A Graph-Based Approach for the Summarization of Scientific Articles

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

vorgelegt
von

Daraksha Parveen
aus India

Referent: Prof. Dr. Michael Strube
Korreferent: Prof. Dr. Katja Markert
Einreichung:

Abstract

Automatic text summarization is one of the eminent applications in the field of Natural Language Processing. Text summarization is the process of generating a gist from text documents. The task is to produce a summary which contains important, diverse and coherent information, i.e., a summary should be self-contained. The approaches for text summarization are conventionally extractive. The extractive approaches select a subset of sentences from an input document for a summary. In this thesis, we introduce a novel graph-based extractive summarization approach.

With the progressive advancement of research in the various fields of science, the summarization of scientific articles has become an essential requirement for researchers. This is our prime motivation in selecting scientific articles as our dataset. This newly formed dataset contains scientific articles from the *PLOS Medicine* journal, which is a high impact journal in the field of biomedicine.

The summarization of scientific articles is a single-document summarization task. It is a complex task due to various reasons, one of it being, the important information in the scientific article is scattered all over it and another reason being, scientific articles contain numerous redundant information. In our approach, we deal with the three important factors of summarization: importance, non-redundancy and coherence. To deal with these factors, we use graphs as they solve data sparsity problems and are computationally less complex.

We employ bipartite graphical representation for the summarization task, exclusively. We represent input documents through a bipartite graph that consists of sentence nodes and entity nodes. This bipartite graph representation contains entity transition information which is beneficial for selecting the relevant sentences for a summary. We use a graph-based ranking algorithm to rank the sentences in a document. The ranks are considered as relevance scores of the sentences which are further used in our approach.

Scientific articles contain reasonable amount of redundant information, for example, *Introduction* and *Methodology* sections contain similar information regarding the motivation and approach. In our approach, we ensure that the summary contains sentences which are non-redundant.

Though the summary should contain important and non-redundant information of the input document, its sentences should be connected to one another such that it becomes coherent, understandable and simple to read. If we do not ensure that a summary is coherent, its sentences may not be properly connected. This leads to an obscure summary. Until now, only few summarization approaches take care of coherence. In our approach, we take care of coherence in two different ways: by using the graph measure and by using the structural information. We employ outdegree as the graph measure and coherence patterns for the structural information, in our approach.

We use integer programming as an optimization technique, to select the best subset of sentences for a summary. The sentences are selected on the basis of relevance, diversity and coherence measure. The computation of these measures is tightly integrated and taken care of simultaneously.

We use human judgements to evaluate coherence of summaries. We compare ROUGE scores and human judgements of different systems on the *PLOS Medicine* dataset. Our approach performs considerably better than other systems on this dataset. Also, we apply our approach on the standard DUC 2002 dataset to compare the results with the recent state-of-the-art systems. The results show that our graph-based approach outperforms other systems on DUC 2002. In conclusion, our approach is robust, i.e., it works on both scientific and news articles. Our approach has the further advantage of being semi-supervised.

Zusammenfassung

Automatische Textzusammenfassung ist eine der bedeutendsten Anwendungen auf dem Gebiet des Natural Language Processing. Textzusammenfassung ist der Prozess der Erzeugung eines Kerns aus Textdokumenten. Die Aufgabe besteht darin, eine Zusammenfassung zu erzeugen, die wichtige, vielfältige und kohärente Informationen enthält, d.h. eine Zusammenfassung sollte in sich geschlossen sein. Die Ansätze zur Textzusammenfassung sind herkömmlich extraktiv. Die extraktiven Ansätze wählen eine Teilmenge von Sätzen aus einem Eingabedokument für eine Zusammenfassung aus. In dieser Arbeit stellen wir einen neuartigen grafischen Extraktionsansatz vor. Mit der fortschreitenden Förderung der Forschung in den verschiedenen Bereichen der Wissenschaft ist die Zusammenfassung von wissenschaftlichen Artikeln, eine wesentliche Voraussetzung für die Forscher geworden. Dies ist unsere primäre Motivation bei der Auswahl wissenschaftlicher Artikel als unseren Datensatz. Dieser neu gegründete Datensatz enthält wissenschaftliche Artikel aus der Zeitschrift PLOS Medicine, einem hochwissenschaftlichen Journal auf dem Gebiet der Biomedizin.

Die Zusammenfassung von wissenschaftlichen Artikeln ist eine Einzeldokumentation. Es ist eine komplexe Aufgabe aus verschiedenen Gründen, einer davon ist, dass die wichtige Information in dem wissenschaftlichen Artikel überall verstreut ist und ein weiterer Grund ist, dass wissenschaftliche Artikel zahlreiche redundante Informationen enthalten. In unserem Ansatz befassen wir uns mit den drei wichtigen Faktoren: Wichtigkeit, Nicht-Redundanz und Kohärenz. Dazu verwenden wir Graphen, da sie Datenprobleme lösen und rechnerisch weniger komplex sind.

Wir verwenden exclusiv eine bipartite grafische Darstellung für die Verdichtungsaufgabe. Wir repräsentieren Eingangsdokumente durch einen bipartiten Graphen, der aus Satzknoten und Entitätsknoten besteht. Diese bipartite Graphdarstellung enthält Entitätsübergangsinformationen, die für die Auswahl der relevanten Sätze für eine Zusammenfassung vorteilhaft sind. Wir verwenden einen grafischen Klassifizierungsalgorithmus, um die Sätze in einem Dokument zu ordnen. Die Reihen werden als Relevanz-Scores der Sätze betrachtet, die in unserem

Ansatz weiter verwendet werden. Wissenschaftliche Artikel enthalten eine beträchtliche Menge an redundanten Informationen, so enthalten z.B. die Abschnitte Einleitung und Methodik ähnliche Informationen über die Motivation und den Ansatz. In unserem Ansatz stellen wir sicher, dass die Zusammenfassung Sätze enthält, die nicht-redundant sind. Obwohl die Zusammenfassung wichtige und unredundante Informationen des Input-Dokuments enthalten sollte, sollten die Sätze so miteinander verbunden sein, dass es kohärent, verständlich und einfach zu lesen ist. Wenn wir nicht sicherstellen, dass eine Zusammenfassung kohärent ist, könnten die Sätze nicht korrekt verbunden sein. Dies führt zu einer undurchsichtigen Zusammenfassung. Bisher kümmern sich nur wenige Verdichtungsansätze um Kohärenz.

In unserem Ansatz kümmern wir uns um die Kohärenz auf zwei verschiedene Arten: durch Verwendung der graphischen Maßnahme und durch Verwendung der strukturellen Informationen. Wir verwenden outdegere als Diagrammmaß und Kohärenzmuster für die Strukturinformation in unserem Ansatz. Wir verwenden Integer-Programmierung als Optimierungstechnik, um die beste Teilmenge von Sätzen für eine Zusammenfassung auszuwählen. Die Sätze werden auf Basis von Relevanz, Diversität und Kohärenzmaß ausgewählt. Bei der Berechnung dieser Maßnahmen wird auf enge Integration und Gleichzeitigkeit geachtet. Wir verwenden menschliche Bewertungen, um die Kohärenz der Zusammenfassungen zu beurteilen/evaluieren. Wir vergleichen ROUGE Scores und menschliche Bewertungen verschiedener Systeme auf dem PLOS Medicine Datensatz. Unser Ansatz ist wesentlich besser als andere Systeme dieses Datensatzes. Darüber hinaus wenden wir unseren Ansatz auf den Standard DUC 2002-Datensatz an, um die Ergebnisse mit den jüngsten state-of-the-art-Systemen zu vergleichen. Die Ergebnisse zeigen, dass unser grafischer Ansatz andere Systeme auf DUC 2002 übertrifft. Zusammenfassend ist unser Ansatz robust, d.h. er funktioniert sowohl bei Artikeln auf wissenschaftlicher Ebene als auch bei Zeitungsartikeln. Unser Ansatz hat den weiteren Vorteil, dass er halbüberwacht ist.

Acknowledgments

First, I would like to thank my advisor Prof. Dr. Michael Strube. His persistent support made my PhD possible. I really appreciate his patience, listening to my ideas; and correcting the draft version of my papers. He even went to the extent of making me learn using articles in the English language. He has been a very supportive and helpful supervisor. I could not have imagined, a better supervisor for my thesis.

I would also like to thank my colleagues, Nafise Moosavi, Mohsen Mesgar, Angela Fahrni, Sebastian Martschat, Alex Judea, Yufang Hou, and Mark-Christopher Muller, who were also human judges in one of my experiments. Moreover, they helped me in correcting the thesis by giving their useful comments. I am highly obliged for this.

I would again like to thank my colleague, Nafise Moosavi, for the tea breaks we had, whenever she ran her feature mining algorithm on the stanford system. Her love and care turned even my worst days cheerful.

I would like to thank my colleague, Mohsen Mesgar, for the coherence patterns he built which later I used in my approach. Working with him made me learn, teamwork and cooperation.

I would like to thank my family and friends for always being supportive. My parents, Mr. Jabbar Ali Ansari and Mrs. Shahista Parveen Ansari, support has been unrelenting force which motivated me to stand where I do now. Their simple advice could solve even the most complicated problems of my life. I thank my husband, Squadron Leader Sameer Tahir, for being right at my side in all my ups and downs. His endless optimism and patience gave me the strength to leap over all my hurdles. He has been constantly, encouraging and inspiring me throughout the research. I would like to thank my sister, Alisha Parveen, for being always present; physically being in Heidelberg and mentally, for being at the forefront of my joy and my distress. I hope she has learned from my mistakes and submit her own thesis with flying colors. I also would like to thank my uncle, Mr. Mohd. Afroz, for his inception to convince my parents for starting my PhD from Heidelberg, Germany. I really appreciate the support I got during my research from my

extended family, Mr. Tahir Iqbal and Mrs. Rana Tahir, who patiently listened to rantings of my problems without flinching.

Lastly, I would like to thank HITS, Heidelberg for the PhD scholarship and AIPHES, Darmstadt for travel grants.

Contents

Chapter 1

Introduction

The topic of this thesis is a graph-based approach for the summarization of scientific articles. Summarization is a task that has drawn the attention of researchers in natural language processing and recently also in artificial intelligence and information retrieval. In automatic text summarization, we give text documents as an input and obtain a coherent gist as an output, also referred to as a summary.

Summaries should consist of relevant but non-redundant information from the input text documents. Moreover, summaries should be readable to the users, hence they should be coherent and grammatically correct. The main goal of various summarization approaches is to extract important and non-redundant information from the input document, for instance, maximal marginal relevance (Carbonell & Goldstein, 1998) based approaches (Chapter 7).

We can broadly categorize summarization approaches as: extractive summarization and abstractive summarization. In case of an extractive summarization, grammaticality is not a concern as complete sentences are extracted from text documents. In contrast, an abstractive summary contains the same information as an input text but may not consists of same sentences as the input text. In this thesis, we mainly focus on the extractive summarization approach.

In this thesis, we represent text documents graphically (Chapter 5), where the graphs are bipartite graphs based on the entity grid (Barzilay & Lapata, 2008). However, to our knowledge, the entity grid has not been used directly in extractive summarization to ensure the coherence of a summary. In our summarization approach, the three factors; importance, non-redundancy and coherence, are incorporated in a principled way. Our approach has the further benefit of being completely robust and scalable, which is shown in Chapter 6.

We apply our graph-based approach to scientific articles from the *PLOS Medicine* journal and the standard dataset for single-document summarization (DUC 2002). We discuss in detail about this journal and DUC 2002 in Chapter 2.

In this chapter, we will give the general overview of summarization and scientific articles.

In the end of this chapter, we will discuss the research questions dealt in this thesis and the research contributions. We then outline the thesis structure (Section 1.7) and indicate our publications related to this thesis (Section 1.8).

1.1 Automatic Summarization

The exponential growth of World Wide Web (WWW) has resulted in a well-known problem of information overload. A large amount of information available online has overwhelmed the user and rendered the information useless, when needed in a short amount of time. For instance, in the field of medical science, the researchers have to make some important decisions by considering all the information available to them. To enable a user to have an understanding of important information out of the large expanse of data available online; the automatic summarization technique is pre-eminent.

Automatic summarization is a technique which takes large documents as an input, extracts the most important content from it, and then output the important content in a condensed form to the user. In this section, we introduce the basic idea of automatic text summarization.

The Automatic summarization technique simplifies the tedious task of manually summarizing various texts to gather meaningful information. This technique is useful in various inevitable information sources such as:

- A Newspaper Headline
- A Handheld PC
- A Movie Review
- An Abstract of a Scientific Article
- A Table of Contents
- A catalog of Products
- A Medical record

The applications of the automatic summarization technique can be found in nearly every written medium (Mani & Maybury, 1999).

In the above examples, it is evident that the output summaries can be in different formats like videos, texts or pictures. Similarly, the input data to the summarizer can also be in diverse formats. It is also possible that the source format is different from the output format. For example, a meeting summarizer gives condensed information about the meeting to a user, it captures audio recorded voice during meeting and gives summarized text output. A summary

necessarily may not contain the exact texts from the source, for example, "an abstract of a scientific article".

The different variants of text summaries depend on the methodology of condensation of textual information from the source, as per the user's requirement. In other words, with the same input source, we can have different summaries based on the requirement of the reader or the application. For example, a researcher while traveling would prefer to read the condensed form of an article on their Personal Digital Assistant (PDA), so that they can decide whether that article is of their interest. In the medical domain, for a long term treatment a doctor would prefer to study a patient's medical history in detailed version without losing any meaningful information to decrease the risk factor, however, he would want to read a summarized history of a patient who is in emergency, so that he can treat him as soon as possible by knowing his important details.

Depending on the requirement of applications, the length of a summary (number of words/ number of sentences) is determined. Fundamentally, summary's length should range from just shorter than the input document to something more than 0% of the input document, which defines the compression rate. For example, a news headline of a long news story only contains a few words, which is approximately 99% compression rate. However, an abstract of scientific articles has important information which amounts to a compression rate of approximately 10%.

The above description of text summarization shows that there are some research areas which can potentially improve the quality of output summaries, if included in the automatic text summarization technique. For example, information extraction is helpful in extracting important information for a summary. Similarly, question answering system can be used as one of the components of query-based summarization, where user gives a query, based on which, the summarizer produces a summary.

In the next section, we will discuss different parameters for the automatic text summarization task.

1.2 Types of Summarization

Automatic summarization varies in respect of output summaries and source documents. A basic distinction in automatic text summarization is determined by various different parameters as shown in Figure 1.1 (Mani, 2001). These parameters are distinguished not only by the type of output summaries but also by the type of source documents:

- **Relation to Source**: This parameter defines the type of units which can be included in a summary. A summary which is composed of extracts having exact sentences of a source document (Figure 1.2, i) is known as an extractive summary (Figure 1.2, ii). In contrast,

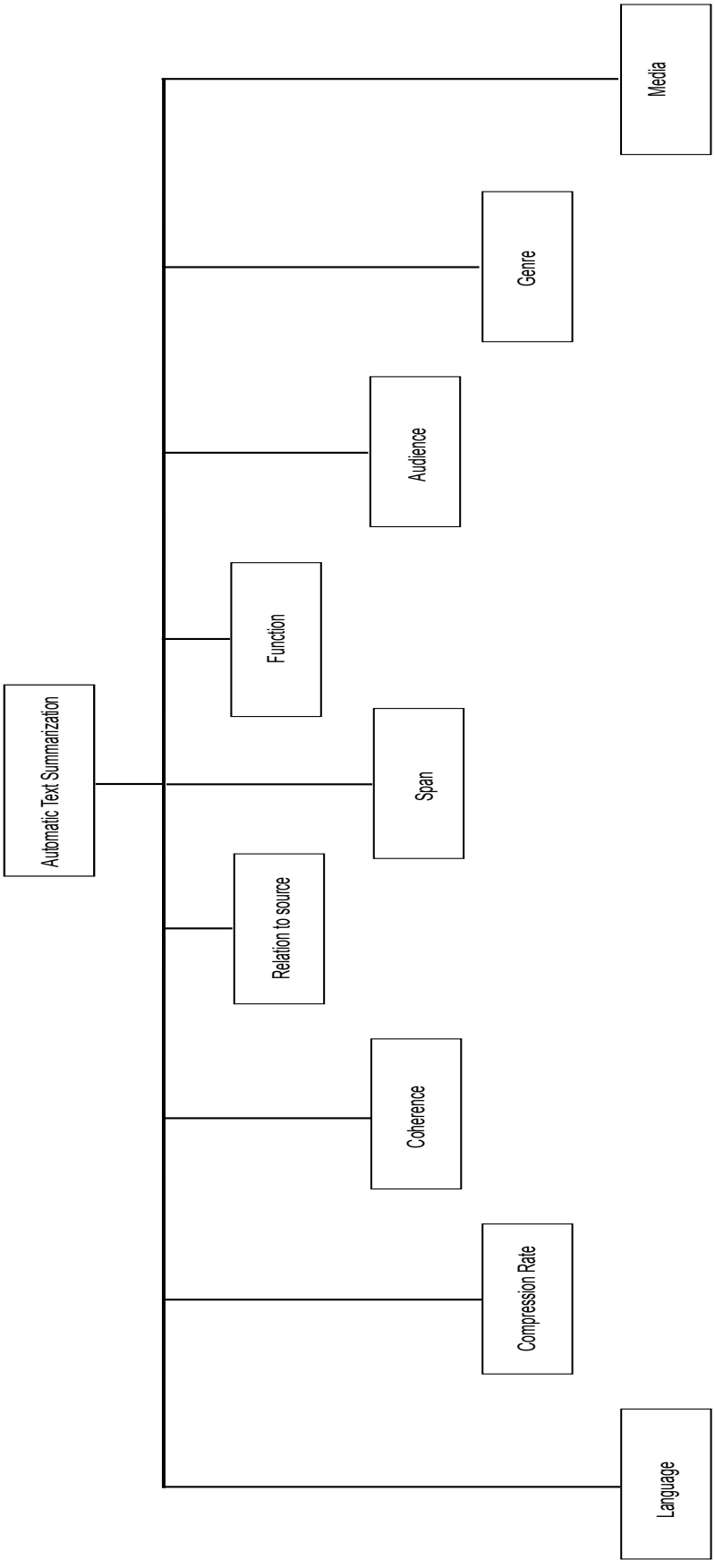


Figure 1.1: The parameteres for summarization.

if a summary gives the same information as the input document (Figure 1.2, i) but is not using the exact sentences is known as an abstractive summary (Figure 1.2, iii).

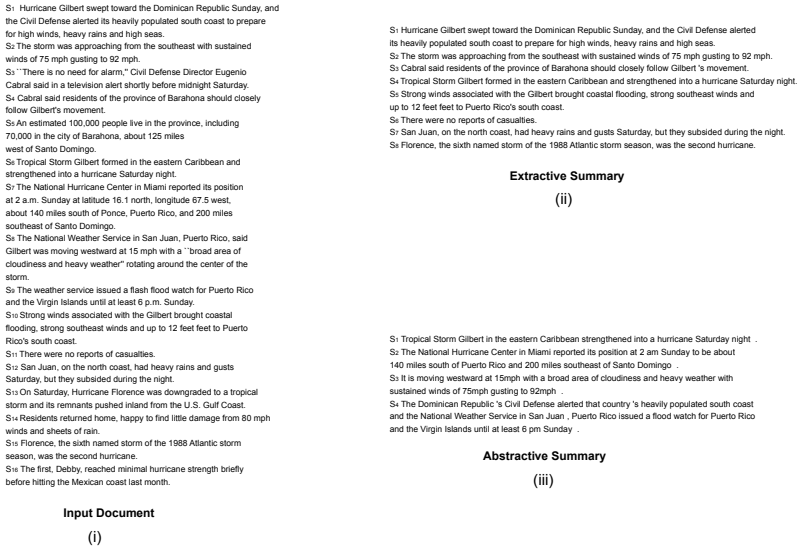


Figure 1.2: An example of abstractive and extractive summarization.

