Chapter 6

Conclusions

This thesis presented several approaches to the problem of ranking for translation-based Cross-Language Information Retrieval (CLIR). Through cross-lingual mate ranking and mining of comparable data, we adapted an out-of-domain translation system to the domain of the search collection, namely Twitter messages. By extending this to an iterative approach, we showed that Statistical Machine Translation (SMT) and Cross-Language Information Retrieval can benefit from each other. Namely, a CLIR model provides additional training data for SMT training, which subsequently enables improved translation-based retrieval.

In Chapter 3, we focused on the aspect of parameter optimization for SMT models used in Cross-Language Information Retrieval. As expounded in the introduction, translation for Cross-Language Information Retrieval differs from standard reference-optimized translation. Such differences should be considered when SMT models are integrated into Cross-Language Information Retrieval. We thus introduced a measure of retrieval quality for discriminative SMT training. This decomposable measure, based on the well established $BM25$ metric, allows ranking of translations, i.e. derivations within the search space, to be efficiently evaluated according to their retrieval utility. Despite only moderate improvements over a Direct Translation baseline, retrieval oracles clearly showed the expressive power of the translation model to match relevant documents. We claim that an effective Cross-Language Information Retrieval system design, on the basis of Statistical Machine Translation, should strive to make use of the expressive freedom in query translation search spaces.

For the core retrieval task of document ranking, we thus presented a new model that allows efficient query decoding with respect to candidates in the document repository. By once again exploiting the decomposability of $BM25$, we extended the standard SMT dynamic programming procedure to a ranking function over documents, essentially combining translation and retrieval into a single linear model. This combination solved two problems at once: first, we allow the model to pick the best translation alternatives with respect to the current document candidate, and do not need to commit to a fixed distribution over query expansion terms, as is the case in the Probabilistic Structured Query framework. Second, a combined linear model for document ranking gives way to the application of established learning-to-rank techniques directly for CLIR, allowing the joint optimization of translation- and retrieval-related feature weights in the model. We empirically showed substantial gains over state-of-the-art CLIR models of Direct Translation and Probabilistic Structured Queries. From these results, we conclude that the key for designing robust cross-lingual retrieval systems is a joint modeling approach, where each component is aware of its use in the larger context.

Lastly, another central finding in this work is, once again, the importance of context-sensitive information. While standard retrieval models traditionally use context-free Bag-of-Words representations of their documents and queries, Cross-Language Information Retrieval clearly benefits from context-sensitive information at translation time. Even when retrieval features decompose nicely over terms, a context-sensitive selection of (translated) query terms reduces the problem of query drift significantly. For combined models of translation and retrieval, such as in Chapter 5, language models provide the means to accomplish this. They are thus not only central to Statistical Machine Translation, as shown many times before, but should also become a key ingredient for Cross-Language Information Retrieval, if the trend towards natural language queries continues to persist.

List of Figures

List of Tables

List of Algorithms

List of Abbreviations

BOW	Bag-of-Words
BOW-FD	Bag-of-Words Forced Decoding
CLIR	Cross-Language Information Retrieval
DT	Direct Translation
PSQ	Probabilistic Structured Queries
EM	Expectation Maximization
HIT	Human Intelligence Task
MAP	Mean Average Precision / Maximum a Posteriori
MERT	Minimum Error Rate Training
MIRA	Margin Infused Relaxed Algorithm
NDCG	Normalized Discounted Cumulative Gain
NIST	National Institute for Standards and Technology
NTCIR	NII Testbeds and Community for Information access Research
OOVs	Out-of-Vocabulary Words
PRES	Patent Retrieval Evaluation Score
PRO	Pairwise Ranking Optimization
SMT	Statistical Machine Translation
SVM	Support Vector Machine
TREC	Text REtrieval Conference
WSCFG	Weighted Synchronous Context-Free Grammar

Bibliography

Sadaf Abdul-Rauf and Holger Schwenk. On the use of comparable corpora to improve SMT performance. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL'09)*, Athens, Greece, 2009.

Steven Abney. *Semisupervised Learning for Computational Linguistics.* Chapman and Hall, London, UK, 2008.