- **Function**: This is one of the traditional distinctions for the output summary (Borko & Bernier, 1975). In this parameter summary can be indicative, informative or critical depending on its usage. The relationship between these three types of summaries is shown in Figure 1.3. An indicative summary may not contain any important information from the input document but is rather used to convince the user to further read the corresponding documents, for example, a review at the back of a book. An informative summary consists of useful and important information from source documents, for example, an abstract of a scientific article. A critical summary expresses the summarizer's views on the quality of work in the source article, for example, a movie review is written by a critic who summarizes the story of the movie in a commentative way.

- **Coherence**: A coherent summary is easy to read and understand (Figure 1.4). In coherent text sentences are connected with each other via some relations, for instance, lexical or semantic relations (Mesgar & Strube, 2015). In contrast, an incoherent summary may not contain any ordered or connected sentences as shown in Figure 1.4. The reason for this can be unresolved anaphors, repetitive sentences, and badly organized sentences, etc (Mani, 2001).

- **Span**: This parameter quantifies the source documents to be summarized. The source may contain single document, i.e., single document summarization (Figure 1.5, i) or multiple documents, i.e., multi-document summarization (Figure 1.5, ii).

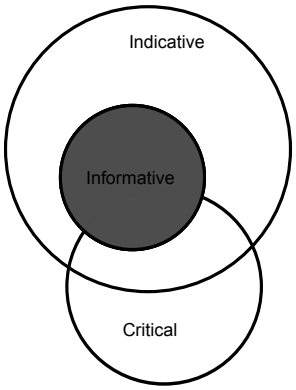


Figure 1.3: Relationship between indicative, informative and crtitical summaries (Mani, 2001).

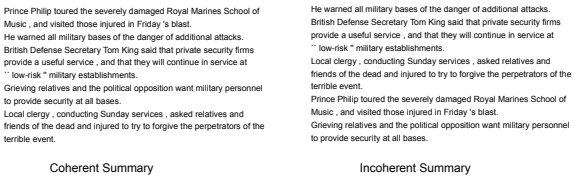


Figure 1.4: An example of a coherent and an incoherent summary.

- **Purpose**: This parameter differentiates between query-based summaries or generic summaries. In query-based summarization, a query is given by the user, and the output summary contains information relevant to the query. In generic summarization, the summary contains important information from the source document without any specific demand by the user.

- **Compression Rate**: This parameter gives us the information about the length of a summary with respect to the input document. The compression rate is determined by the application of the summary. For instance, for a headline of a news, a summarizer may need a compression rate of 10%, whereas for an abstract of a scientific article requires 15% compression rate.

- **Genre**: A summarizer uses different approaches for different genres. For example, in the summarization of news articles, top few sentences are commonly considered as good candidates for a final summary whereas this approach is not applicable in the summarization of scientific articles.

- **Language**: A summarizer can be monolingual or multilingual. The monolingual summarizer uses only one language, and produces an output summary in the same language as the input document whereas, the multilingual summarizer uses multiple languages,

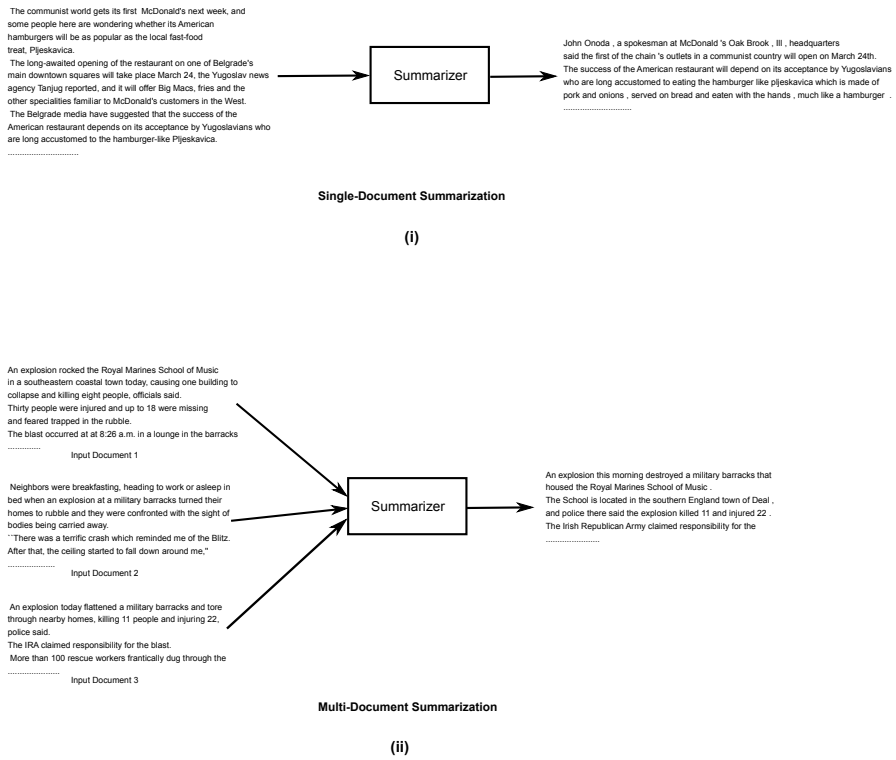


Figure 1.5: An example of multi-document and single-document summarization.

and gives an output summary in one of the languages from the input document.

- **Media**: Multimedia summarization consists of various media formats as an input and produces output summary in one of the media types.

1.3 General Architecture for Extractive Summarization

In this section, we will discuss the general architecture of summarization which can be followed by any approach. The general architecture for summarization is shown in figure 1.6. This is an abstract architecture in which each block can be fulfilled in numerous ways. This architecture first takes source documents as an input and then processes them in the document representation block. In the document representation block, the representation of documents takes place after preprocessing the documents. The representation can be achieved by various methods, for example, documents can be represented by calculating the frequency of words in the documents (term frequency method) (Luhn, 1958) or by representing the documents graphically (Mihalcea & Tarau, 2004; Erkan & Radev, 2004; Parveen & Strube, 2014). The document representation is then used in calculating the importance of the unit in the analysis phase, for example, using some statistical measure by considering the term frequency

or graph importance measure such as betweenness, centrality etc. Here, units are referred to as sentences, as we only focus on extractive summarization in this thesis. The selection of sentences is then achieved on the basis of the importance measure, i.e., units with a high importance measure leads to the next phase. In the final phase of summary generation, the candidates from the earlier phase are included in the summary on the basis of compression rate and other factors if applicable.

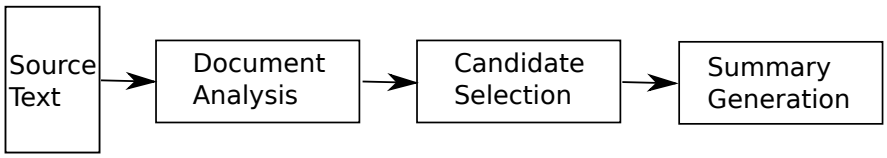


Figure 1.6: A general architecture for the summarization approach.

1.4 Scientific Articles

There are various types of text documents like scientific articles, news articles, reviews, records, etc. We have chosen the dataset consisting of scientific articles in this thesis. There are several reasons for selecting this dataset in our work:

- Researchers, in any domain, are referring to scientific articles, books, or web sources as a source of information to facilitate their research (Mani, 2001).
- There is a dire requirement of summaries of scientific articles in digital repositories for researchers, to facilitate quick access of necessary information (Mani, 2001).

There are various factors which makes scientific articles distinct from general texts (Teufel & Moens, 2002):

- **Time**: In news articles, an occurrence of some incident is reported in chronological order. The chronological order of news articles is important because it makes their summarization fairly easy. A summarizer exploits the chronological property of news articles by considering recently occurred incidents more important than the previous incidents. In contrast, scientific articles do not follow any time dependent events. Instead, they report the intellectual work done by researchers within a certain time frame. However, in scientific articles, *related work* section may consist chronological order of discussions about the previous works done by researchers.

- **Writing Style**:Scientific articles have a diverse writing style which may vary significantly as per the requirements. For example, a review article contains the review of various research works in a specific field, whereas some articles argue with the results of the state-of-the-art in their field. Some articles only discuss about the implementation of a tool developed by the author, others combine various methods from different fields into interdisciplinary fields, for instance, computational analysis of a text is a well known interdisciplinary field, i.e., computational linguistics.

- **Bias**: Scientific articles are biased by authors because they illustrate the research from their own point of view. They are written by researchers for recognition in the community. In contrary to this, editors of news articles write reports without being biased, i.e., their reports are generally neutral. This bias factor of scientific articles can be exploited in summarization techniques.

Scientific articles do not follow any specific pattern even though they are written by the same author. It is difficult to analyze scientific articles when compared with general texts. Summarization of scientific articles is considerably difficult because scientific articles do not have any specific document structure.

This thesis is specific to scientific articles of biomedical domain because biomedical scientific articles are from diverse interdisciplinary fields. Hence, this motivates us to develop a domain independent approach for summarizing scientific articles.

1.5 Research Goals

Automatic text summarization is a task to reduce a text document automatically in order to produce a summary which contains important information from the document. To accomplish this task, many researchers use various techniques of natural language processing, machine learning, data mining and information retrieval. These techniques can be applied in various manner to obtain the summary of an input document.

In this thesis, we are focusing on the summarization of scientific articles. We mainly analyse (1) the different document representations of scientific articles (2) the important information in scientific articles (3) the non-redundant information in scientific articles (4) the coherence of a summary (5) robustness and scalability across different domains of text documents. We explain in detail these research questions in the following sections.

1.5.1 Representation of Documents

Various representations of text documents have been explored in the field of natural language processing, for instance, the vector space model (Luhn, 1958) or the graphical model (Mi-

halcea & Tarau, 2004) for representing a text document. There are several approaches for summarization that use the graphical representation of a text document, such as Mihalcea & Tarau (2004) introduce a graphical representation of a text document, where nodes are sentences and then apply the *PageRank* algorithm (Brin & Page, 1998) on the graph to extract important sentences from the document. This type of graphical representation connects the two sentences on the basis of their similarity; however, this graph only contains bag-of-word information (Harris, 1954) that is not sufficient for the summarization task. In this task, understanding of a text is very important. Consequently, the document representation should be informative to comprehend the text.

Question 1. What type of graphical representation can be used to produce informative summaries?

1.5.2 Important Information

In the summarization task, a summary of a document must contain important information. This aspect of summarization is taken care of by many approaches. Some approaches calculate the importance of a sentence in a supervised manner, such as, Liakata et al. (2013) use conditional random fields to obtain important sentences, whereas other approaches give the importance score to a sentence in an unsupervised manner (Luhn, 1958; Edmundson, 1969).

In summarization, obtaining the training dataset is a demanding task. There are some recent approaches which attempt to create the training data. However, the scope of these approaches is very limited (Contractor et al., 2012). Thus, an unsupervised way to obtain important information is more suitable for summarization.

Question 2. How to deal with the importance aspect of the summarization task in an unsupervised manner?

1.5.3 Non-redundant Information

Scientific articles contain plenty of redundant information, such as, *Introduction* and *Conclusion* sections contain a lot of similar information. There are few approaches which consider non-redundancy while summarizing scientific articles. Usually, these approaches deal with non-redundancy using word overlapping among sentences. However, these approaches do not consider two sentences redundant if they share same information but different words.

Question 3. How to extract the non-redundant information while summarizing scientific articles?

1.5.4 Coherent Information

A summary of a document must contain coherent information, i.e., it should be readable for users. Only few works have considered coherence while summarizing scientific articles, such as, Abu-Jbara & Radev (2011) introduce an approach which refine the summary sentences to improve readability. However, none of them deal with the problem of coherence within the task of sentence selection. The selection of sentences and the assurance of coherent summaries are not tightly integrated in their techniques to obtain best possible set of sentences which are important, non-redundant and coherent. Moreover, they model coherence in summarization by only considering adjacent sentences which may not be coherent. It is possible that some sentences that are few sentences apart are more coherent as compared to the adjacent sentences.

There are few methods (Hirao et al., 2013; Gorinski & Lapata, 2015) which integrate coherence concisely. These methods do not take into account the overall structure of the summary, however.

Question 3. How to incorporate coherence aspect while extracting the important and non-redundant information for the summary?

1.5.5 Robustness and Scalability across Different Domains

Now a days, information in different fields are increasing exponentially due to which summarization is required in every field. If we develop a summarization approach which is extremely domain dependent, then we need to build numerous summarization approaches. Thus, a summarization approach should be less domain dependent so that it is easy to switch from one domain to another domain with few minor changes in the approach.

Until now, most of the approaches are not scalable and robust. They do not perform effectively if the size or the domain of the input document varies. For instance, *Mead* is a summarizer, which performs substantially well with news articles, as it considers the property of news articles [1] while summarizing them (Radev et al., 2004a). However, *Mead* may not perform efficiently with some other domain that does not have the same property as the news articles.

Question 4. How to create a robust and scalable summarizer?

1.6 Research Contributions

The research contributions of this thesis are: a novel summarization approach and the new dataset of scientific articles. We introduce the new dataset in Chapter 2 and the proposed

[1] Sentences in the beginning of the document can be considered as the good candidate for the summary.

approach in Chapter 5.

1.6.1 A New Dataset for Summarization

In this thesis, we introduce a new dataset of scientific articles for the summarization task. This dataset has several advantages over already available datasets (Joseph & Radev, 2007; Teufel & Moens, 2002). Unlike other datasets, we do not consider abstracts as human summaries for evaluation. In our dataset, every scientific article is accompanied with a summary not written by the author. In contrast to abstracts, these summaries have general perspective about scientific articles. We discuss in detail the benefits of using this dataset in Chapter 2.

Contribution 1. We propose a new dataset of scientific articles for the summarization task.

1.6.2 Identifying Relevant Information

We use the graphical representation of documents as they are computationally less expensive as compared to vector space model (Wolfram, 2003). Graphical representation has been used frequently for the summarization task; however, they do not contain sufficient information about the document. Since the approach for summarization must have the deep understanding of the document, we utilize more informative graphical representation (Guinaudeau & Strube, 2013), i.e., a bipartite graph, of the document which consists of discourse entities. We show in the results (Chapter 6) that bipartite graphs provide more relevant summary as compared to the other graph-based approaches. Using the graphical representation of the document, we extract information from a document for the summary which is important but non-redundant. This leads to our research contribution:

Contribution 2. We show that bipartite graphs are more informative and proficient for the summarization task than general graphs and are more capable of obtaining relevant information.

1.6.3 Detecting Coherent Information

Only few approaches for summarization take care of the coherence of a summary. However, these approaches incorporate coherence by considering adjacent sentences, which is not sufficient to generate coherent summaries.

To overcome these problems, we propose a novel graph-based method for creating a coherent summary. Mesgar & Strube (2015) use the frequency of coherence patterns (Daneš, 1974) to rank documents by coherence and readability. Instead of ranking summaries by coherence, we use coherence patterns directly to extract sentences for creating a coherent summary. Thus,

we extract not only important and non-redundant sentences from the input document but also coherent ones. This leads to our third research contribution:

Contribution 3. We introduce a method that is not limited to the adjacent sentences and show that long distance sentences can make better coherent summaries.

1.6.4 Global Optimization

Until now, approaches for summarization do not deal with coherence, importance and non-redundancy simultaneously. In general, these approaches use greedy techniques to select the sentences for a summary, however only few approaches employ the global optimization technique. The advantage of using global optimization is that it attempts to extract the set of sentences which are best among all the other possible set of sentences in the document.

In the proposed approach, we produce globally optimized summaries by maximizing over the factors: relevance, non-redundancy and coherence. Unlike greedy approaches, these factors are tightly integrated in our approach.

Contribution 4. We show that a global optimization approach produces better summaries as compared to greedy approaches.

1.6.5 High Robustness and Scalability

We introduce a new dataset in this thesis which consists of scientific articles. We apply our approach on this dataset and a standard dataset. We show in the results that the proposed approach is largely scalable and robust across domains and it only needs a few modifications. Our proposed approach achieves state-of-the-art results on the datasets from different domains. This indicates that the importance, non-redundancy and coherence measures are domain independent, at least for the domains we take into account.

Contribution 5. Our proposed summarization approach is highly robust across different domains and achieves state-of-the-art results. This approach performs efficiently with short and long documents; hence it is fairly scalable.

1.7 Outline of this Thesis

This thesis consists of eight chapters. We first describe the new scientific articles' dataset introduced by us; we use for the evaluation of the proposed method (Chapter 2). Further, we describe the techniques, which are used in the proposed method: Mixed Integer Programming (Chapter 3) and Latent Dirichlet Allocation (Chapter 4). Afterwards, we explain our approach (Chapter 5) in detail. Then, Chapter 6 gives the detailed description of the setup for the experiments and the results of the proposed approach on different data sets. Further, we discuss

the related work of the summarization task in Chapter 7. Finally, we conclude with some discussions over the proposed method and results and mention the future research directions (Chapter 8). We discuss the content of each chapter in detail as follows:

1. **Introduction:** In this chapter, we introduce summarization broadly. Then, the parameters that distinguishes the automatic text summarization task. Afterwards, we provide the general architecture of summarization. We provide the research questions that are addressed in this thesis and summarize the contribution of this thesis. Moreover, this chapter contains the outline of this thesis and indicate to previously published work.

2. **Dataset:** Many standard datasets are being used for the summarization task. Document Understanding Conferences (DUC) build datasets for the summarization task with different goals, for instance, DUC 2002 is a dataset for generic single-document summarization whereas DUC 2005 is a topic-based multi-document summarization dataset. In this chapter, we introduce a new dataset for summarization (Question 4). This dataset consists of scientific articles from a well-known biomedicine journal. We also give the detailed description of the standard dataset for a generic single-document summarization (Question 4).

3. **Mixed Integer Programming:** In the proposed approach, we utilize mixed integer programming to achieve globally optimized results. In this chapter, we describe in detail various approximation techniques to solve the integer programming problems. We also discuss different types of integer programming problems and their solutions using the approximation techniques. This optimization technique tightly integrates importance, non-redundancy, and coherence in our approach.

4. **Latent Dirichlet Allocation:** We represent the documents graphically in our proposed approach. For that, we utilize latent dirichlet allocation to find the topics in documents. In this chapter, we discuss about the topic models and give the detailed description of latent dirichlet allocation (a fundamental topic model). Then, we discuss about the assumptions in latent dirichlet allocation and drawbacks of these assumptions.

5. **Methodology:** In this chapter, we give the detailed description of our approach. We discuss in detail, how we incorporate importance, non-redundancy and coherence in our method. In this chapter, we address mainly Question 2 and Question 3. We discuss the intuition of our approach for the summarization task.

6. **Results:** In this chapter, we give the details of experimental setup of our approach. We discuss in detail the tools utilized in this approach. We employ the gurobi optimization tool to achieve globally optimized results. Further, we compare the results of different

versions of our approach to some baselines and state-of-the-art approaches. We employ two levels of evaluation: relevance evaluation and coherence evaluation. For the relevance evaluation, ROUGE scores are used to compare the results and for the coherence evaluation, we employ human coherence assessments. We present some analysis to examine the effectiveness of the proposed approach (Question 1-4).

7. **Related Work:** Automatic text summarization is an important topic in various fields, including Artificial Intelligence, Data Mining, Information Extraction, and Information Retrieval. In this chapter, we give the insight of ongoing research on summarization. We begin with the classical approaches and finish with the deep learning approaches for the summarization task. We focus on corpus-based approaches, discourse-based approaches, graph-based approaches, topic modeling-based approaches, integer linear programming-based approaches, and neural network-based approaches.

8. **Conclusions and Future Work:** In this chapter, we summarize the results obtained by the proposed approach and indicate the future research directions.

1.8 Published Work

Most of the contributions and ideas described in this thesis are published in various conferences. We have employed the entity graph representation for multi-document summarization. This work is presented in Parveen & Strube (2014). Our approach which is based on the entity graph and outdegree for summarization along with the new dataset of scientific articles has been published in Parveen & Strube (2015). Our approach based on the weighted topical graph and weighted outdegree for summarizing scientific articles is presented in Parveen et al. (2015). Our approach based on coherence patterns is presented in Parveen et al. (2016). Table 1.1 exhibits the relation between the published work and this thesis.

Published Work	Chapters
Parveen & Strube (2014)	Chapter 7
Parveen & Strube (2015)	Chapter 2, 5, 6
Parveen et al. (2015)	Chapter 5, 6
Parveen et al. (2016)	Chapter 5, 6

Table 1.1: Relation between the thesis and published work

Chapter 2

Dataset

In this thesis, we introduce a new dataset of scientific articles for the task of summarization. This dataset consists of scientific articles from the field of biology and medicine and is derived from the *PLOS Medicine* journal. The journal has a high impact factor in the field of medicine and is openly accessible to everyone. We discuss in Section 1.4, how the genre of scientific articles is different from that of the other genre.

We chose biomedicine as a domain for a number of reasons. One reason is that it is a domain in which numerous researches are being conducted. Hence, this domain has a large number of research articles. This makes the formation of the dataset less difficult. The more notable reason is that the biomedicine domain is a heterogeneous domain as it consists of research articles from different disciplines. Consequently, the structure of documents in this domain varies substantially. Due to which we are forced to select an approach which is domain independent, to summarize scientific articles from the *PLOS Medicine* journal. In short, our dataset consists of complex scientific articles for the summarization task. We discuss the reasons for choosing scientific articles from *PLOS Medicine* in Sections 2.1 and 2.2

Furthermore, we discuss the details of the DUC 2002 dataset. This dataset was introduced by the Document Understanding Conference (DUC) in year 2002. This dataset is a standard dataset for single-document summarization. The articles in DUC 2002 belong to the news genre.

The *PLOS Medicine* journal and dataset has been described in detail in Section 2.1 and 2.2, respectively. The statistics of the DUC 2002 dataset and its comparison with the *PLOS Medicine* dataset is shown in Section 2.3.

November 18, 2014

Computerized Cognitive Training in Cognitively Healthy Older Adults

Michael Valenzuela and colleagues systematically review and meta-analyze the evidence that computerized cognitive training improves cognitive skills in older adults with normal cognition.

Image credit: Saad Faruque, Flickr

PLOS Medicine celebrates 10 years!

In honor of the occasion, the *PLOS Medicine* editors reflect on some of our most interesting and influential articles from the past decade.

Read more

PERSPECTIVE 11/18/2014

What Could Computerized Brain Training Learn from Evidence-Based Medicine?

In a Perspective linked to the study by Michael Valenzuela and colleagues, Druin Burch considers the context, content, and limitations of what we know about computerized cognitive enhancement.

Image credit: Karl-Ludwig Poggemann, Flickr

RECENTLY PUBLISHED ARTICLES

Computerized Cognitive Training in Cognitively Healthy Older Adults: A Systematic Review and Meta-Analysis...

What Could Computerized Brain Training Learn from Evidence-Based Medicine?

Association of FKBP51 with Priming of Autophagy Pathways and Mediation of Antidepressant Treatment Response:...

SEE ALL ARTICLES

BROWSE ISSUES
2004 – 2014

RESEARCH ARTICLE 11/11/2014

A Transdiagnostic Community-Based Mental Health Treatment for Comorbid Disorders

In a randomized controlled trial, Paul Bolton and colleagues investigate whether a transdiagnostic community-based intervention can improve mental health symptoms among Burmese refugees in Thailand.

Image credit: Pete Brown, Flickr

RESEARCH ARTICLE 11/11/2014

The kSORT Assay to Detect Renal Transplant Patients at High Risk for Acute Rejection

Minnie Sarwal and colleagues developed a gene expression assay using peripheral blood samples to identify patients with renal transplant at high risk for acute rejection.

Image credit: Hey Paul Studios, Flickr

2.1 PLOS Medicine

Researchers in the field of medicine desire to publish in a high impact and openly accessible journal like *PLOS Medicine*. This journal issues articles on environmental, biomedical and political determinants of health and society. The journal has been established in 2004, and offers following advantages [1]:

- **Open Access for reading, educating, and applying**

  In PLOS Medicine, research articles are available to everyone at no cost as soon as they are published. There is no subscription or one-time registration cost involved in accessing the publications. These publications are accessible to everyone worldwide, i.e., Educators, students, clinicians, researchers, patients, and policy makers without any barriers of cost or access control. Hence, the published works in this journal can be utilized by anyone to advance their research.

- **Editors' summaries**

  Every month there is one professional editor who explains the context, methods, results and its implications in a comprehensible language so that any amateur can understand the summary. This makes *PLOS Medicine* journal beneficial for students, education of patients, researchers from other fields and non-physicians, and enables them to learn from it.

- **No arbitrary constraints on article length or presentation**

  There is no arbitrary page limit in PLOS Medicine journals which gives the flexibility to authors to explain their research in detail. The research articles, on submission, should explain the methods and the obtained results in detail after performing the experiments explained in the paper.

2.2 Scientific Articles

The dataset being used in this thesis consists of scientific articles from *PLOS Medicine*, which have been created for summarization. The foremost reasons for using the *PLOS Medicine* journal articles as a dataset for automatic text summarization are discussed below:

- **Evaluation**

  In summarization, evaluation of summaries has always been a challenge. The evaluation of summaries is only possible if there are gold summaries or human written summaries

[1] http://journals.plos.org/plosmedicine/s/why-publish-with-plos-medicine

of source documents available. It is arduous, demanding and expensive to create gold summaries. Moreover, it is difficult to achieve high inter-annotator agreement for the gold summaries. Inter-annotator agreement is a measure of how effectively two or more annotators can make the same decision for a certain category.

In PLOS Medicine, every article is accompanied with its editor's summary which can be used as a gold summary. As discussed above, editors' summaries have a very general perspective, which can easily be understood by any non-expert.

In conclusion, the editor's summaries can be considered as high standard gold summaries. We are using abstracts of scientific articles as gold summaries, too. However, our goal is to produce summaries which are comparable to the editors' summaries. In the results, we emphasize more on the scores obtained by our approach using them as gold summaries. An example of this summary is shown in Figure 2.1

Cardiometabolic diseases—cardiovascular diseases that affect the heart and/or the blood vessels and metabolic diseases that affect the cellular chemical reactions needed to sustain life—are a growing global health concern. In sub-Saharan Africa, for example, the prevalence (the proportion of a population that has a given disease) of adults with diabetes (a life-shortening metabolic disease that affects how the body handles sugars) is currently 3.8%. By 2030, it is estimated that the prevalence of diabetes among adults in this region will have risen to 4.6%. Similarly, in 2004, around 1.2 million deaths in sub-Saharan Africa were attributed to coronary heart disease, heart failure, stroke, and other cardiovascular diseases. By 2030, the number of deaths in this region attributable to cardiovascular disease is expected to double. Globally, cardiovascular disease and diabetes are now responsible for around 17.3 million and 1.3 million annual deaths, respectively, together accounting for about one-third of all deaths.

Experts believe that increased consumption of saturated fats, sugar, and salt and reduced physical activity are partly responsible for the increasing global prevalence of cardiometabolic diseases. These lifestyle changes, they suggest, are related to urbanization—urban expansion into the countryside and migration from rural to urban areas. If this is true, the prevalence of unhealthy lifestyles should increase as rural areas adopt urban characteristics. Sub-Saharan Africa is the least urbanized region in the world, with about 60% of the population living in rural areas. However, rural settlements across the subcontinent are increasingly adopting urban characteristics. It is important to know whether urbanization is affecting the health of rural residents in sub-Saharan Africa to improve estimates of the future burden of cardiometabolic diseases in the region and to provide insights into ways to limit this burden. In this cross-sectional study (an investigation that studies participants at a single time point), the researchers examine the distribution of urban characteristics across rural communities in Uganda and the association of these characteristics with lifestyle risk factors for cardiometabolic diseases....

For their study, the researchers used data collected in 2011 by the General Population Cohort study, a study initiated in 1989 to describe HIV infection trends among people living in 25 villages in rural southwestern Uganda that collects health-related and other information annually from its participants. The researchers quantified the "urbanicity" of the 25 villages using a multi-component scale that included information such as village size and economic activity. They then used statistical models to examine associations between urbanicity and lifestyle risk factors such as body mass index (BMI, a measure of obesity) and self-reported fruit and vegetable consumption for more than 7,000 study participants living in those villages. None of the villages had paved roads or running water. However, urbanicity varied markedly across the villages, largely because of differences in economic activity, civil infrastructure, and the availability of educational and healthcare services. Notably, increasing urbanicity was associated with an increase in lifestyle risk factors for cardiovascular diseases. So, for example, people living in villages with the highest urbanicity scores were nearly 20% more likely to be physically inactive and to eat less fruits and vegetables and nearly 50% more likely to have a high BMI than people living in villages with the lowest urbanicity scores.

These findings indicate that, across rural communities in Uganda, even a small increase in urbanicity is associated with a higher prevalence of potentially modifiable lifestyle risk factors for cardiometabolic diseases. These findings suggest, therefore, that simply classifying settlements as either rural or urban may not be adequate to capture the information needed to target strategies for cardiometabolic disease management and control in rural areas as they become more urbanized. Because this study was cross-sectional, it is not possible to say how long a rural population needs to experience a more urban environment before its risk of cardiometabolic diseases increases. Longitudinal studies are needed to obtain this information. Moreover, studies of other countries in sub-Saharan Africa are needed to show that these findings are generalizable across the region. However, based on these findings, and given that more than 553 million people live in rural areas across sub-Saharan Africa, it seems likely that increasing urbanization will have a substantial impact on the future health of populations throughout sub-Saharan Africa.

Figure 2.1: An example of an editor's summary.

- **XHTML Format**

  In PLOS Medicine, articles can be downloaded in the Extensible Hypertext Markup Language (XHTML) format as shown in Figure 2.2. It is an extended version of widely used Hypertext Markup Language (HTML). It is convenient to extract sentences of scientific articles from XHTML than extracting them from the Portable Document Format (PDF). In the example (Figure 2.2), sentences of a scientific article are shown in bold with boundary tags in the XHTML format.

```
  <div id="section1" class="section"><a id="s2" name="s2" toc="s2"
title="Introduction"></a><h3>Introduction</h3> <a id="article1.body1.sec1.p1"
name="article1.body1.sec1.p1"></a><p> Cardiometabolic diseases are a growing concern
across sub-Saharan Africa (SSA). According to current estimates, the prevalence of
diabetes among adults aged 20–79 y in Africa is 3.8\% and will increase to 4.6\% by 2030
<a href="#pmed.1001683-Shaw1">[1]</a>.  Similarly, in 2004, around 1.2 million deaths
were attributable to cardiovascular disease in the region, and this figure is expected to
double by 2030 <a href="#pmed.1001683-Wu1">[2]</a>.  Urban environments and associated
lifestyles, including diets high in salt, sugar, and fat, and physical inactivity,
have been widely implicated as leading causes of the rise in cardiometabolic diseases
<a href="#pmed.1001683-Yusuf1">[3]</a>–<a href="#pmed.1001683-Ezzati1">[5]</a>.</p> <a
id="article1.body1.sec1.p2" name="article1.body1.sec1.p2"> </a><p>Although SSA remains
the least urbanized region in the world, with over 60\% of the population still residing
in rural areas, rural settlements across the subcontinent are increasingly adopting urban
characteristics through technological improvements in transportation and telecommunication <a
href="#pmed.1001683-Sodjinou1">[6]</a>–<a href="#pmed.1001683-Chen1">[8]</a>.  If and how these
changes affect the health of rural residents, however, remains poorly understood.</p>
```

Figure 2.2: An example of XHTML format of a scientific article

The PLOS Medicine journal has been publishing 10 to 15 articles per month since 2004. We categorize all the PLOS Medicine articles up to January 2014 into training, development and testing data. The statistics of training, development and testing data is shown in Table 2.1. Also, the statistics of editors' summaries and abstracts are shown in Table 2.2 and Table 2.3 respectively.

	No. of Documents	Avg. No. of Sentences per Document	Avg. No. of Words per Document
Training	750	265.09	6013.96
Development	25	274.46	6159.35
Testing	50	154	4756

Table 2.1: Statistics of the *PLOS Medicine* dataset

	No. of Editor's Summary	Avg. No. of Sentences per Editor's Summary	Avg. No. of Words per Editor's Summary
Training	750	24.20	766.35
Development	25	24.79	762.45
Testing	50	25.72	751.25

Table 2.2: Statistics of editor's summary of the *PLOS Medicine* dataset

	No. of Documents	Avg. No. of Sentences per Abstract	Avg. No. of Words per Abstract
Training	750	13.20	365.97
Development	25	13.54	362.19
Testing	50	14.9375	369.1875

Table 2.3: Statistics of abstract of the *PLOS Medicine* dataset

2.3 News Articles

We also perform experiments on the DUC 2002 dataset which is a standard dataset for single-document summarization (Figure 2.3). DUC is a Document Understanding Conference for summarization. It is sponsored by Advanced Research and Development Activity (ARDA). The conference series are managed by National Institute of Standards and Technology (NIST) to motivate researchers in the field of Natural language Processing (NLP) to participate in large-scale experiments.

The statistics of DUC 2002 is shown in Table 2.4. These statistics show that articles in *PLOS Medicine* are longer than the articles in the DUC 2002 dataset.

We show the statistics of the DUC 2005 dataset which is being used for query based multi-document summarization in Table 2.4. In the DUC 2005 dataset, average number of sentences per document is 28.72 which is comparatively shorter than the average number of sentences per article in the *PLOS Medicine* dataset. Moreover, in case of summarization of scientific articles, non-redundancy must be considered, as they might contain many redundant information in different sections, for example, *Introduction* and *Methodology* sections of an article gives the main idea of the article. Hence, summarization of *PLOS Medicine* articles is as difficult as multi-document summarization.

Table 2.5 exhibits the statistics of the gold summaries of DUC 2002 and DUC 2005. The gold summaries of DUC 2002 are generic abstracts of the documents with a length of approximately 100 words. The abstract consists of grammatically correct and complete sentences.

	No. of Documents	Avg. No. of sentences per document	Avg. No. of words per document
DUC 2002	576	25	627
DUC 2005	1593	28.72	710.39

Table 2.4: Statistics of DUC 2002 and DUC 2005 dataset

	No. of Documents	Avg. No. of sentences per document	Avg. No. of words per document
DUC 2002	1112	5.61	113.46
DUC 2005	300	12.42	250

Table 2.5: Statistics of DUC 2002 and DUC 2005 gold summaries

2.4 Summary

In this chapter, we have provided the details of the datasets used in this thesis. We have introduced a new dataset of scientific articles of biomedicine domain. We have explained the benefits of this dataset for the summarization task. Further, we have described the standard dataset for single-document summarization. We then compared the statistics of both the datasets.

In the end, we have provided a brief description of the standard dataset for multi-document summarization. We have concluded that summarizing scientific articles is as difficult as summarizing multi-documents.

```
  <DOC>
<DOCNO> AP880911-0016 </DOCNO>
<FILEID>AP-NR-09-11-88 0423EDT</FILEID>
<FIRST>r i BC-HurricaneGilbert 09-11 0339</FIRST>
<SECOND>BC-Hurricane Gilbert,0348</SECOND>
<HEAD>Hurricane Gilbert Heads Toward Dominican Coast</HEAD>
<BYLINE>By RUDDY GONZALEZ</BYLINE>
<BYLINE>Associated Press Writer</BYLINE>
<DATELINE>SANTO DOMINGO, Dominican Republic (AP) </DATELINE>
<TEXT>
Hurricane Gilbert swept toward the Dominican Republic Sunday, and the Civil Defense alerted its
heavily populated south coast to prepare for high winds, heavy rains and high seas.  The storm
was approaching from the southeast with sustained winds of 75 mph gusting to 92 mph.  "There is
no need for alarm," Civil Defense Director Eugenio Cabral said in a television alert shortly
before midnight Saturday.  Cabral said residents of the province of Barahona should closely follow
Gilbert's movement.  An estimated 100,000 people live in the province, including 70,000 in the city
of Barahona, about 125 miles west of Santo Domingo.  Tropical Storm Gilbert formed in the eastern
Caribbean and strengthened into a hurricane Saturday night.  The National Hurricane Center in Miami
reported its position at 2 a.m.  Sunday at latitude 16.1 north, longitude 67.5 west, about 140
miles south of Ponce, Puerto Rico, and 200 miles southeast of Santo Domingo.  The National Weather
Service in San Juan, Puerto Rico, said Gilbert was moving westward at 15 mph with a "broad area of
cloudiness and heavy weather" rotating around the center of the storm....
</TEXT>
</DOC>
```

Figure 2.3: An example of a news article from DUC 2002 (d061j.AP880911-0016)

Chapter 3

Mixed Integer Programming

Integer programming models are being employed in various applications for optimization. We can categorize integer programming models on the basis of the type of variables, constraints and objective functions as shown below (Figure 3.1):

- **Linear Programming (LP)**: This model consists of variables belonging to any real number. In this model, the objective function and its constraints are linear.
- **Integer Linear Programming (ILP)**: This model consists of variables with integrality constraint. In this model, the objective function and its constraints are linear.
- **Binary Linear Programming (BLP)**: This model consists of variables which can take either 0 or 1. In this model, the objective function and its constraints are linear.
- **Integer Quadratic programming (IQP)**: This model consists of variables with integrality constraint. In this model, either the objective function or its constraints are quadratic.

For simplicity, in this section, we do not discuss integer quadratic programming in detail. We first discuss the applications of integer programming in real life to motivate the usage of integer programming models; subsequently we describe in detail the approaches to solve integer linear programming problems and binary linear programming problems in polynomial time.

We use two famous examples from real life, the 0-1 knapsack problem and the traveling salesman problem, where integer programming models are being utilized to solve them efficiently. **0-1 Knapsack Problem**: The 0-1 knapsack problem is an example of resource allocation problems. It comes under the category of binary linear programming. This problem states that, "which of n coins should be selected so that the final profit value is maximized without exceeding the maximum weight of the selected coins?". We can use integer linear programming to formulate and solve this problem.

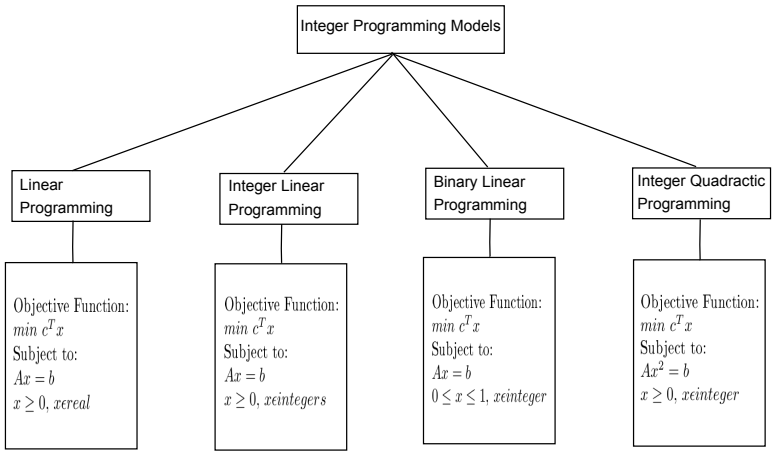


Figure 3.1: Integer programming models with examples

Variable:

$$x_i$$

Objective Function:

$$f(x) = max(\sum_{i=1}^{n} v_i x_i) \tag{3.1}$$

Subject to:

$$\sum_{i=1}^{n} w_i x_i \leq W \tag{3.2}$$

$$x_i \in \{0, 1\} \tag{3.3}$$

We have binary variable x_i which corresponds to coin i in the set of n coins. The objective function, in Equation 3.1, maximizes the profit value of the selected coins. In the objective function, v_i is a profit value associated with coin i.

The constraint in Equation 3.2 shows that the weight of all the selected coins must not exceed the maximum weight limit. Here, w_i corresponds to the weight of coin i and W corresponds to the maximum weight limit.