Alekh Agarwal, Oliveier Chapelle, Miroslav Dudík, and John Langford. A reliable effective terascale linear learning system. *Journal of Machine Learning Research*, 15:1111–1133, 2014.

Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern information retrieval.* Addison-Wesley Longman Publishing Co, Boston, Massachussetts, 1999.

Bing Bai, Jason Weston, David Grangier, Ronan Collobert, Kunihiko Sadamasa, Yanjun Qi, Olivier Chapelle, and Kilian Weinberger. Learning to rank with (a lot of) word features. *Information Retrieval*, 13(3):291–314, 2010.

Nicholas Belkin, Diane Kelly, Giyeong Kim, Jayoung Kim, Hyukjin Lee, Gheorghe Muresan, Muh Chyun Tang, Xiaojun Yuan, and Colleen Cool. Query length in interactive information retrieval. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval (SIGIR'03)*, Toronto, Canada, 2003.

Adam Berger and John Lafferty. Information retrieval as statistical translation. In *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'99)*, Berkeley, CA, 1999.

Adam Berger, Rich Caruana, David Cohn, Dayne Freitag, and Vibhu Mittal. Bridging the lexical chasm: Statistical approaches to answer-finding. In *Proceedings of the*

23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'00), Athens, Greece, 2000.

Nicola Bertoldi and Marcello Federico. Domain adaptation for statistical machine translation with monolingual resources. In *Proceedings of the 4th Workshop on Statistical Machine Translation (WMT'09)*, Athens, Greece, 2009.

Dimitri P. Bertsekas. Incremental gradient, subgradient, and proximal methods for convex optimization: A survey. In Suvrit Sra, Sebastian Nowozin, and Stephen J. Wright, editors, *Optimization for Machine Learning*. MIT Press, 2011.

Andrei Broder. A taxonomy of web search. *SIGIR Forum*, 36(2):3–10, 2002.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, Meredith J. Goldsmith, Jan Hajic, Robert L. Mercer, and Surya Mohanty. But dictionaries are data too. In *Proceedings of the Workshop on Human Language Technology (HLT'93)*, Plainsboro, New Jersey, 1993a.

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311, 1993b.

Marine Carpuat and Michel Simard. The trouble with smt consistency. In *Proceedings of the Seventh Workshop on Statistical Machine Translation (WMT'12)*, Montreal, Canada, 2012.

Daniel Cer, Christopher D. Manning, and Daniel Jurafsky. Positive diversity tuning for machine translation system combination. In *Proceedings of the Eighth Workshop on Statistical Machine Translation (WMT'13)*, Sofia, Bulgaria, 2013.

Sougata Chaudhuri and Ambuj Tewari. Perceptron-like algorithms and generalization bounds for learning to rank. *arXiv preprint arXiv:1405.0591*, 2014.

David Chiang. Hierarchical phrase-based translation. *Computational Linguistics*, 33 (2):201–228, 2007.

David Chiang. Hope and fear for discriminative training of statistical translation models. *Journal for Machine Learning Research*, 13(1):1159–1187, 2012.

David Chiang, Yuval Marton, and Philip Resnik. Online large-margin training of syntactic and structural translation features. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'08)*, Honolulu, Hawaii, 2008.

Jeffrey Chin, Maureen Heymans, Alexandre Kojoukhov, Jocelyn Lin, and Hui Tan. Cross-language information retrieval. Patent Application, 2008. US 2008/0288474 A1.

Noam Chomsky. *Aspects of the theory of syntax.* MIT Press, Cambridge, Massachusetts, 1965.

Charles Clarke, Nick Craswell, and Ian Soboroff. Overview of the trec 2004 terabyte track. In *Proceedings of the 13th Text Retrieval Conference (TREC'04)*, 2004.

Paul R. Cohen. *Empirical Methods for Artificial Intelligence.* MIT Press, Cambridge, MA, USA, 1995.

Nick Craswell, David Hawking, Ross Wilkinson, and Mingfang Wu. Overview of the trec 2003 web track. In *Proceedings of the 12th Text Retrieval Conference (TREC'03)*, 2003.