The constraint in Equation 3.3 shows that variable x_i associated with coin i cannot be a rational. The value of variable x_i must be integer either 0 or 1, i.e., if coin i is selected the $x_i = 1$ else $x_i = 0$.

Traveling Salesman Problem: The traveling salesman problem is an example of scheduling problems, which belongs to the category of binary linear programming. The problem states that, in a given list of n cities and the cost to visit them, find the minimal cost path that covers all cities and returns back to the starting point. Each city must be visited only once. We

can use integer programming to formulate and solve this problem as accomplished in the 0-1 knapsack problem.

Variable:

$$x_{ij}$$

Objective Function:

$$f(X) = min(\sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij}) \tag{3.4}$$

Subject to:

$$\sum_{i=1}^{n} x_{ij} = 1 \quad (j = 1, 2, \ldots, n) \tag{3.5}$$

$$\sum_{j=1}^{n} x_{ij} = 1 \quad (i = 1, 2, \ldots, n) \tag{3.6}$$

$$x_{ij} \in \{0, 1\} \tag{3.7}$$

We have a binary variable x_{ij} associated with the path from city i to city j. The objective function, in Equation 3.4, minimizes the cost function of the path for the salesman. In the objective function, c_{ij} is the cost to visit city j from city i.

The constraint in Equation 3.5 shows that the salesman must leave city i exactly once. The constraint in Equation 3.6 shows that the salesman must enter city j exactly once. The constraint in Equation 3.7 shows that variable x_{ij} is a binary variable. $x_{ij} = 1$, if the salesman goes from city i to city j, else $x_{ij} = 0$.

We define an artificial example [1] to make the integer linear programming models clearer. This is an example of integer linear programming as it contains non-binary variables with integrality constraint.

Telfa Co. produces tables and chairs, and wants to maximize profit. Each table makes $8 profit; each chair makes $5 profit. A table requires 1 hour of labor and 9 sq. feet of wood. A chair requires 1 hour of labor and 5 sq. feet of wood. We have only 6 hours of work and 45sq. feet of wood.

The graphical depiction of the problem (Figure 3.3) is shown in Figure 3.2. We plot the constraint on the axis which corresponds to variables y_1 and y_2. Here, y_1 represents the number of chairs and y_2 represents the number of tables. The feasible region is E_0 and delimited with the constraints as shown in Figure 3.2. The solution to this problem without integrality constraint (linear programming problem) is the intersection point of the constraints in the graph plot, which is evident in Figure 3.2, whereas the solution with integrality constraint

[1] This example is taken from a presentation given by *Dan Roth* in the University of Heidelberg in year 2013

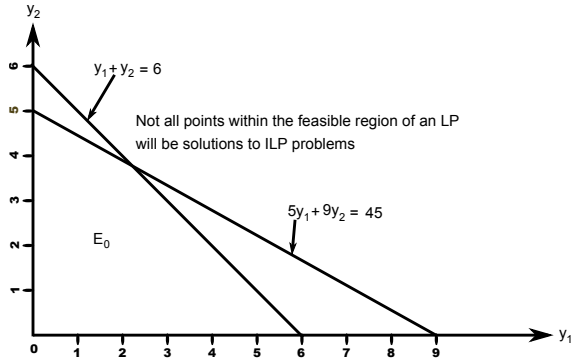


Figure 3.2: An integer linear programming example.

Variables:

$$y_1 \quad \& \quad y_2 \tag{3.8}$$

Objective Function:

$$f(y_1, y_2) = max_{y_1,y_2} 5y_1 + 8y_2$$

Subject to:

$$\begin{aligned} y_1 + y_2 &\leq 6 \\ 5y_1 + 9y_2 &\leq 45 \\ y_1, y_2 &\geq 0 \\ y_1, y_2 &\in Integers \end{aligned}$$

Figure 3.3

(integer linear programming problem) is $y_1 = 0$ and $y_2 = 5$, objective function $f(y_1, y_2) = 40$. The linear programming solution defines the upper limit for the integer linear programming problem.

The integer linear programming has NP-Hard complexity, whereas linear programming can be solved in polynomial time using the simplex method. It will be very exhaustive to search the best solution to an integer linear programming problem among all the potential optimal solutions. There are sophisticated procedures which can solve integer linear programming problems in polynomial time. These procedures can be classified into three different categories:

- enumeration techniques
- cutting plane techniques; and

- group theoretic techniques.

Before discussing the techniques used to solve integer linear programming, we discuss in detail the simplex method which is used to solve linear programming. Then, we provide the detailed description of the enumeration techniques (see Sections 3.2 and 3.3) and the cutting plane technique (see Section 3.4).

3.1 Simplex

The simplex method is used to solve the linear programs in polynomial time. This method is the base of all other techniques used to solve integer linear programs. For simplicity, we give an illustrative example [2] of the simplex method. We explain the steps of this method by considering an example of linear program. The graphical representation of the problem is shown in Figure 3.4.

Variables:

$$x_1 \quad \& \quad x_2 \tag{3.9}$$

Objective Function:

$$max \quad x_1 + x_2$$

Subject to:

$$2x_1 + x_2 \leq 4$$
$$x_1 + x_2 \leq 3$$
$$x_1 \geq 0, \quad x_2 \geq 0$$

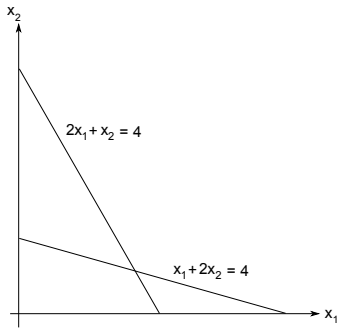


Figure 3.4: A graph plot of the problem.

First, we transform the inequality constraints to the standard form by adding slack variables. Thus, the problem can be rewritten as:

[2] http://mat.gsia.cmu.edu/classes/QUANT/

Objective Function:

$$max \quad x_1 + x_2$$

Subject to:

$$\begin{array}{cccccc} 2x_1 & +x_2 & +s_3 & & = & 4 \\ x_1 & +2x_2 & & +s_4 & = & 3 \\ x_1 \geq 0, & x_2 \geq 0, & s_3 \geq 0, & s_4 \geq 0 & & \end{array}$$

We denote the objective function with variable z. Therefore, $z = x_1 + x_2$, or can be rewritten as $z - x_1 - x_2 = 0$. Now, we place this equation with the constraints as shown below:

$$\begin{array}{ccccccc} z & -x_1 & -x_2 & & & = & 0 \quad \text{ROW 0} \\ & 2x_1 & +x_2 & +s_3 & & = & 4 \quad \text{ROW 1} \\ & x_1 & +2x_2 & & +s_4 & = & 3 \quad \text{ROW 2} \end{array}$$

The goal is to maximize the objective function, while fulfilling the constraints and, also, $x_1 \geq 0$, $x_2 \geq 0$, $s_3 \geq 0$, $s_4 \geq 0$.

In the simplex method, variables are divided into two categories: *basic* variables and *non-basic* variables. *Basic* variables are those which appear in only one equation. Here, *basic* variables are x_3 and x_4 and *non-basic* variables are x_1 and x_2. A basic solution to the problem is by setting all non-basic variables to zero, which yields the following solution: $x_1 = x_2 = 0$ $s_3 = 4$ $s_4 = 3$ $z = 0$. This basic solution is not optimal; hence we can increase the value of objective function. $ROW0$ above shows that we can increase z by increasing the non-basic variables x_1 or x_2. This is due to the fact that the coefficients of the non-basic variables are negative. This is Rule 1 in the simplex method. Rule 1 states that: if all variables have non-negative coefficients in $ROW0$, then the basic solution is the optimal solution. Otherwise one variable from $ROW0$ with the negative coefficient is selected.

The variable with the negative coefficient chosen by the above stated rule is known as *entering* variable. Here, we choose x_1 as the *entering* variable. Choosing a variable from $ROW0$ does not affect the final solution, provided the variable should have negative coefficient in $ROW0$. The intuition is to *pivot* so that *non-basic* variable x_1 becomes a *basic* variable. This is done by using the *Gauss-jordan* procedure.

We need to choose the *pivot* element by using Rule 2 of the simplex method. Rule 2 states that: "If there is a strictly positive *entering* variable coefficient in each Row i, $i > 0$, then compute the ratio of the right-hand side (RHS) to the coefficient of *entering* variable. Subsequently, choose the row as *pivot* if it has the minimum ratio". In this example, *pivot* row is $ROW1$, since the ratio we get in $ROW1$ is $\frac{4}{2}$ which is less than the ratio in case of ROW 2, i.e., $\frac{3}{1}$.

The simplex method iterates between Rule 1 and Rule 2 until the optimal solution is obtained. After using Rule 2 to find the *pivot* row, we apply *Gauss-jordan* pivot and obtain the equations as follows:

$$\begin{array}{rrrrcll} z & & -\frac{1}{2}x_2 & +\frac{1}{3}s_3 & = & 2 & \text{ROW 0} \\ & x_1 & +\frac{1}{2}x_2 & +\frac{1}{2}s_3 & = & 2 & \text{ROW 1} \\ & & \frac{3}{2}x_2 & -\frac{1}{2}s_3 \quad +s_4 & = & 1 & \text{ROW 2} \end{array}$$

The basic solution with these equations is $x_2 = s_3 = 0$, $x_1 = 2$, $s_4 = 1$, $z = 2$. Then, we again check Rule 1 for the optimal solution. Here, the solution is not optimal due to one negative entry in $ROW0$. According to Rule 1 x_2 is chosen as an *entering* variable. Subsequently, Rule 2 determines the pivot row and element. $ROW2$ is a *pivot* row, since the ratio for $ROW1$ ($\frac{4}{1}$) is greater than the ratio for $ROW2$ ($\frac{2}{3}$). This results in the following equations:

$$\begin{array}{rrrrcll} z & & +\frac{1}{3}x_3 & +\frac{1}{3}x_4 & = & \frac{7}{3} & \text{ROW 0} \\ & x_1 & +\frac{2}{3}x_3 & -\frac{1}{3}x_4 & = & \frac{5}{3} & \text{ROW 1} \\ & x_2 & -\frac{1}{3}x_3 & +\frac{2}{3}x_4 & = & \frac{2}{3} & \text{ROW 2} \end{array}$$

The basic solution with these equations is $s_3 = s_4 = 0$, $x_1 = \frac{5}{3}$, $x_2 = \frac{2}{3}$, $z = \frac{7}{3}$. According to Rule 1 this basic solution is the optimal solution of the problem, as there is no negative coefficients in $ROW0$.

The above computations can be represented in the *tableau form* as shown below:

z	x_1	x_2	s_3	s_4	RHS	$Basic\_solution$
1	-1	-1	0	0	0	$basic\ x_3 = 4\ x_4 = 3$
0	2	1	1	0	4	$non-basic\ x_1 = x_2 = 0$
0	1	2	0	1	3	$z = 0$
1	0	$-\frac{1}{2}$	$\frac{1}{2}$	0	2	$basic\ x_1 = 2\ x_4 = 1$
0	1	$\frac{1}{2}$	$\frac{1}{2}$	0	2	$non-basic\ x_2 = x_3 = 0$
0	0	$\frac{3}{2}$	$-\frac{1}{2}$	1	1	$z = 2$
1	0	0	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{7}{3}$	$basic\ x_1 = \frac{5}{3}\ x_2 = \frac{2}{3}$
0	1	0	$\frac{2}{3}$	$-\frac{1}{3}$	$\frac{5}{3}$	$non-basic\ x_3 = x_4 = 0$
0	0	1	$-\frac{1}{3}$	$\frac{2}{3}$	$\frac{2}{3}$	$z = \frac{7}{3}$

3.2 Branch-and-Bound

Branch-and-Bound technique is an enumeration technique, which is used for solving integer linear programming problems. This technique is based on the strategy of "divide and conquer". The basic idea is to divide the feasible region into subregions iteratively until the integer optimal solution is obtained.

Let us take the same example which we have discussed above in Figure 3.3. In this problem, if there is no integrality restriction, then the solution is $y_1 = 2.25$, $y_2 = 3.75$ and $ob_{up} = f(y_1, y_2) = 41.25$. In this maximization problem, the value of the objective function obtained by the linear programming model will always be greater than the value obtained by the integer linear programming model. Hence, $f(y_1, y_2) = 41.25$ will be the upper bound in Figure 3.5, but for integer linear programming, the upper bound of the objective function must be integral, therefore $f^*(y_1, y_2) \leq 41$. The solution obtained from the linear program-

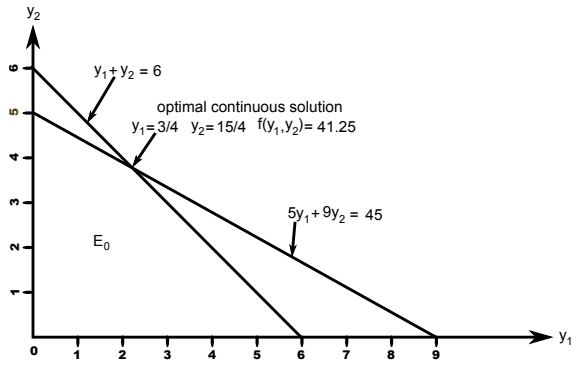


Figure 3.5: A graph plot of another problem.

ming model will be an optimal solution for integer linear programming, only if the solution is an integer.

In order to make at least one of the variables an integer, we can divide the feasible region (Figure 3.5) into subregions. It is evident from the value of $y_2 = 3\frac{3}{4}$, which is obtained from linear programming, that the solution of integer linear programming lies in the region where y_2 must be an integer, either ≤ 3 or ≥ 4. Thus, the first subdivision of the feasible region is where $y_2 \leq 3$ and $y_2 \geq 4$ as shown in Figure 3.6. Due to the subdivisions, we discard the linear programming solution ($y_2 = 3.75$), which is represented in Figure 3.6 as the non-shaded part. We can also take y_1 in place of y_2, then the subdivisions would be $y_1 \leq 2$ and $y_1 \geq 3$.

We illustrate the calculations up to this point, in the enumeration tree as shown in Figure 3.7. Here, E_0 represents the linear optimal region, and the solution is shown in the E_0 box. The upper bound of the objective function for integral linear programming is shown next to the E_0 box. The boxes E_1 and E_2 connected with E_0 correspond to the new subdivisions (Figure 3.7).

We continue the subdivisions of regions until we get the optimal integral solution for the problem. Considering region E_1 from Figure 3.6, we see that the optimal linear programming solution lies on the second constraint ($5y_1 + 9y_2 \leq 45$) with $y_2 = 4$, producing $y_1 = \frac{1}{5}(45 - 9(4)) = \frac{9}{5}$ and an objective value $f(y_1, y_2) = 5(\frac{9}{5}) + 8(4) = 41$. Since y_1 is not an integer, we subdivide region E_1 further. After division, the regions are E_3 with $y_1 \geq 2$ and E_4 with $y_1 \leq 1$ (Figure 3.9), but region E_3 does not have a feasible solution, so this is no longer considered.

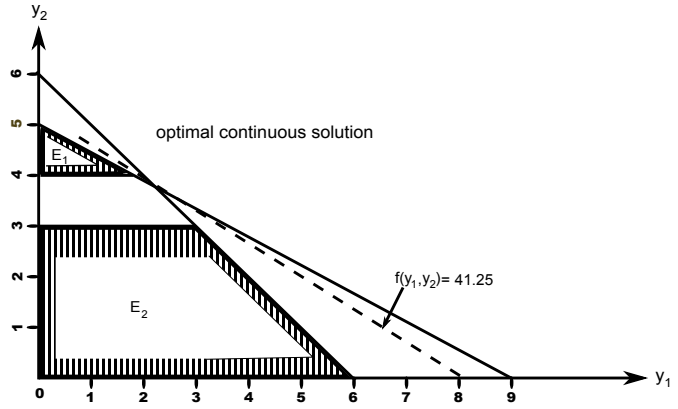


Figure 3.6: A graph plot with subdivisions E_1 and E_2.

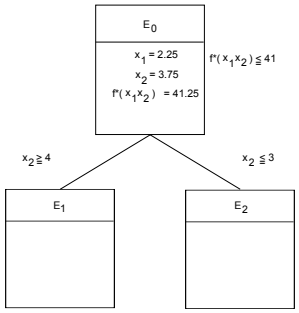


Figure 3.7: An enumeration tree with subdivisions E_1 and E_2.

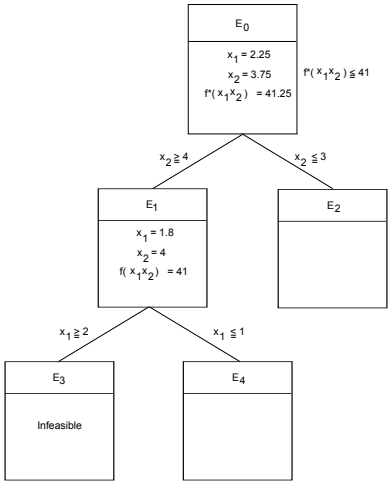


Figure 3.8: An enumeration tree with subdivisions E_3 and E_4.

The enumeration tree with regions E_3 and E_4 is shown in Figure 3.8. According to the tree (Figure 3.8), we have two regions E_2 and E_4. We can continue with either of the regions. For simplicity, we consider the recent region E_4, and the associated optimal solution $y_1 = 1$,

$y_2 = \frac{1}{9}(45 - 5) = \frac{40}{9}$ as shown in Figure 3.10.

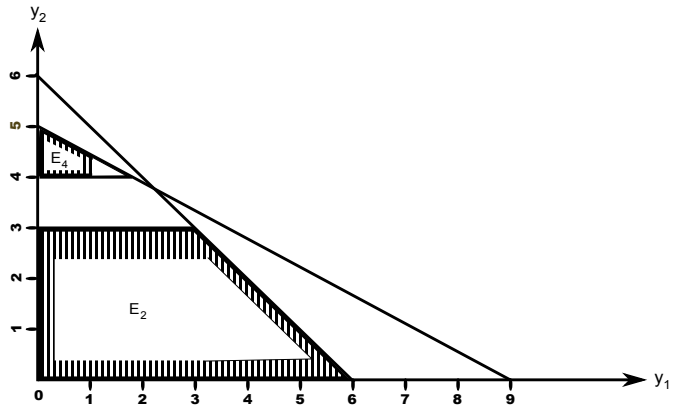


Figure 3.9: A graph with subdivisions E_3 and E_4.

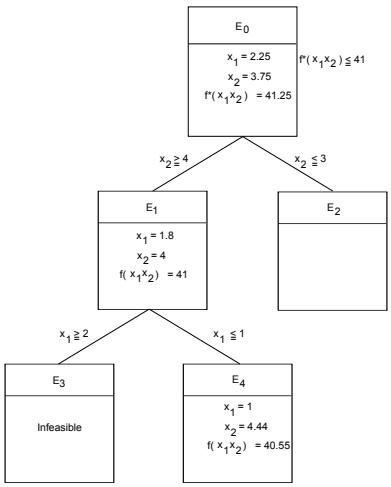


Figure 3.10: An enumeration tree with the solution over region E_4.

Since y_2 is not an integer, so we further divide E_4 into E_5 with $y_2 \leq 4$ and E_6 with $y_2 \geq 5$ (Figure 3.11). We consider E_5, the optimal solution E_5 has $y_1 = 1$, $y_2 = 4$ and $f(y_1, y_2) = 37$. This is the best integer linear programming solution we can get by considering region E_5. Therefore, we do not need to divide E_5 further. We can terminate the search for the solution, but it is possible that E_6 and E_2, which are not yet considered, may contain even a better integer solution.