Kareem Darwish and Douglas W. Oard. Probabilistic structured query methods. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval (SIGIR'03)*, Toronto, Canada, 2003.

Hal Daumé and Jagadeesh Jagarlamudi. Domain adaptation for machine translation by mining unseen words. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT'11)*, Portland, Oregon, 2011.

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

Meiping Dong, Yong Cheng, Yang Liu, Jia Xu, and Maosong Sun. Query lattice for translation retrieval. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING'14)*, Dublin, Ireland, 2014.

Doug Downey, Susan Dumais, Dan Liebling, and Eric Horvitz. Understanding the relationship between searchers' queries and information goals. In *Proceedings of the 17th ACM conference on Information and knowledge management (CIKM'08)*, Napa Valley, California, 2008.

Sergio Duarte-Torres, Djoerd Hiemstra, and Pavel Serdyukov. Query log analysis in the context of information retrieval for children. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval (SIGIR'10)*, Geneva, Switzerland, 2010.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research*, 12:2121–2159, 2011.

Christopher Dyer. *A Formal Model of Ambiguity and Its Applications in Machine Translation*. PhD thesis, University of Maryland, College Park, Maryland, 2010.

Christopher Dyer, Adam Lopez, Juri Ganitkevitch, Jonathan Weese, Ferhan Ture, Phil Blunsom, Hendra Setiawan, Vladimir Eidelman, and Philip Resnik. cdec: A decoder, alignment, and learning framework for finite-state and context-free translation models. In *Proceedings of the ACL-10 System Demonstrations (ACL'10)*, Uppsala, Sweden, 2010.

Christopher Dyer, Victor Chahuneau, and Noah Smith. A simple, fast, and effective reparameterization of IBM Model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technologies (NAACL-HLT'13)*, Atlanta, Georgia, 2013.

Christiane Fellbaum. *WordNet: An Electronic Lexical Database*. Bradford Books, 1998.

Atsushi Fujii, Masao Utiyama, Mikio Yamamoto, and Takehito Utsuro. Overview of the patent translation task at the NTCIR-7 workshop. In *Proceedings of the 7th NII Testbeds and Community for Information access Research Workshop (NTCIR-7'08)*, Tokyo, Japan, 2008.

Atsushi Fujii, Masao Utiyama, Mikio Yamamoto, and Takehito Utsuro. Evaluating effects of machine translation accuracy on cross-lingual patent retrieval. In *Proceed-*

ings of the 32rd international ACM SIGIR conference on Research and development in information retrieval (SIGIR'09), Boston, Massachusetts, 2009.

Giorgio Gallo, Giustino Longo, Stefano Pallottino, and Sang Nguyen. Directed hypergraphs and applications. *Discrete Applied Mathematics – Special issue: combinatorial structures and algorithms*, 42(2-3):177–201, 1993.

Kevin Gimpel and Noah Smith. Structured ramp loss minimization for machine translation. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technologies (NAACL-HLT'12)*, Stroudsburg, Pennsylvania, 2012.

Joshua Goodman. Semiring parsing. *Computational Linguistics*, 25(4):573–605, 1999.

Erik Graf and Leif Azzopardi. A methodology for building a patent test collection for prior art search. In *Proceedings of the 2nd International Workshop on Evaluating Information Access (EVIA'08)*, Tokyo, Japan, 2008.

Spence Green, Daniel Cer, and Christopher D. Manning. An empirical comparison of features and tuning for phrase-based machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation (WMT'14)*, Baltimore, Maryland, 2014.

Yunsong Guo and Carla Gomes. Ranking structured documents: A large margin based approach for patent prior art search. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI'09)*, Pasadena, CA, 2009.

Nizar Habash, Owen Rambow, and Ryan Roth. Mada+Tokan: A toolkit for arabic tokenization, diacritization, morphological disambiguation, pos tagging, stemming and lemmatization. In *Proceedings of the 2nd International Conference on Arabic Language Resources and Tools (MEDAR'09)*, Cairo, Egypt, 2009.