Until now the best integer solution is $y_1 = 1$, $y_2 = 4$ over region E_5, without exploring regions E_6 and E_2. However, we must consider region E_6 and E_2 to find a better solution. In region E_6, the only feasible point is $y_1 = 0$, $y_2 = 5$ and the objective value $f(y_1, y_2) = 40$. The objective value over region E_6 is better than the objective value over region E_5. Hence, the best solution becomes $y_1 = 0$, $y_2 = 5$. We could terminate the search with this solution,

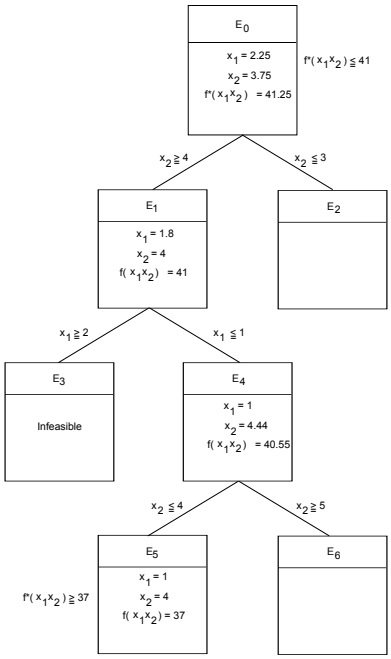


Figure 3.11: An enumeration tree with subdivisions E_5 and E_6.

but we have not yet considered region E_2. In region E_2, the linear programming solution is $y_1 = 3$, $y_2 = 3$ and the objective value $f(y_1, y_2) = 39$. The solution over E_2 is less than the solution over E_6 so the best integer programming solution obtained is $y_1 = 0$, $y_2 = 5$. The final solution of the problem is shown in the enumeration tree in Figure 3.13.

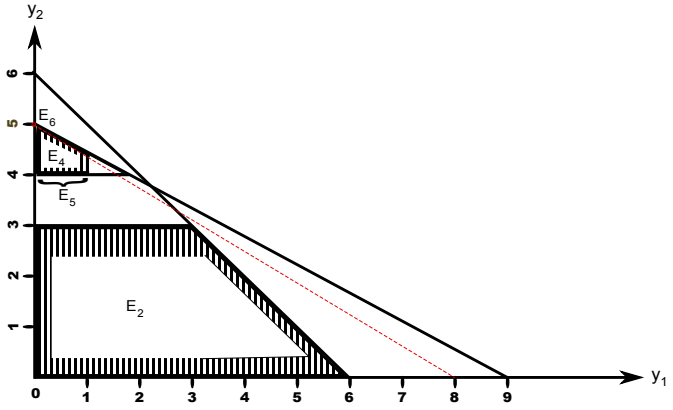


Figure 3.12: A graph plot with subdivisions E_5 and E_6.

The E_j region cannot be divided further if it meets at least one of the following conditions:

- **Fathoming by infeasibility**: The solution in the E_j region is infeasible;
- **Fathoming by integrality**: The optimal linear programming solution over E_j is integer; or

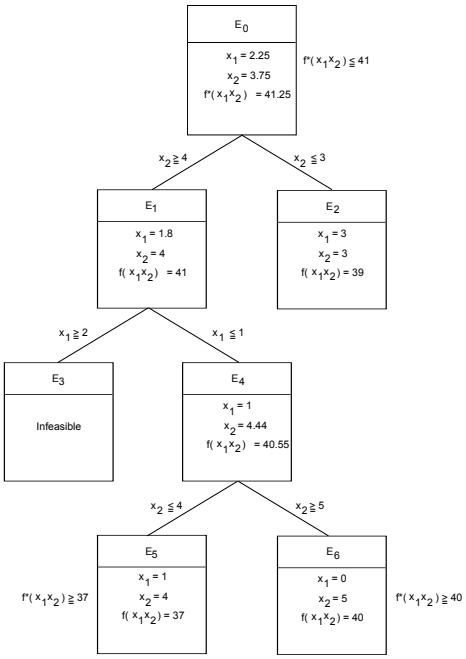


Figure 3.13: An enumeration tree with all subdivisions

- **Fathoming by bounds**: The objective value of the linear programming solution Ob_j over E_j is less than or equal to the solution obtained from linear programming model (ob_{up}).

The flowchart in Figure 3.14 [3] is a summary of branch-and-bound algorithm to solve integer linear programing problems in polynomial time.

[3] http://web.mit.edu/15.053/www/AMP-Chapter-09.pdf

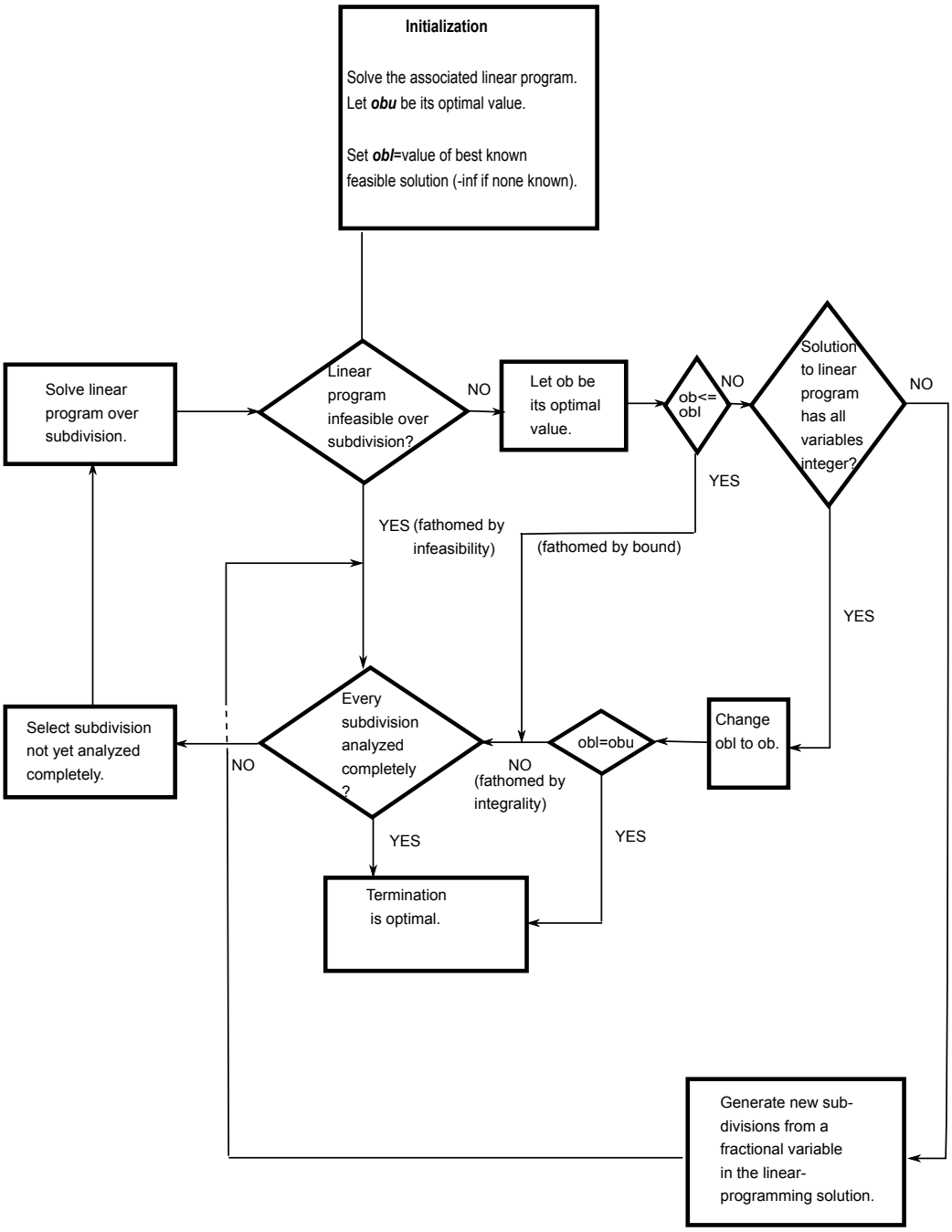


Figure 3.14: Flowchart of Branch-and-bound for integer linear programming

3.3 Implicit Enumeration

Implicit Enumeration is a special branch-and-bound procedure which solves binary linear programming problems in polynomial time. In contrast to the branch-and-bound procedure, implicit enumeration does not require linear programming solutions. We explain implicit enumeration in detail using the example given below [4]:

Variables:

$$x_1, x_2, x_3, x_4, x_5$$

Objective Function:

$$z^* = max(z) = -8x_1 - 2x_2 - 4x_3 - 7x_4 - 5x_5 + 10$$

Subject to:

$$\begin{aligned} &-3x_1 - 3x_2 + x_3 + 2x_4 + 3x_5 \leq -2, \\ &-5x_1 - 3x_2 - 2x_3 - x_4 + x_5 \leq -4, \\ &x_j = 0 \quad or \quad 1 \qquad (j = 1, 2, ..., 5). \end{aligned}$$

To solve this problem, we always maintain the 0-1 restriction, but ignore the linear inequalities. We utilize the idea of the branch-and-bound procedure to set few variables at either 0 or 1. The variables which are not set to 0 or 1 are called *free variables*. We can choose any variables as *fixed variables* or *free variables*. Since we ignore the inequality constraints, so to maximize the objective function in Equation 3.3, we set the *free variables* to zero. The reason to fix *free variables* to zero is due to the negative coefficients of variables in the objective function. For instance, if $x_1 = x_4 = 1$ and $x_5 = 0$ are fixed, then the free variables are x_2 and x_3. The resulting problem after ignoring the inequalities, and considering fixed variables is:

Variables:

$$x_1, x_2, x_3, x_4, x_5$$

Objective Function:

$$z^* = max(-8(1) - 2x_2 - 4x_3 - 7(1) - 5(0) + 10) = max(-2x_2 - 4x_3 - 5),$$

Subject to:

$$x_2 \quad \& \quad x_3 \text{ either } 0 \text{ or } 1.$$

[4] http://web.mit.edu/15.053/www/AMP-Chapter-09.pdf

The objective function z^* is maximized by assigning $x_2 = x_3 = 0$ because the variables have negative coefficients.

Coming back to the example, we start with all the variables as *free variables* and set them to zero. The solution does not satisfy the inequality constraints; hence to search for feasible solutions, we must subdivide with any *free variable*. We start subdividing with x_1 variable:

subdivision 1 : $x_1 = 1$,

subdivision 2 : $x_1 = 0$.

In subdivision 1, we assign the value one to variable x_1, whereas in subdivision 2, we assign the value zero. If we ignore the inequalities in subdivision 1, the optimal solution over subdivision 1 has $x_2 = x_3 = x_4 = x_5 = 0$. The resulting value of the objective function in subdivision 1 gives

$$z = -8(1) - 2(0) - 4(0) - 7(0) - 5(0) + 10 = 2. \tag{3.10}$$

The obtained solution over subdivision 1 satisfies the inequalities; hence, the optimal value to the original problem is at least 2. The optimal solution over subdivision 1 is best among all the 0-1 combinations of the variables with $x_1 = 1$. We do not need to evaluate all the combinations of the variables in subdivision 1 explicitly as they have been considered implicitly in the solution 3.10.

The solution over subdivision 2 has $x_2 = x_3 = x_4 = x_5 = 0$ and $x_1 = 0$, and is not feasible as this does not satisfy the inequality constraints. Hence this must be subdivided further, for example with variable x_2:

subdivision 3: $x_1 = 0$,$x_2 = 1$; subdivision 4: $x_1 = 0$, $x_2 = 0$.

Similar to the branch-and-bound procedure, we illustrate the solution up to this point in the enumeration tree as shown in Figure 3.15.

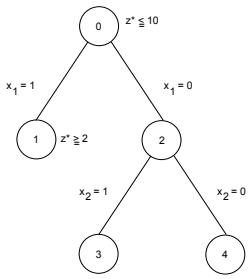


Figure 3.15: A binary enumeration tree with subdivisions.

In subdivision 3, we set $x_3 = x_4 = x_5 = 0$ with $x_1 = 0$ and $x_2 = 1$, but the solution over subdivision 3 is not feasible as it does not satisfy the inequality constraints. Consequently, the region must be subdivided further.

Continuing to subdivide and fix variables in the same way gives the complete enumeration tree shown in Figure 3.16. The optimal solution as shown in the final enumeration tree is at subdivision 5 with $x_1 = x_4 = x_5 = 0$ with $x_2 = x_3 = 0$, and objective value is 4.

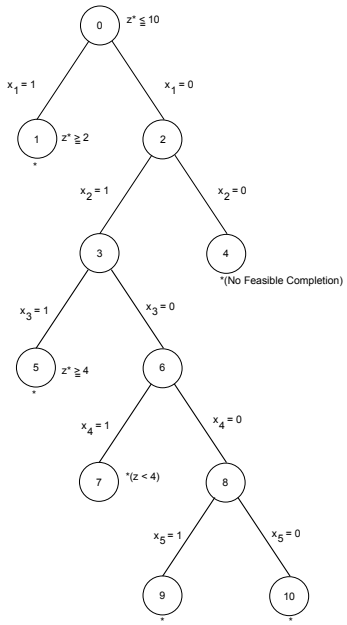


Figure 3.16: A final enumeration tree [5].

3.4 Cutting Planes

The cutting-plane algorithm is another approach to solve integer linear programming problems in polynomial time. This algorithm modifies the linear programming solution until the integer solution is acquired. It refines the linear programming problem by adding new constraints which consecutively reduces the feasible region until an integer solution is obtained. In contrast to the branch-and-bound procedure, it does not partition the feasible region into subdivisions. However, branch-and-bound generally outperforms the cutting-plane algorithm.

We discuss in detail the cutting-plane algorithm by considering the same problem of Section 3.2:

Variables:

$$y_1 \quad \& \quad y_2 \tag{3.11}$$

Objective Function:

$$f(y_1, y_2) = max_{y_1, y_2} 5y_1 + 8y_2$$

Subject to:

$$\begin{aligned} y_1 + y_2 &\leq 6 \\ 5y_1 + 9y_2 &\leq 45 \\ y_1, y_2 &\geq 0 \\ y_1, y_2 &\in Integers \end{aligned}$$

We first convert the inequality constraints to equality constraints by adding slack variables as shown below:

Subject to:

$$\begin{array}{rrrrcl} y_1 & +y_2 & +s_1 & & = & 6, \\ 5y_1 & +9y_2 & & +s_2 & = & 45, \end{array}$$

where s_1 and s_2 are slack variables for the first and second constraints respectively.

The optimal tableau obtained after applying simplex method to solve the problem is shown below:

$$\begin{array}{rrrrrcl} (-z) & & & -\frac{5}{4}s_1 & -\frac{3}{4}s_2 & = & -41\frac{1}{4} \\ & y_1 & & +\frac{9}{4}s_1 & -\frac{1}{4}s_2 & = & \frac{9}{4} \\ & & y_2 & -\frac{5}{4}s_1 & +\frac{1}{4}s_2 & = & \frac{15}{4} \\ & y_1, & y_2, & s_1, & s_2 & \geq & 0 \end{array}$$

We rewrite the above constraints so that both sides of each equality contain only integer coefficients and the other side contains only fractional coefficients. The constant terms on the right-hand side of the constraints are all positive and the coefficients of the slack variables on the right-hand side are all negative.

$$\begin{array}{rrrrrcrrr} (-z) & & -2s_1 & -s_2 & +42 & = & \frac{3}{4} & -\frac{3}{4}s_1 & -\frac{1}{4}s_2 \\ & y_1 & +2s_1 & -s_2 & -2 & = & \frac{1}{4} & -\frac{1}{4}s_1 & -\frac{3}{4}s_2 \\ & y_2 & -2s_1 & & -3 & = & \frac{3}{4} & -\frac{3}{4}s_1 & -\frac{1}{4}s_2 \\ & y_1, & y_2, & s_1, & s_2 & \geq & 0 & & \end{array}$$

The right-hand side of the constraints contain slack variables s_1 and s_2 with negative coefficients, which shows that each right-hand side must be less than or equal to the fractional constant term. The left-hand side of the constraints contains integer coefficients and integer constant terms hence in any integer solution the left-hand side must be an integer. As a result of the above two observations, we state that both sides of the equations must be integers, less than or equal to zero as shown below:

$$\begin{aligned}
\tfrac{3}{4} - \tfrac{3}{4}s_1 - \tfrac{1}{4}s_2 \leq 0 \quad & and \quad integer, \\
\tfrac{1}{4} - \tfrac{1}{4}s_1 - \tfrac{3}{4}s_2 \leq 0 \quad & and \quad integer, \\
\tfrac{3}{4} - \tfrac{3}{4}s_1 - \tfrac{1}{4}s_2 \leq 0 \quad & and \quad integer.
\end{aligned}$$

The equations can be rewritten by introducing slack variables,

$$\begin{aligned}
\tfrac{3}{4} - \tfrac{3}{4}s_1 - \tfrac{1}{4}s_2 + s_3 = 0, \quad s_3 \geq 0 \quad & and \quad integer, \qquad (C_1) \\
\tfrac{1}{4} - \tfrac{1}{4}s_1 - \tfrac{3}{4}s_2 + s_4 = 0, \quad s_4 \geq 0 \quad & and \quad integer, \qquad (C_2) \\
\tfrac{3}{4} - \tfrac{3}{4}s_1 - \tfrac{1}{4}s_2 + s_5 = 0, \quad s_5 \geq 0 \quad & and \quad integer. \qquad (C_3)
\end{aligned}$$

The above derived equations (C_1), (C_2) and (C_3) are known as *cuts*. The effect of a cut is to remove the optimal linear solution from the feasible region without eliminating the optimal integer solution of a problem.

The mathematical analysis of cuts which is obtained above can be recognized easily. For this, we consider the equations where we added slack variables to convert inequality constraints to equality constraints (Equation 3.4). The equations are rewritten to obtain only slack variables s_1 and s_2 on the left-hand side (Equation 3.4).

$$\begin{aligned}
s_1 &= 6 - y_1 - y_2 \\
s_2 &= 45 - 5y_1 - 9y_2.
\end{aligned}$$

After substituting the values of slack variables s_1 and s_2 in the cut inequality constraints, we obtain the equations shown below:

$$\begin{aligned}
2y_1 + 3y_2 &\leq 15 \quad C_1 \text{or } C_3 \\
4y_1 + 7y_2 &\leq 35 \quad C_2
\end{aligned}$$

We plot these cut constraints on the y_1 and y_2 axis shown in Figure 3.17. We can interpret from the plot, that the cut removes the linear programming solution from the feasible region and does not exclude any feasible integer solution.

The cutting-plane technique adopts the strategy of adding the cuts to the constraints which define feasible regions and then to solve the subsequent linear program using simplex or dual simplex. In Figure 3.17, $C_1 = C_3$ gives the optimal integer solution to the problem and say it as a final solution to the problem. C_2 does not give any integer solution, and hence further cut is added to solve the problem.

In the end, we say that the basic strategy in the cutting-plane technique is to solve the linear programming problem using the simplex method and introduce additional constraints known as cuts until we get an integer solution. We add the cut constraints with the guarantees discussed below:

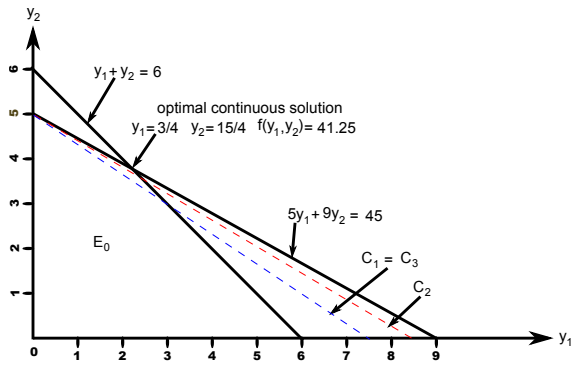


Figure 3.17: A graph plot with cutting planes.

- The optimal integer solutions must not be eliminated.
- Every cut constraint must reduce the feasible region.
- Every cut constraint must pass through an integer point.

3.5 Summary

In this chapter, we have explained the optimization problems and their solutions. Moreover, we have discussed integer linear programming which is utilized to solve the optimization problems. Integer linear programming provides the globally optimal solutions to the problems. It is difficult to obtain the exact solutions of the optimization problems. To solve this intractability problem, we have explained the approximate solutions.

Chapter 4

Latent Dirichlet Allocation

Nowadays, online text documents are growing exponentially. We do not have enough human power to sort the documents topic-wise for further analysis. In order to find topics in the documents, researchers in the field of machine learning have developed probabilistic topic models. Probabilistic topic models are algorithms utilized to find the hidden thematic structure in a large cluster of documents. These structures can be useful in various fields, such as, information retrieval, summarization, question answering, and classification.

In this section, we discuss in detail the most basic probabilistic topic model which is Latent Dirichlet Allocation (LDA) (Blei et al., 2003). The main intuition behind LDA is that "Every document consists of several topics". For instance, in Figure 4.1, we show a bio-medicine article along with the topics and their distribution associated with the article, which is entitled *"Risk of Adverse Pregnancy Outcomes among Women Practicing Poor Sanitation in Rural India: A Population-Based Prospective Cohort Study"*.

LDA is a statistical model that tries to find the topics from the document collections. It is a generative process by which the model assumes that the documents are being generated.

In LDA, a topic is a distribution over words which is derived from a fixed vocabulary. For instance, in Figure 4.1, we show the distribution over words in every topic. Every topic contains few words having high probability; "topic 1" has words related to "places" and "topic 3" has words related to "diseases". Since LDA is a generative process, it assumes that these topics are identified before the documents are produced. The words are generated by LDA in two stages for each document in the collection.

1 A topic distribution is chosen randomly for the document.

2 For every word in the document,

  2a A topic from the distribution over topics from step 1 is chosen randomly.

  2b A word is chosen randomly from the distribution over the vocabulary.

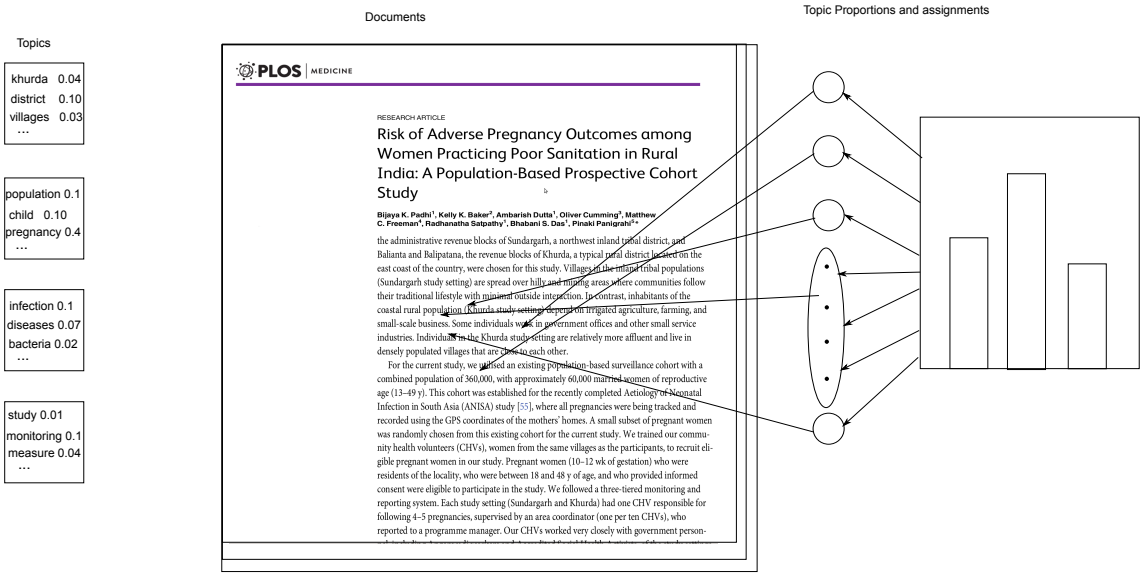


Figure 4.1: Latent Dirichlet Allocation.

The distinguishing feature of LDA is that all the documents in the collection share the same topics, but in a different proportion. The example article in Figure 4.1 consists of topics "small places", "unhygienic diseases", and "pregnancy outcomes" having high probability. However, it is possible that the next article contains topics related to "data analysis", "genetics" etc., with high probability.

LDA hypothesizes that the documents are generated by a hidden structure, viz. the topic distribution. The hidden structure of the observed document consists of the topics, per document topic distributions, and per document per word topic assignments. The hidden structure inferred from the documents is similar to the thematic structure of the document collection. This structure assists in annotating each document of the collection with the topics.

4.1 Details of LDA

LDA is a part of the generative probabilistic modeling field. In generative models, the observed data is assumed to be emerging from a process that consists of hidden variables. This process models a joint probability distribution over both the observed data and hidden variables. This joint probability distribution is used to compute conditional probability distributions of the hidden variables given the observed variables. The conditional probability distribution is also referred to as the posterior distribution. LDA computes the posterior distribution to infer the hidden structure from the documents.

LDA is more formally described with the following notations:

- $\beta_1 : K$ are the topics, where β_k is a distribution over words in the vocabulary.
- θ_d is the topic proportions of document d (lefthand side of Figure 4.1), where the topic proportion for topic k in document d is $\theta_{d,k}$.
- z_d is the topic assignments for the words in document d, where $z_{d,n}$ is the topic assignment for word n in document d.
- w_d are the observed words in document d, where $w_{d,n}$ is word n in document d.

The notations described above are used to formulate the joint probability distribution of the observed and hidden variables for LDA,

$$p(\beta_{1:K}, \theta_{1:K}, z_{1:K}, w_{1:N}) = \prod_{i=1}^{K} p(\beta_i) \prod_{d=1}^{D} p(\theta_d)(\prod_{n=1}^{N} p(z_{d,n}|\theta_d)p(w_{d,n}|\beta_{1:K}, z_{d,n})). \tag{4.1}$$

The probability of word $w_{d,n}$ is formulated as Equation 4.2,

$$p(w|\alpha, \beta) = \int p(\theta|\alpha) \left(\prod_{n=1}^{N} \sum_{z_n} p(z_n|\theta)p(w_n|z_n, \beta) \right) d\theta \tag{4.2}$$

, where α is a proportion parameter. The probability of the document cluster is shown in Equation 4.3

$$p(D|\alpha, \beta) = \prod_{d=1}^{D} \int p(\theta_d|\alpha) \left(\prod_{n=1}^{N_d} \sum_{z_{d,n}} p(z_{d,n}|\theta_d)p(w_{d,n}|z_{d,n}, \beta) \right) d\theta_d. \tag{4.3}$$

In Equation 4.1, the joint probability distribution shows a number of dependencies, such as $z_{d,n}$ depends on per document topic distribution θ_d. Word $w_{d,n}$ depends on topics $\beta_{1:K}$ and the topic assignment $z_{d,n}$. The dependencies of observed and hidden variables are shown graphically in Figure 4.2, where η is a topic parameter. Each node in Figure 4.2 is a random variable, where the shaded node $w_{d,n}$ is an observed variable and the unshaded nodes are the hidden variables. The rectangular boxes around the variables are "plates" which represent replication. Plate N denotes the words in a document and plate D denotes the documents in the document cluster.

Formally, Blei & Lafferty (2009) define the basic steps for the generative probabilistic topic model as shown below:

1 Draw a distribution over words for each topic k, denoted as $\varphi \sim Dir(\alpha)$

2 For each document d in document collection

   a Draw a topic proportions' vector $\theta_d \sim Dir(\beta)$

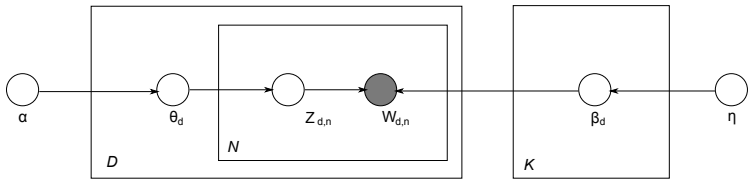


Figure 4.2: The plate representation of LDA.

b For each word $w_{d,n}$ in document d

i Draw a topic assignment $z_{d,n} \sim Mult(\theta_d)$, where $z_{d,n} \in 1 \cdots K$

ii Draw a word $w_{d,n} \sim Mult(\varphi_{z_{d,n}})$, where $w_{d,n} \in 1 \cdots V$

In the above description by Blei & Lafferty (2009), $Dir(\alpha)$ is the dirichlet distribution of α which we discuss in detail in Section 4.2.

Tables 4.1, 4.2 and 4.3 demonstrate a simple example. The example consists of four documents in the document collection. Table 4.1 exhibits the topic distribution of each document. Table 4.2 exhibits the per word topic assignment of each document. Table 4.3 exhibits the per topic term distribution, where terms are words in the specific vocabulary. These tables give an overview of LDA, which is formally described above.

	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
Document 1	0.3	0.2	0.2	0.1	0.2
Document 2	0.2	0.18	0.02	0.24	0.36
Document 3	0.54	0.10	0.16	0.13	0.07
Document 4	0.35	0.23	0.2	0.2	0.02

Table 4.1: Topic distribution of each document

	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7
Document 1	k=4	k=1	k=5	k=3	k=4	k=1	k=2
Document 2	k=3	k=4	k=4	k=1	k=2	k=3	k=5
Document 3	k=1	k=5	k=4	k=2	k=2	k=1	k=3
Document 4	k=4	k=2	k=5	k=1	k=3	k=1	k=2

Table 4.2: Topic assignment to each word

The probability computation consists of summation of all possible combinations of topic assignments, which is intractable if the dataset is large. Hence, machine learning techniques are used to solve this problem, which we discuss in Section 4.3.

	Term 1	Term 2	Term 3	Term 4	Term 5	Term 6
Topic 1	0.3	0.4	0.07	0.1	0.03	0.1
Topic 2	0.2	0.2	0.1	0.3	0.11	0.09
Topic 3	0.18	0.5	0.22	0.05	0.0	0.05
Topic 4	0.15	0.25	0.1	0.35	0.1	0.05
Topic 5	0.13	0.27	0.3	0.12	0.08	0.1

Table 4.3: Term distribution for each topic

4.2 Dirichlet Distribution

To understand the dirichlet distribution in detail, we would consider an example of "Rolling Dice". **Rolling Dice**: A manufacturing company wants to produce six faced dice but only wants the outcome to be 1,2, and 3. So the sides of a dice should not have numbers greater than 3 and smaller than 1. If the company wants to produce a fair dice than the probability of each outcome will be equal, $\theta = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$. However, if the company produces loaded dice, i.e., more than two faces with the same number let us assume 3, than the probability of 3 will be greater than the probability of 1 and 2, $\theta = (\frac{1}{6}, \frac{2}{6}, \frac{3}{6})$.

Regardless of the type of a dice, the probability θ has two properties: the sum of probabilities of each outcome must be 1, $\sum_i \theta_i = \theta_1 + \theta_2 + \theta_3 = 1$, and none of the probabilities can be negative. When these properties hold, probabilities related to the rolling dice are defined by the multinomial distribution.

The probabilities of the outcomes do not only depend on whether the dice is fair or loaded but also on the characteristics of the dice. So even if the dice is a fair dice, we do not expect the probability of each outcome to be $\frac{1}{3}$. The reason behind this is the lack of precision in the manufacturing process of a dice. If a dice is a hand-made dice, we would anticipate significant variability in the produced dice. The quality and precision of a dice depends on the density of wood, tools used to create a dice, etc. If a dice is made by a machine than the dice is more precise and significantly less variable than the hand-made dice.

To formulate this variability, we need to know the probability of each outcome of a dice (θ) for a specific manufacturing process. For this, we assume that each element of θ is an independent variable, i.e, θ can be represented as a 3-dimensional vector as shown in Figure 4.3. The equilateral triangle (a 2-simplex) in Figure 4.3 shows all the possible values of the elements of θ.

We compute the probability density at each point on this triangle which can be accomplished by using the dirichlet distribution. The dirichlet distribution is a probability distribution which defines the probability density for an input vector. This distribution possesses the

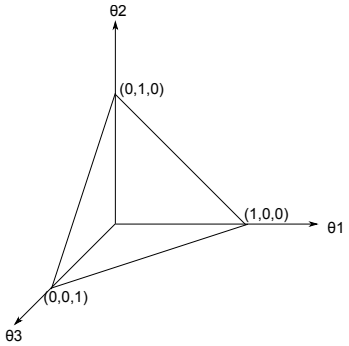


Figure 4.3: The equilateral triangle shows all possible values of θ.

same properties as the multinomial distribution (θ).

The formula for the probability density related to the dirichlet distribution is:

$$Dir(\alpha) \rightarrow p(\theta|\alpha) = \frac{\Gamma(\sum_i \alpha_i)}{\prod_i \Gamma(\alpha_i)} \prod_i \theta_i^{\alpha_i - 1} \tag{4.4}$$

Here, the dirichlet distribution depends on parameter α. We exhibit dirichlet distributions in Figure 4.4 with different values of α in a two-dimensional space. We create the distribution plot using the code [1]. When $\alpha_j < 1$ then the distribution concentrates in the corners and along the borders of the simplex; $\alpha_j = 1$ produces a uniform distribution; $\alpha_j > 1$ the distribution concentrates around the center of the simplex.

In LDA, we have per document topic proportions and the topics as a dirichlet distribution with K and V dimensions, respectively. The dirichlet distribution is parameterized over α as shown in Figure 4.4. It is essential to choose the right value of α, since a high α value in the dirichlet distribution associates many topics to each term, whereas a low α value leads to only a few topics associated to each term.

4.3 Posterior Computation for LDA

We now discuss the problem of computing the conditional distribution of the topic structure given the observed documents (posterior probability). The formula of a posterior probability is shown below:

$$p(\beta_{1:K}, \theta_{1:D}, z_{1:D}|w_{1:D}) = \frac{p(\beta_{1:K}, \theta_{1:D}, z_{1:D}, w_{1:D})}{p(w_{1:D})}. \tag{4.5}$$

In Equation 4.5, the joint distribution of latent variables is the numerator and the marginal probability of the observed documents is the denominator. The marginal probability is the

[1] `https://gist.github.com/tboggs/8778945`

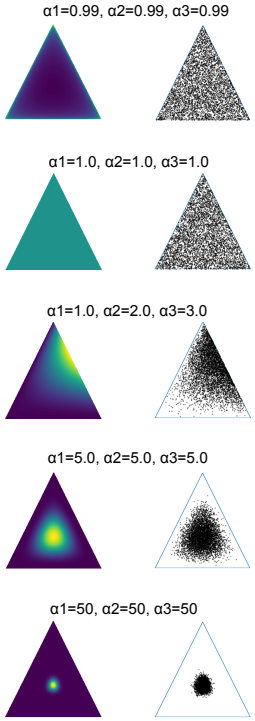


Figure 4.4: The dirichlet distribution plot with different values of α.

sum of the joint distribution over every viable instantiation of the hidden variables or topic structure. The possible number of topic structures is exponentially large; hence it is intractable to compute the marginal probability.

To solve the problem of intractability, topic model algorithms use an approximation of Equation 4.5 for inference. The inference algorithms of topic models fall into two categories: sampling-based algorithms and variational algorithms. In the following sections, we discuss the Gibbs sampling algorithm and the variational inference algorithm.

4.3.1 Gibbs Sampling Algorithm

The gibbs sampling algorithm (Casella & George, 1992) is a sampling-based inference algorithm for LDA. It is a special case of the Metropolis-Hastings algorithm (Chib & Greenberg, 1995). The basic idea behind gibbs sampling is that it is easier to sample from a conditional distribution instead of marginalizing a joint distribution. Let us assume that we need to produce M samples of $Y = (y_1, y_2, \cdots, y_n)$ from the joint distribution $p(y_1, y_2, \cdots, y_n)$. The j^{th} sample of Y is denoted as $Y^j = y_1^j, y_2^j, \cdots, y_n^j$. The steps involved in the gibbs sampling algorithm are as follows:

1 We initialize the j^{th} sample with some value.

2 We need a next sample, i.e., $Y^{j+1} = y_1^{j+1}, y_2^{j+1}, \cdots, y_n^{j+1}$. Since Y^{j+1} is a vector, each element of the vector is sampled using the distribution of that element conditioned over all the sampled elements of the vector. To formulate this, we compute $p(y_i^{j+1}|y_1^{j+1}, y_2^{j+1}, \ldots, y_{i-1}^{j+1}, y_{i+1}^{j}, y_{i+2}^{j}, \ldots, y_n^{j})$.

3 Repeat Step 1 and Step 2 M times.

The initialization of the j^th sample can be accomplished randomly or by using the expectation maximization algorithm (Moon, 1996).

The important characteristics of the samples obtained from the above algorithm are discussed below:

- The samples obtained from the gibbs sampling approximate the joint distribution of all variables.
- The approximation of marginal distributions of a subset of variables can be obtained by only considering them, ignoring the rest of the variables.
- The approximation of the expected value of a variable can be obtained by averaging over all the samples.

4.3.2 Mean-field Algorithm

The mean-field algorithm is a type of variational inference algorithm which approximates the posterior distribution of a latent dirichlet distribution by using a tractable distribution. Here, the tractable distribution considers a set of free variational parameters to compute the distribution.

The variational inference considers the problem of posterior distribution as an optimization problem. It defines the problem as the KL-divergence between the tractable and intractable distribution by making the following assumptions: all the variables are independent, and each variable has its own variational parameter as shown in the plate representation in Figure 4.5.

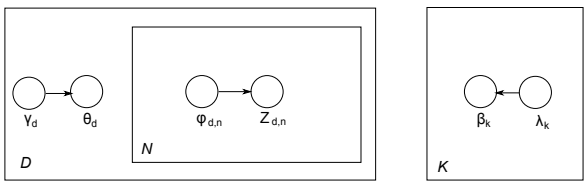


Figure 4.5: The plate representation of variational inference for LDA.

To formulate the variational inference, we first define the variational posterior (Anzai, 2012)

$$q(\theta_{1:D}, z_{1:D,1:N}, \beta_{1:K}) = \prod_{k=1}^{K} q(\beta_k|\lambda_k) \prod_{d=1}^{D} \left(q(\theta_d|\gamma_d) \prod_{n=1}^{N} q(z_{d,n}|\phi_{d,n}) \right) \tag{4.6}$$

The variational inference is accomplished by minimizing the KL-divergence between the variational posterior and the conditional distribution of the latent dirichlet distribution as shown in Equation 4.7.

Objective Function:

$$argmin_{\gamma_{1:D},\lambda_{1:K},\phi_{1:D,1:N}} KL(q(\theta_{1:D}, z_{1:D,1:N}, \beta_{1:K})||p(\theta_{1:D}, z_{1:D,1:N}, \beta_{1:K}|w_{1:D,1:N})) \tag{4.7}$$

The objective function in Equation 4.7 can be written in detail:

$$\mathcal{L} = \sum_{k=1}^{K} E[log(p(\beta_k|\eta))] + \sum_{d=1}^{D} E[log(p(\theta_d|\alpha))] + \sum_{d=1}^{D}\sum_{n=1}^{N} E[log(p(z_{d,n}|\theta_d))] + \sum_{d=1}^{D}\sum_{n=1}^{N} E[log(p(w_{d,n}|z_{d,n}, \beta_{1:K}))] + H(q)$$

The detailed objective function is the sum of the expectations of the logarithmic posterior probabilities and the entropy of q.

Generally, both inference algorithms try to explore the topic structures (Blei & Lafferty, 2009). The observed random variables (collection of documents) remain fixed and serve as a guide. The performance of the inference algorithms depends on the type of topic model being used.

4.4 Assumptions of LDA

LDA is a statistical model which makes assumptions about the collection of documents (Blei & Lafferty, 2009). Many researchers in the field of machine learning are conducting research to relax the assumptions, so that more sophisticated hidden structures are discovered in the documents.