Kenneth Heafield. KenLM: faster and smaller language model queries. In *Proceedings of the EMNLP 2011 Sixth Workshop on Statistical Machine Translation (WMT'11)*, Edinburgh, UK, 2011.

Felix Hieber and Stefan Riezler. Improved Answer Ranking in Social Question-Answering Portals. In *Proceedings of the 3rd International Workshop on Search and Mining User-generated Contents (SMUC 2011)*, pages 19–26, Glasgow, Scotland, UK, 2011.

Felix Hieber and Stefan Riezler. Bag-of-words forced decoding for cross-lingual information retrieval. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technologies (NAACL-HLT'15)*, Denver, CO, USA, 2015.

Felix Hieber, Laura Jehl, and Stefan Riezler. Task alternation in parallel sentence retrieval for twitter translation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL'13)*, Sofia, Bulgaria, 2013.

Geoffrey E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002.

Mark Hopkins and Jonathan May. Tuning as ranking. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, Edinburgh, United Kingdom, 2011.

Liang Huang and David Chiang. Better k-best parsing. In *Proceedings of the Ninth International Workshop on Parsing Technology (IWPT'05)*, Vancouver, Canada, 2005.

Liang Huang and David Chiang. Forest rescoring: Faster decoding with integrated language models. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL'07)*, Prague, Czech Republic, 2007.

Zhongqiang Huang, Martin Čmejrek, and Bowen Zhou. Soft syntactic constraints for hierarchical phrase-based translation using latent syntactic distributions. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP'10)*, Cambridge, Massachusetts, 2010.

Bernard J. Jansen, Amanda Spink, and Tefko Saracevic. Real life, real users, and real needs: A study and analysis of user queries on the web. *Information Processing and Management*, 36(2):207–227, 2000.

Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002.

Laura Jehl, Felix Hieber, and Stefan Riezler. Twitter translation using translation-based cross-lingual retrieval. In *Proceedings of the Seventh Workshop on Statistical Machine Translation (WMT'12)*, Montreal, Quebec, Canada, 2012.

Dan Klein and Christopher D. Manning. Parsing and hypergraphs. In *Proceedings of the Seventh International Workshop on Parsing Technologies (IWPT'01)*, Beijing, China, 2001.

Kevin Knight. Decoding complexity in word-replacement translation models. *Computational Linguistics*, 25(4):607–615, 1999.

Philipp Koehn. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of Machine Translation Summit X*, Phuket, Thailand, 2005.

Philipp Koehn. *Statistical Machine Translation.* Cambridge University Press, New York, New York, 1st edition, 2010.

Philipp Koehn and Josh Schroeder. Experiments in domain adaptation for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation (WMT'07)*, Prague, Czech Republic, 2007.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (NAACL'03)*, Edmonton, Canada, 2003.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Christopher Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the ACL-07 2007 Demo and Poster Sessions (ACL'07)*, Prague, Czech Republic, 2007.

Hans P. Krings and Geoffrey S. Koby, editors. *Repairing texts : empirical investigations of machine translation post-editing processes.* Kent State University Press, Kent, Ohio, 2001.

Frank LaRue. Internet should remain as open as possible. In *Seventeenth Session of the Human Rights Council: Report of the Special Rapporteur on the promotion and protection of the right to freedom of opinion and expression, Frank La Rue*, 2011.

Percy Liang, Alexandre Bouchard-Côté, Dan Klein, and Ben Taskar. An end-to-end discriminative approach to machine translation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting* *of the Association for Computational Linguistics444 (COLING-ACL'06)*, Sydney, Australia, 2006.

Chin-Yew Lin and Franz Josef Och. Orange: a method for evaluating automatic evaluation metrics for machine translation. In *Proceedings the 20th International Conference on Computational Linguistics (COLING'04)*, Geneva, Switzerland, 2004.

Michael Littman, Susan T. Dumais, and Thomas K. Landauer. Automatic cross-language information retrieval using latent semantic indexing. In *Cross-Language Information Retrieval*, pages 51–62. Kluwer Academic Publishers, 1998.

Chunyang Liu, Qi Liu, Yang Liu, and Maosong Sun. Thutr: A translation retrieval system. In *Proceedings of COLING'12: Demonstration Papers*, Bombay, India, 2012.

Ding Liu and Daniel Gildea. Semantic role features for machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING'10)*, Beijing, China, 2010.

Tie-yan Liu, Jun Xu, Tao Qin, Wenying Xiong, and Hang Li. LETOR: Benchmark dataset for research on learning to rank for information retrieval. In *Proceedings of the SIGIR Workshop on Learning to Rank for Information Retrieval (SIGIR'07)*, Amsterdam, Netherlands, 2007.

Walid Magdy and Gareth Jones Jones. PRES: A score metric for evaluating recall-oriented information retrieval applications. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'10)*, Geneva, Switzerland, 2010.

Walid Magdy and Gareth Jones Jones. Studying machine translation technologies for large-data CLIR tasks: a patent prior-art search case study. *Information Retrieval*, 17(5-6):492–519, 2013.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval.* Cambridge University Press, New York, New York, 2008.

Yuval Marton and Philip Resnik. Soft syntactic constraints for hierarchical phrased-based translation. In *Proceedings of the 46st Annual Meeting of the Association for Computational Linguistics (ACL'08)*, Columbus, Ohio, 2008.

Ryan McDonald, Keith Hall, and Gideon Mann. Distributed training strategies for the structured perceptron. In *Proceedings of the 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technologies (NAACL-HLT'10)*, Los Angeles, California, 2010.

Coskun Mermer, Murat Saraçlar, and Ruhi Sarikaya. Improving statistical machine translation using bayesian word alignment and gibbs sampling. *IEEE Transactions on Audio, Speech and Language Processing*, 21(5):1090–1101, 2013.

Mehryar Mohri. Weighted automata algorithms. In *Handbook of weighted automata*, pages 213–254. Springer Berlin Heidelberg, 2009.

Robert C. Moore. Improving IBM word alignment model 1. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics (ACL'04)*, Barcelona, Spain, 2004.

Dragos Stefan Munteanu and Daniel Marcu. Improving machine translation performance by exploiting non-parallel corpora. *Computational Linguistics*, 31(4):477–504, 2005.

Dragos Stefan Munteanu and Daniel Marcu. Extracting parallel sub-sentential fragments from non-parallel corpora. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics (COLING-ACL'06)*, Sydney, Australia, 2006.

Jian-Yun Nie. *Cross-Language Information Retrieval*, volume 3:1. Morgan & Claypool Publishers, 2010.

Vassilina Nikoulina, Bogomil Kovachev, Nikolaos Lagos, and Christof Monz. Adaptation of statistical machine translation model for cross-lingual information retrieval in a service context. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL'12)*, Avignon, France, 2012.

Eric W. Noreen. *Computer Intensive Methods for Testing Hypotheses. An Introduction.* Wiley-Interscience, New York, New York, 1st edition, 1989.

Franz Josef Och. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics (ACL'03)*, pages 160–167, Sapporo, Japan, 2003.

Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51, 2003.

Franz Josef Och and Hermann Ney. The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4):417–449, 2004.

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 on Association for Computational Linguistics (ACL'02)*, Philadelphia, Pennsylvania, 2002.

Jay M. Ponte and Bruce Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'98)*, Melbourne, Australia, 1998.

M. F. Porter. An algorithm for suffix stripping. In Karen Sparck-Jones and Peter Willett, editors, *Readings in Information Retrieval*, pages 313–316. Morgan Kaufmann Publishers Inc., San Francisco, California, 1997.

Yonggang Qiu and Hans-Peter Frei. Concept based query expansion. In *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'93)*, Pittsburgh, Pennsylvania, 1993.

Stefan Riezler and Yi Liu. Query rewriting using monolingual statistical machine translation. *Computational Linguistics*, 36(3):569–582, 2010.

Stefan Riezler and John Maxwell. On some pitfalls in automatic evaluation and significance testing for MT. In *Proceedings of the ACL-05 Workshop on Intrinsic and Extrinsic Evaluation Measures for MT and/or Summarization (ACL'05)*, Ann Arbor, Michigan, 2005.