1 One assumption made by latent dirichlet distribution is the "bag of words" assumption, i.e., latent dirichlet distribution does not take into account the order of words in the texts. The order of words does not matter if we want to find the central topics in the document. For more advanced tasks, such as language generation, it is not desirable to neglect the order of words. There are various approaches that extend LDA to relax this "word order" assumption. For instance, Wallach (2006) introduces a topic model that relaxes

this assumption by considering previous words; Griffiths et al. (2004) introduce a topic model that uses LDA and a standard Hidden Markov Model (HMM). These topic models increase significantly the number of parameters, but they improve the performance, too.

2 Another assumption is the order of documents in the collection. LDA does not take into account the order of documents. This assumption is not feasible if we wish to analyse the collection of documents where chronological order matters. In this type of collection, it is possible that topics change over time. Blei & Lafferty (2006) develop a dynamic topic model that considers the order of documents and gives a better topical structure than LDA. A topic in a dynamic topic model is not a single distribution but a sequence of distributions over words (Figure 4.6). Figure 4.6 is taken from the paper on probabilistic topic models (Blei et al., 2003).

3 A third assumption is that the number of topics is known and fixed. Teh et al. (2012) develop a bayesian nonparametric topic model that relaxes this assumption. The number of topics is learned from the collection of documents during inference. This topic model is extended to hierarchies of topics whose structure is learned from the collection of documents (Blei et al., 2010).

4.5 Summary

In this chapter, we have explained in detail the basic topic modelling approach, referred to as latent dirichlet allocation. The notion of the topic modelling approaches is to find the topics in documents, where topics consists of words and their probability distributions. It is difficult to obtain the exact inference for LDA. Hence, LDA utilizes the approximate algorithms for inference. We have provided the detailed description of the approximate inference algorithms. Furthermore, we have explained the dirichlet distribution using a simple example of "Rolling Dice", which is used in LDA to obtain the topic distribution for the documents.

We have pointed out the assumptions made by latent dirichlet allocation. Then, we have provided the brief descriptions of the solutions to these assumptions.

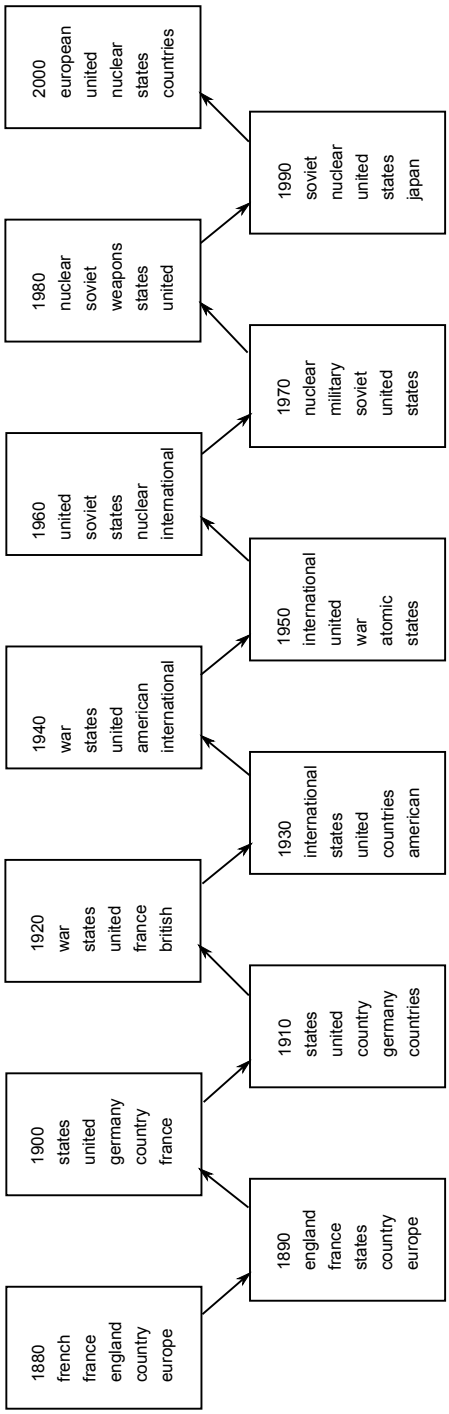


Figure 4.6: A topic from a dynamic topic model.

Chapter 5

Methodology

This chapter describes our approach for summarizing scientific articles. This approach firmly integrates three important factors of summarization: importance, non-redundancy and coherence, in an optimization phase. Figure 5.1 shows the framework of the approach. We describe the subtasks of the approach in the following sections.

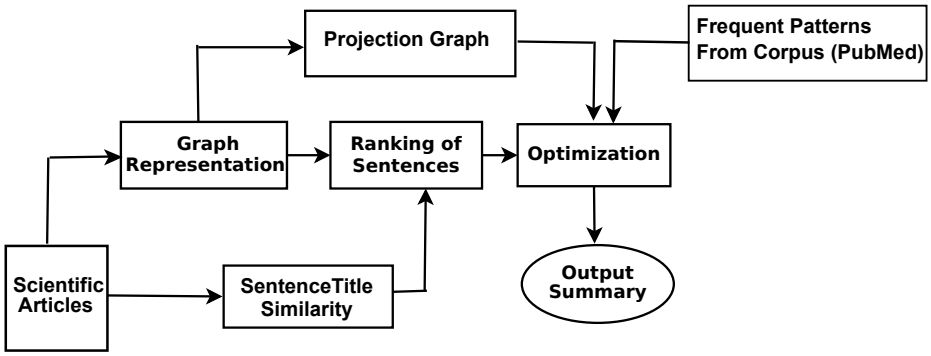


Figure 5.1: An overview of our approach.

5.1 Document Representation

Graphs are generally used as an illustrative representation of documents. The graphical representation of text documents have been widely used in summarization by various methods (Mihalcea & Tarau, 2004; Erkan & Radev, 2004; Parveen & Strube, 2015; Parveen et al., 2015). They are used for various tasks in the field of natural language processing. For example, Guinaudeau & Strube (2013) represent a document graphically to compute the local coherence of documents. Similarly, Mesgar & Strube (2015) utilize graphs for local coherence.

We use two types of graphical representation in our approach: the entity graphs and the topical graphs. We describe entity and topical graphs in Sections 5.1.1 and 5.1.2, respectively.

5.1.1 Entity Graph Representation

Barzilay & Lapata (2008) represent a text document as an entity grid representation to model local coherence. The entity grid representation contains information about entity transitions in a text. The entity grid representation of an example abstract of a scientific article (Figure 5.2, i) is shown in Figure 5.2, ii. In this representation, rows correspond to the sentences and the columns correspond to the entities from the document. A cell in the grid represents the presence or absence of an entity in the corresponding sentence. In addition, the cell contains the syntactic information of an entity in a sentence. The presence of an entity in a sentence is denoted by its syntactic role such as Subject (S), Object (O) and Other (X), whereas the absence of an entity is represented by (-). Guinaudeau & Strube (2013) replace the entity grid representation with the entity graph to compute local coherence. The benefits of using the entity graph is that it solves the problem of sparsity of the entity grid, and the sentences which are not adjacent can be seen connected in the entity graph.

In our approach, we use entity graphs for the representation of scientific articles (see Sections 5.5.1 and 5.5.3). The entity graph contains the information of the distribution of entities across sentences in a text document. This information is used for finding the important and coherent sentences in the document.

The entity graph is a bipartite graph $G = (V_s, V_e, E)$, i.e., it contains two sets of nodes. One set of nodes is entities V_e and the other set of nodes is sentences V_s. The entity graph is a dyadic graph in which same set of nodes are not connected with each other. Edges E are made only between different set of nodes. Figure 5.2, iii shows the entity graph of an example abstract of a scientific article in Figure 5.2, i. Here, entities are denoted by e_i and sentences are denoted by s_j. The abstract contains four sentences which leads to four sentence nodes in the graph. There are 20 entities in the example abstract (Figure 5.2, i) that are represented in bold letters; hence, the entity graph contains 20 entity nodes. The entities are the head nouns of noun phrases of the document.

Barzilay & Lapata (2008) calculate local coherence of a document by utilizing the connections between sentences; two sentences are connected if there is at least one common entity among them. With the same idea, Guinaudeau & Strube (2013) apply one-mode projection on the sentence nodes in the entity graph to model the connections between the sentences to compute local coherence. Similarly, we employ the projection graph to find the coherent sentences in the document (see Section 5.4).

The projection graph is created by performing one-mode projection on the sentence nodes of the entity graph (Figure 5.2, iii). In the projection graph, two sentences are connected if they share at least one entity. As shown in Figure 5.2, iii, s_1 and s_3 are connected because they share entity e_3. The direction in the projection graph encodes the order of sentences present in the original document, i.e., s_1 is connected with s_3 with an outgoing edge from s_1 to s_3. None

of the sentences can have an outgoing edge to any of the preceding sentences, i.e., s_3 cannot have an outgoing edge to s_1.

S_1 **Haemorrhage** is a common **cause** of **death** in trauma **patients**.

S_2 Although **transfusions** are extensively used in the **care** of bleeding trauma **patients**, **there** is **uncertainty** about the **balance** of risks and **benefits** and how this balance depends on the baseline **risk** of **death**.

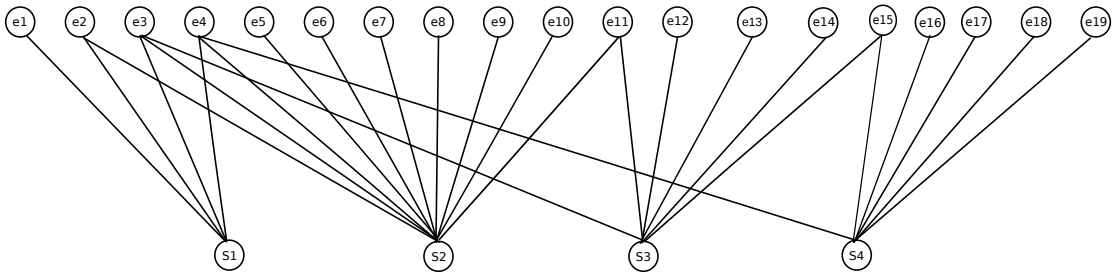
S_3 Our **objective** was to evaluate the **association** of red blood cell (**RBC**) transfusion with **mortality** according to the predicted **risk** of **death**.

S_4 A secondary **analysis** of the CRASH-2 **trial** (which originally evaluated the **effect** of tranexamic **acid** on **mortality** in trauma **patients**) was conducted.

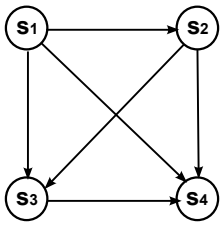
(i)

	HAEMORRHAGE (e_1)	CAUSE (e_2)	DEATH (e_3)	PATIENTS (e_4)	TRANSFUSIONS (e_5)	CARE (e_6)	THERE (e_7)	UNCERTAINTY (e_8)	BALANCE (e_9)	BENEFITS (e_{10})	RISK (e_{11})	OBJECTIVE (e_{12})	ASSOCIATION (e_{13})	RBC (e_{14})	MORTALITY (e_{15})	ANALYSIS (e_{16})	TRIAL (e_{17})	EFFECT (e_{18})	ACID (e_{19})
S_1	S	O	X	X	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
S_2	-	-	X	X	S	X	S	X	S	X	X	-	-	-	-	-	-	-	-
S_3	-	-	X	-	-	-	-	-	-	-	X	S	O	X	X	-	-	-	-
S_4	-	-	-	X	-	-	-	-	-	-	-	-	-	-	X	S	X	O	X

(ii)



(iii)



(iv)

Figure 5.2: An abstract from *PLOS Medicine* (i), entity grid (ii), bipartite entity graph (iii), one-mode projection (iv).

5.1.2 Topical Graph Representation

The topical graph is introduced by ramsl (2015) to compute the local coherence of documents. The topical graph contains topics and sentences as nodes. The benefit of topical graph over the entity graph is that it is a weighted graph and is less sparse. In our approach, we use the topical graph representation for the summarization of scientific articles. The intuition of using the topical graph is that every topic contains various semantically related words which can be connected via a chain. This chain can be considered as a lexical chain. Hence, the topical graph contains the essence of a lexical chain, used for searching important sentences (Barzilay & Elhadad, 1997).

Topical graphs are constructed by using topics extracted from the corpus of text documents (see Chapter 2). The topical graph is built on the top of the topical grid. In Figure 5.3, ii, the topical grid is a grid representation of an abstract of a scientific article, which consists of topics in place of entities. Each cell corresponds to the presence or absence of the topic in a sentence. The weight on each edge, $weight(t_i, s_j)$, is obtained by computing the sum of the logarithmic probabilities of the words present in the corresponding sentence (Equation 5.1). The topics in the topical grid are obtained by applying Latent Dirichlet Allocation (LDA) on the corpus (see Chapter 2). For this, the corpus consists of scientific articles from *PubMed* for the *PLOS Medicine* dataset and *Wikipedia* for the dataset of news articles (see Chapter 6).

$$Weight(t_i, s_j) = \Sigma_{k=1}^{n} log(p(w_k)), \tag{5.1}$$

where, n is the number of words, w_k, that are common in topic t_i and sentence s_j.

The Topical graph is a bipartite graph $G = (V_t, V_s, E)$ representation of documents. It consists of two set of nodes; one set is composed of topics V_t and another set is composed of sentences V_s. An edge is constructed between a topical node and a sentence node, only if the topic contains at least one-word present in the sentence. The edge weight is the corresponding entry in the topical grid representation. The edge weights in the topical graph are used to compute the importance of sentences (Section 5.2). Similar to the entity graph, topical graph is also a dyadic graph, i.e., same set of nodes are not connected with each other.

The one-mode projection is performed on the sentence nodes of the topical graph to prepare a weighted directed one-mode projection graph. The sentence nodes are connected in the weighted one-mode projection graph if they share at least one topic. The weight in the directed one-mode projection graph is the total number of topics shared between two sentences. For example, in Figure 5.3, iv, s_1 and s_2 share six topics hence the edge weight is six. We use the weights of the projection graph to calculate the coherence measure (Section 5.4). The direction in this graph encodes the order of sentences in the input.

S_1 WHO recommends prompt diagnosis and quinine plus clindamycin for treatment of uncomplicated malaria in the first trimester and artemisinin-based combination therapies in subsequent trimesters.

S_2 We undertook a systematic review of women's access to and healthcare provider adherence to WHO case management policy for malaria in pregnant women.

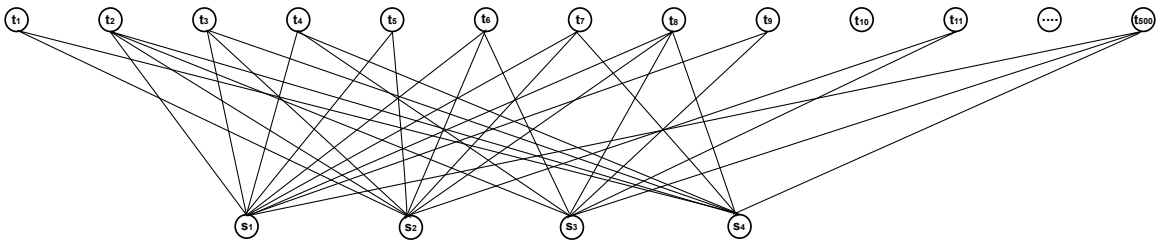
S_3 Data were appraised for quality and content.

S_4 Determinants of women's access and providers' case management practices were extracted and compared across studies.

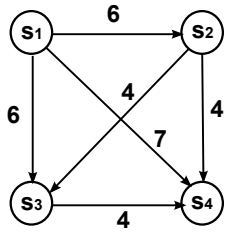
(i)

	$Topic_1$ (t_1)	$Topic_2$ (t_2)	$Topic_3$ (t_3)	$Topic_4$ (t_4)	$Topic_5$ (t_5)	$Topic_6$ (t_6)	$Topic_7$ (t_7)	$Topic_8$ (t_8)	$Topic_9$ (t_9)	$Topic_{10}$ (t_{10})	$Topic_{11}$ (t_{11})	.....	$Topic_{500}$ (t_{500})
S_1	-	0.3	0.46	0.50	0.23	0.55	0.89	0.56	0.09	-	-	.....	0.73
S_2	0.3	0.29	0.3	-	0.66	0.35	0.36	0.63	-	-	0.55	.....	-
S_3	-	0.28	-	0.43	-	0.39	-	0.42	0.65	-	0.12	.....	0.58
S_4	0.22	0.43	0.29	0.6	-	-	0.14	0.26	-	-	-	.....	0.41

(ii)



(iii)



(iv)

Figure 5.3: An abstract from *PLOS Medicine* (i), topical grid (ii), bipartite topical graph (iii), one-mode projection (iv).

5.2 Ranking of Sentences

This section explains the method to compute the importance score of the sentences of an input document. The importance score is computed on the basis of the information conveyed by the sentences. This score is utilized to find the sentences containing important information which is a primary goal of the summarization task.

We represent an input document graphically; the well-suited method to compute the importance of sentences in our approach is a graph-based ranking algorithm, such as PageRank (Page et al., 1998). The ranks obtained after applying the algorithm on a graph of the input document, are considered as the importance of sentences. The sentences with higher ranks are important sentences; however, this does not infer that the higher ranked sentences have to be included in the summary.

We apply the Hyperlink Induced Topic Search (HITS) algorithm (Kleinberg, 1999), also known as the Hubs and Authorities algorithm, to rank sentences in the input document. The HITS algorithm works well on a bipartite graph, to rank web pages, as the well-known PageRank algorithm (Brin & Page, 1998) is not able to handle the nodes with no outgoing edges.

Kleinberg (1999) represents internet pages as a bipartite graph where two sets of nodes are Authorities and Hubs (Figure 5.4). Authority pages are the important pages and hubs are the pages which contain links to the important pages. Authorities and Hubs exhibit a relationship called as a *mutually reinforcing relationship*. The HITS algorithm attempts to obtain the central pages on World Wide Web (WWW). The pseudocode of the HITS is shown in Algorithm 1

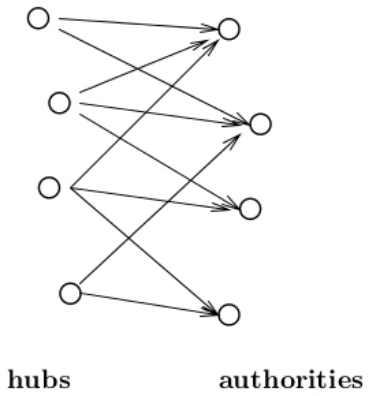


Figure 5.4: The representation of internet pages by Kleinberg (1999)

In our approach, we represent documents as bipartite graphs, i.e., entity graphs (see Section 5.1.1) or topical graphs (see Section 5.1.2). We consider entities as Hub pages and sentences as Authority pages. Then we apply the HITS algorithm on the bipartite graph (entity graphs or

topical graphs). To apply the HITS algorithm, we give initial ranks to the nodes of the bipartite graph. We discuss about the initialization of nodes in Section 5.5

The basic operations of the HITS algorithm are *Authority Update* and *Hub Update*. *Authority Update* is the sum of ranks of Hub pages pointing to the Authority page. For example, in Figure 5.5 (top), the rank of sentence s is the sum of the ranks of entities e_1, e_2 and e_3. *Hub Update* is the sum of ranks of authority pages pointing to, by the Hub page. For example, in Figure 5.5 (bottom), the rank of entity e is the sum of the ranks of sentences s_1, s_2 and s_3.

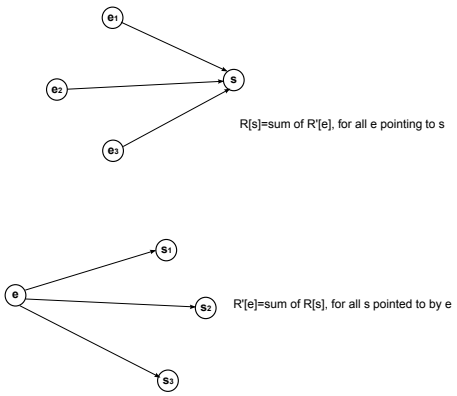


Figure 5.5: Basic Operations of HITS.