C. J. Van Rijsbergen. *Information Retrieval.* Butterworth-Heinemann, Newton, Massachusetts, 2nd edition, 1979.

Stephen Robertson. How okapi came to trec. *TREC - Experiment and Evaluation in Information Retrieval Voorhees and Harman (2005)*, pages 287–299, 2005.

Stephen Robertson and Karen Sparck-Jones. Relevance weighting of search terms. *Journal of the American Society for Information science*, 27(3):129–146, 1976.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009.

Stephen Robertson, Steve Walker, and Micheline Hancock-Beaulieu. Okapi at TREC-7. In *Proceedings of the Seventh Text REtrieval Conference (TREC'98)*, Gaithersburg, Maryland, 1998.

Shigehiko Schamoni, Felix Hieber, Artem Sokolov, and Stefan Riezler. Learning translational and knowledge-based similarities from relevance rankings for cross-language retrieval. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL'14)*, Baltimore, USA, 2014.

Jason R. Smith, Herve Saint-Amand, Magdalena Plamada, Philipp Koehn, Chris Callison-Burch, and Adam Lopez. Dirt cheap web-scale parallel text from the Common Crawl. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL'13)*, Sofia, Bulgaria, 2013.

Mark D. Smucker, James Allan, and Ben Carterette. A comparison of statistical significance tests for information retrieval evaluation. In *Proceedings of the 16th ACM conference on Conference on Information and Knowledge Management (CIKM'07)*, New York, New York, 2007.

Matthew Snover, Bonnie Dorr, and Richard Schwartz. Language and translation model adaptation using comparable corpora. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'08)*, Honolulu, Hawaii, 2008.

Artem Sokolov, Laura Jehl, Felix Hieber, and Stefan Riezler. Boosting cross-language retrieval by learning bilingual phrase associations from relevance rankings. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'13)*, Seattle, WA, 2013.

Artem Sokolov, Felix Hieber, and Stefan Riezler. Learning to translate queries for clir. In *Proceedings of the 37th Annual ACM SIGIR Conference (SIGIR'14)*, Gold Coast, Australia, 2014a.

Artem Sokolov, Guillaume Wisniewski, and François Yvon. Lattice BLEU oracles in machine translation. *ACM Transactions on Speech and Language Processing (TSLP'14)*, 10(4), 2014b.

Karen Sparck-Jones. A statistical interpretation of term specificity and its application in retrieval. In Peter Willett, editor, *Document Retrieval Systems*, pages 132–142. Taylor Graham Publishing, London, UK, 1988.

Andreas Stolcke. SRILM - an extensible language modeling toolkit. In *Proceedings of the International Conference on Spoken Language Processing*, Denver, Colorado, 2002.

Krysta M. Svore and Christopher Burges. A machine learning approach for improved bm25 retrieval. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM'09)*, Hong Kong, China, 2009.

Don R. Swanson. Historical note: Information retrieval and the future of an illusion. *Journal of the American Society for Information Science*, 39(2):92–98, 1988.

Tuomas Talvensaari, Jorma Laurikkala, Kalervo Järvelin, Martti Juhola, and Heikki Keskustalo. Creating and exploiting a comparable corpus in cross-language information retrieval. *ACM Transactions on Information Systems (TOIS'07)*, 25(1), 2007.

Michael Taylor, Hugo Zaragoza, Nick Craswell, Stephen Robertson, and Christopher Burges. Optimisation methods for ranking functions with multiple parameters. In *Proceedings of the 15th ACM International Conference on Information and Knowledge Management (CIKM'06)*, Arlington, Virginia, USA, 2006.

Ioannis Tsochantaridis, Thorsten Joachims, Thomas Hofmann, and Yasemin Altun. Large margin methods for structured and interdependent output variables. *The Journal of Machine Learning Research*, 6:1453–1484, 2005.

Ferhan Ture and Jimmy Lin. Why not grab a free lunch? mining large corpora for parallel sentences to improve translation modeling. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technologies (NAACL-HLT'12)*, Montreal, Canada, 2012.

Ferhan Ture, Jimmy Lin, and Douglas W. Oard. Combining statistical translation techniques for cross-language information retrieval. In *Proceedings of the International Conference on Computational Linguistics (COLING'12)*, Mumbai, India, 2012a.

Ferhan Ture, Jimmy Lin, and Douglas W. Oard. Looking inside the box: Context-sensitive translation for cross-language information retrieval. In *Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'12)*, Portland, Oregon, 2012b.

Nicola Ueffing, Gholamreza Haffari, and Anoop Sarkar. Transductive learning for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL'07)*, Prague, Czech Republic, 2007.

Jakob Uszkoreit, Jay M. Ponte, Ashok C. Popat, and Moshe Dubiner. Large scale parallel document mining for machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING'10)*, Beijing, China, 2010.

Masao Utiyama and Hitoshi Isahara. A Japanese-English patent parallel corpus. In *Proceedings of the 11th Machine Translation Summit (MTS'07)*, Copenhagen, Denmark, 2007.

Ashish Vaswani, Liang Huang, and David Chiang. Smaller alignment models for better translations: Unsupervised word alignment with the l0-norm. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL'12)*, Jeju Island, Korea, 2012.

Stephan Vogel, Hermann Ney, and Christoph Tillmann. Hmm-based word alignment in statistical translation. In *Proceedings of the 16th International Conference on Computational Linguistics (COLING'96)*, Copenhagen, Denmark, 1996.

Ellen M. Voorhees. Query expansion using lexical-semantic relations. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'94)*, Dublin, Ireland, 1994.

Dekai Wu and Pascale Fung. Semantic roles for SMT: A hybrid two-pass model. In *Proceedings of the 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technologies (NAACL-HLT'09)*, Boulder, Colorado, 2009.

Jinxi Xu and W. Bruce Croft. Query expansion using local and global document analysis. In *Proceedings of the 19th Annual International ACM SIGIR Confer-*

ence on Research and Development in Information Retrieval (SIGIR'96), Zurich, Switzerland, 1996.

Jinxi Xu, Ralph Weischedel, and Chanh Nguyen. Evaluating a probabilistic model for cross-lingual information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR'01)*, New Orleans, Louisiana, 2001.

Xiaobing Xue, Jiwoon Jeon, and Bruce Croft. Retrieval models for question and answer archives. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'08)*, Singapore, 2008.

Omar F. Zaidan and Chris Callison-Burch. Feasibility of human-in-the-loop minimum error rate training. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP'09)*, Singapore, 2009.

Hugo Zaragoza, Nick Craswell, Michael J Taylor, Suchi Saria, and Stephen Robertson. Microsoft cambridge at trec 13: Web and hard tracks. *TREC*, 4, 2004.

Matthew D. Zeiler. ADADELTA: An adaptive learning rate method. *Computing Research Repository (CoRR'2012)*, abs/1212.5701, 2012.

Chengxiang Zhai and John Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'01)*, New Orleans, Louisiana, USA, 2001.

Bing Zhao, Matthias Eck, and Stephan Vogel. Language model adaptation for statistical machine translation with structured query models. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING'04)*, Geneva, Switzerland, 2004.

Justin Zobel and Alistair Moffat. Exploring the similarity space. *SIGIR Forum*, 32 (1):18–34, 1998.

Justin Zobel and Alistair Moffat. Inverted files for text search engines. *Computing Surveys (CSUR)*, 38(2), 2006.

Acknowledgments

First and foremost I want to thank my advisor Prof. Stefan Riezler. It has been an honor to be one of his first Ph.D. students. I am very grateful for him giving me the freedom to pursue research independently, but always being available in case of daunting questions. I would also like to give him credit for convincing me to pursue a Ph.D. in Computational Linguistics.

Furthermore, I would like to thank PD Kurt Eberle not only for agreeing to act as my second advisor, but also for lecturing the Introduction to Computational Linguistics in my first semester at Heidelberg University.

Special thanks goes to my co-authors, but also to all my other colleagues, for providing such a friendly and productive work environment.

Lastly, I am deeply grateful for the continuous and ongoing support of my family and friends in everything I do.

Felix Hieber
Heidelberg, December 13th 2014