5.3 Non-Redundancy

Non-redundancy is one of the important factors in summarization. Any type of summarization technique has to consider non-redundancy as one of the constraints. Non-redundancy in summarization means that any information in the final summary must not repeat itself (Mani, 2001). Consider an example:

s_1 The accident happened on NH-24 is due to the rash driving by Smriti Irani.

s_2 Today's car crash on NH-24 happened due to Smriti Irani.

Here, we need to know that *accident* is synonymous with *car crash*, and also the *accident* in sentence s_1 is referring to *today's car crash* in sentence s_2, then sentence s_1 and sentence s_2 are redundant. We can include either sentence s_1 or sentence s_2 but not both in the final summary.

Non-redundancy can be considered in various manners such as semantically equivalent, string identical, informatically equivalent and informationally subsumes (Mani, 2001).

- Two text units are semantically equivalent if their meanings are exactly the same.

Algorithm 1 Hubs and Authorities Algorithm

G := a graph contain internet pages
for each page p in G **do**
    $authrank_p = 1$ // authrank is the authority rank of the authority page p
    $hubrank_p = 1$ // hubrank is the hub score of the hub page p
function HubsAndAuthorities(G)
for step from 1 to k **do** // run the algorithm for k steps//
    norm = 0
    for each page p in G **do** // update all authority values first//
        $authrank_p = 0$
        for each page q in $InNeighbors_p$ **do** // $InNeighbors_p$ is the set of pages that link to p//
            $authrank_p += hubrank_q$
        $norm += square(authrank_p)$ // to normalize
    norm = sqrt(norm)
    for each page p in G **do** // update the auth scores//
        $authrank_p = authrank_p/norm$ // normalise the auth values
    norm = 0
    for each page p in G **do** // then update all hub values//
        $hubrank_p = 0$
        for each page r in $OutNeighbors_p$ **do** // $OutNeighbors_p$ is the set of pages that p links to//
            $hubrank_p += authrank_r$
        $norm += square(hubrank_p)$ // to normalize
    norm = sqrt(norm)
    for each page p in G **do** // then update all hub values//
        $hubrank_p = hubrank_p/norm$ // normalise the hub values

- Two text units are string identical if they have exactly the same strings.
- Two text units are informatically equivalent if they give exactly the same information.
- A text unit S_a informationally subsumes another text unit S_b if the information given by S_b is present in S_a.

We consider *PLOS Medicine* dataset (Chapter2) for our approach which contains long scientific articles. In this dataset, scientific articles contain various sections which share similar information. Thus, scientific articles have a high possibility of redundant information.

In the entity graph, we incorporate non-redundancy by considering entities. We hypothesize that non-redundancy of the final summary is dependent on the number of unique entities present in it (Galanis et al., 2012; Gorinski & Lapata, 2015). Thus, the more unique entities the final summary contains, the least redundant it will be. For example, the entity graph of an input document as shown in Figure 5.6 contains 20 unique entities, so if the final summary is non-redundant then it should contain all the 20 entities of the input document. We use the non-redundancy measure considering entities.

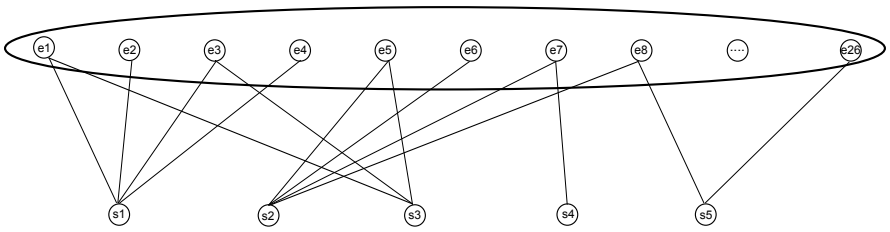


Figure 5.6: Non-redundancy in the entity graph.

In the topical graph, we consider the topics to incorporate non-redundancy in our approach. Similar to the entity graph, we assume that the non-redundancy of the final summary is dependent on the number of unique topics present in it. We denote the number of unique topics as a topical coverage. The more topical coverage a summary has, the less redundant it will be. For example, the topical graph of an input document as shown in figure 5.7 contains 500 topics, in order to be non-redundant, the final summary should contain all the 500 topics of the input document.

5.4 Coherence Measure

In extractive summarization, sentences are extracted from the source document, which can make the summary incoherent. The inherent problem of the extractive summarization technique is that it does not consider the structure of extracted sentences for the final summary.

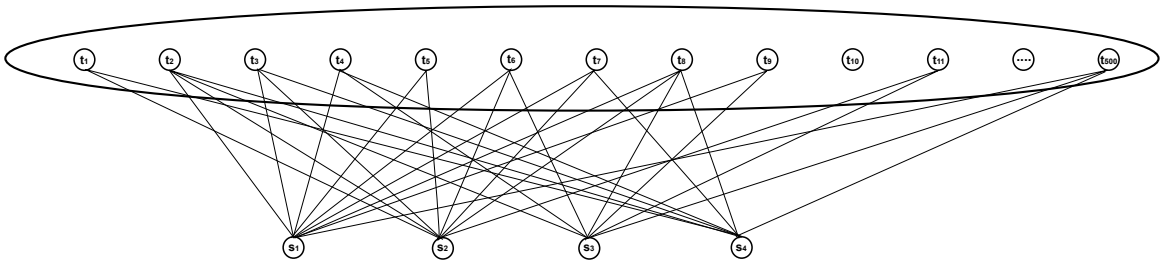


Figure 5.7: Non-redundancy in the topical graph.

The extracted sentences are connected to other sentences via some relation in the source document. If we extract the sentences from the source document then the connectivity information of the sentences may be lost, which can lead to incoherent summaries.

Until now coherence is considered only by few approaches (Abu-Jbara & Radev, 2011; Liakata et al., 2013) in summarizing scientific articles. Abu-Jbara & Radev (2011) consider coherence in a post-processing phase. They select the candidate sentences for the final summary and post-process the sentences to make the final summary readable. This approach does not consider the coherence while summarizing scientific articles. Liakata et al. (2013) incorporate coherence by using discourse information. However, they need a lot of training data for this. Moreover, their approach is not robust.

In this section, we describe an approach to solve the problem of incoherent summaries in extractive summarization of scientific articles. We consider coherence in summarization in two different ways, one is by computing the outdegree of a projection graph (see Section 5.4.1) and other is by taking into account coherence patterns (see Section 5.4.2). We discuss in detail about the coherence patterns in Section 5.4.2.

5.4.1 Coherence Measure Considering Outdegree

Guinaudeau & Strube (2013) introduce the average outdegree of a projection graph as the local coherence measure in Equations 5.2 and 5.3. According to Guinaudeau & Strube (2013), the higher the average outdegree of the projection graph of a text, the more coherent the text is. The average outdegree measures the connectivity of a sentence, in respect of entities, with other sentences of a document.

$$LocalCoherence(T) = AvgOutdegree(P), \tag{5.2}$$

where T is the text and P is the projection graph of T.

$$AvgOutdegree(P) = \frac{1}{N} \sum_{i=1...N} OutDegree(s_i), \tag{5.3}$$

where P is the projection graph of text T. N is the number of sentences in text T. s_i is the i^{th} sentence in text T.

In the entity graph, we consider coherence measure as an outdegree of sentences in the projection graph (Section 5.1) for the summarization of scientific articles. We perform one-mode projection on the sentence nodes of the entity graph (Section 5.1) to create a projection graph. The reason for considering the outdegree of a projection graph as the coherence measure is that, if a sentence with the higher outdegree is included in the final summary, then the probability of its connectivity with other sentences in the summary will increase.

We compute the outdegree of a sentence using an unweighted projection graph as shown in Equations 5.4 and 5.5. In Equation 5.4, we use the positional information of sentences, i.e., higher weight is given to the sentences which are present at the beginning of a document, for instance, $position(s_i)$ will be 1 for the first sentence in a document and 2 for the second sentence in the document, hence the first sentence will have higher weight as compared to the second sentence. In news articles, top few sentences are considered as good candidates for the final summary (Teufel & Moens, 2002). We follow the same notion of using the positional information to calculate the coherence measure. The criterion of positional information is not suitable for summarizing scientific articles because important information is distributed in the article. In Equation 5.5, function $f(s_i)$ is used as the coherence factor in the optimization phase of our approach (Section 5.5).

$$coherence(s_i, P) = \frac{outdegree(s_i, P)}{\sum_{i=1}^{n} outdegree(s_i, P)}, \tag{5.4}$$

where s_i is a sentence in the input document. P is the one-mode projection graph.

$$f(s_i) = \frac{coherence(s_i, P)}{position(s_i)}, \tag{5.5}$$

where $position(s_i)$ gives the positional information of sentence s_i.

In the topical graph, we represent scientific articles using the topical graph (see Section 5.1.2). The topical graph is a weighted graph and is less sparse in comparison to the entity graph. Here, the coherence measure is calculated by a weighted one-mode projection graph. We perform one-mode projection on the sentence nodes of a topical graph (Section 5.1) to create a directed one-mode projection graph. The weight on an edge between the two sentences is the number of common topics shared between them. The edge weights assist in searching for sentences which are strongly connected to each other. An edge with a high weight corresponds to a strong connection between two sentences. For instance, in Figure 5.8, i, the outdegree of sentence s_1 is 3 and sentence s_2 is 3, so here we cannot decide which sentence is better for the summary. However, in case of weighted graph (Figure 5.8, ii), the outdegree of sentence s_1 is higher than sentence s_2, so here we can decide that sentence s_1 should be considered to produce a coherent summary.

Figure 5.8: An example of an unweighted (i) and a weighted projection graph (ii).

We compute the coherence measure by calculating the outdegree using weighted projection graph as shown in Equation 5.6 and Equation 5.7. The weighted outdegree measure is calculated by Equation 5.6. However, this measure is not normalized so we normalize (Mesgar & Strube, 2014) it by Equation 5.7.

$$weighted\_coh(s_i, P_w) = weighted\_Outdegree(s_i, P_w), \tag{5.6}$$

$$norm\_weighted\_coh(s_i, P_w) = \frac{weighted\_coh(s_i, P_w)}{\sum_{i=1}^{n} weighted\_coh(s_i, P_w)}, \tag{5.7}$$

where, s_i is a sentence in the input document. P_w is the weighted one-mode projection graph. n is the total number of sentences in the input document.

Our coherence measures introduced above considers only the single sentence connectivity from the projection graph of an input document, instead of considering the structure of the final summary. Hence, the incorporation of these coherence measures is not sufficient to compute the coherence of the final summary.

5.4.2 Coherence Measure Considering Coherence Patterns

The disadvantage of an outdegree measure as a coherence measure is that it only relies on the single sentence connectivity and does not take care of the final structure of the summary. To overcome this problem, we use coherence patterns in our approach to produce coherent summaries. Coherence patterns can serve as a better feature to compute the coherence of a text document in comparison to the outdegree measure of sentences (Mesgar & Strube, 2015).

We discuss in detail about coherence patterns and introduce them as the coherence measure in this section.

Coherence Patterns The average outdegree as a local coherence measure (Guinaudeau & Strube, 2013) is not sufficient to calculate the local coherence of any text document (Mesgar & Strube, 2015). The major drawback of the average outdegree measure is that it does not take care of the structure of a text document.

The projection graph may consist of connected components which will make it a disconnected graph. The disconnected graph contains at least two nodes, which are not reachable from each other as shown in the left projection graph in Figure 5.9. The projection graph which contains less disconnected components is more coherent (Mesgar & Strube, 2015). The outdegree of a projection graph does not take into account the connected components of the projection graph. For example, in Figure 5.9, the average outdegree of both the projection graphs are equal, whereas the left projection graph contains two components and should be less coherent in comparison to the right projection graph.

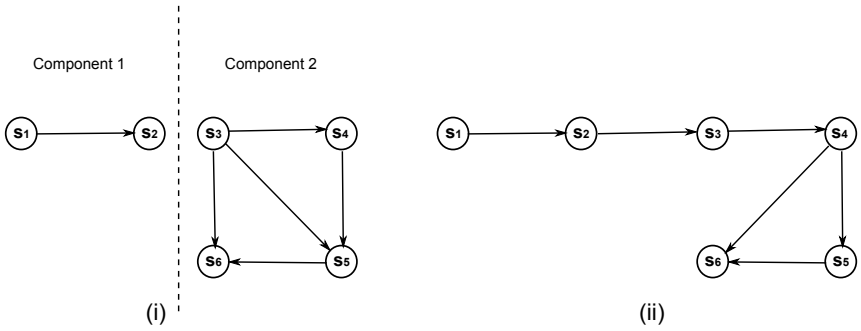


Figure 5.9: Projection graphs from two different texts.

Mesgar & Strube (2015) introduce a novel graph-based approach to assess readability of documents. They outperform various state-of-the-art systems on the Wall Street Journal articles. They show that the coherence of a text document correlates with the coherence patterns. Mesgar & Strube (2015) perform one-mode projection on sentence nodes of the entity graph of a text document. They use one-mode projection graphs to extract frequent subgraphs by using a subgraph mining algorithm [1] . The frequent subgraphs are considered as coherence patterns. They divide subgraphs into two categories: *Basic Subgraphs* and *Frequent Large Subgraphs*. Basic subgraphs have three nodes. There are many possible basic subgraphs but only four basic subgraphs (see Figure 5.10) exist due to lack of backward edges in the projection graphs.

Frequent Large Subgraphs contain more than three nodes. The subgraphs with higher nodes give more information about coherence than Basic Subgraphs (Mesgar & Strube, 2015). There are 24 attainable frequent large subgraphs having 4-nodes (see Figure 5.11).

Mesgar & Strube (2015) use basic subgraphs or frequent subgraphs to calculate the graph signature of a projection graph (see Equations 5.8 and 5.9). The graph signature consists of the relative frequencies of subgraphs in the projection graph. The graph signature is considered as a feature, which consists of connectivity information, for measuring the text coherence of a

[1] A java package for subgraph mining: `http://www.cs.ucsb.edu/~xyan/software/gSpan.htm`

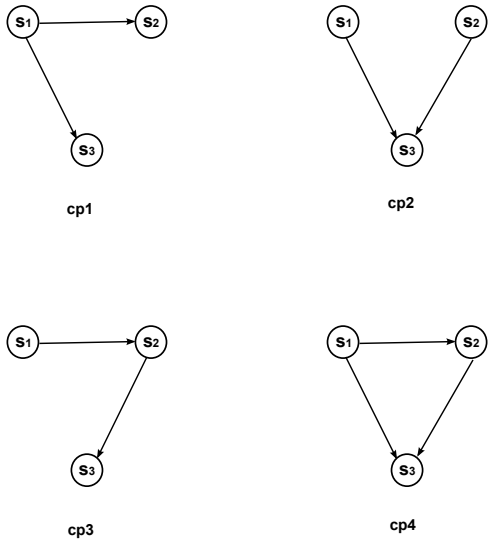
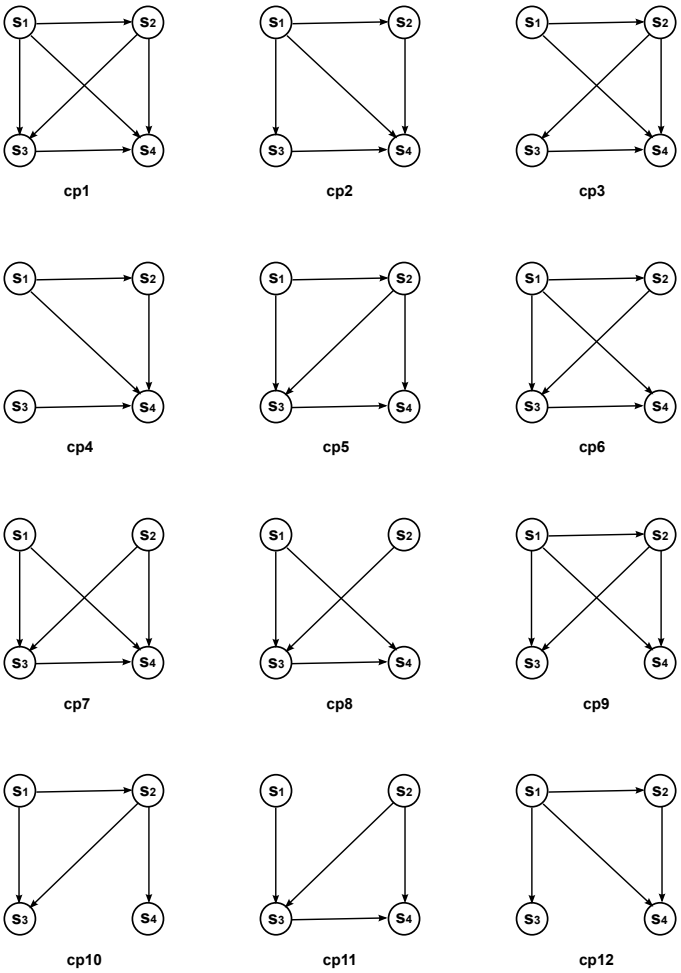


Figure 5.10: Feasible 3-node subgraphs



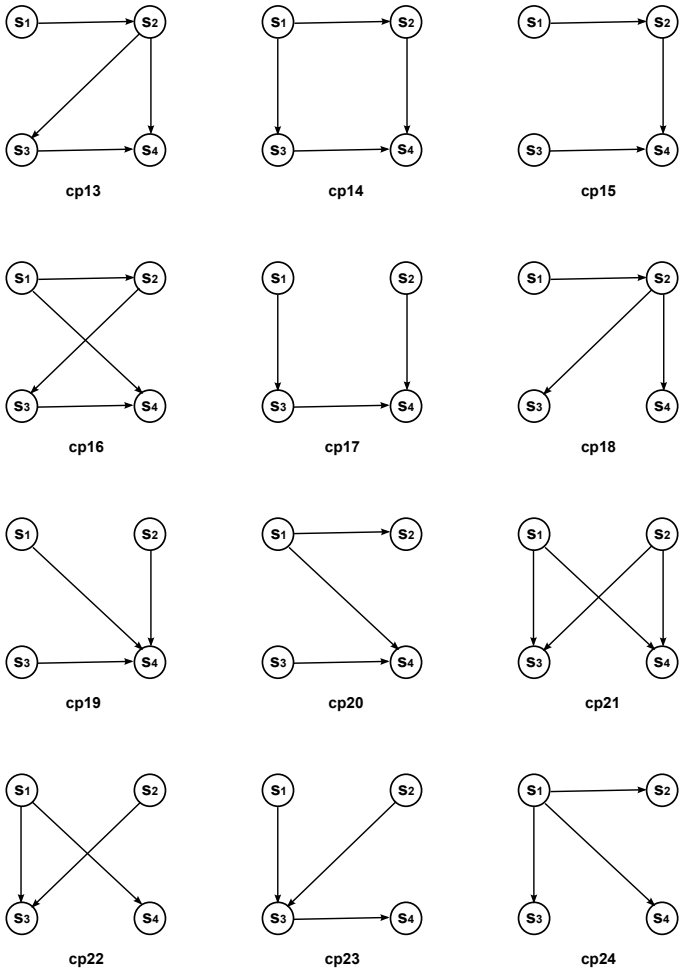


Figure 5.11: Feasible 4-node subgraphs

document.

$$gs(G) = [r\nu(cp_1, G), r\nu(cp_2, G), \ldots, r\nu(cp_m, G)], \tag{5.8}$$

where, $r\nu(cp_m, G)$ is the relative frequency of subgraph cp_m in projection graph G.

$$r\nu(cp_i, G) = \frac{\nu(cp_i, G)}{\sum_{cp_j \in \{cp_1, cp_2, \ldots, cp_m\}} \nu(cp_j, G)}, \tag{5.9}$$

where, $\nu(cp_i, G)$ is the frequency of subgraph cp_i in projection graph G.

Here, subgraphs are always referred to as induced subgraphs. An induced subgraph consists of a subset of vertices and all the edges connecting the vertices in the subset. The example of an induced subgraph is shown in Figure 5.12. The rightmost subgraph is the induced subgraph of the leftmost graph, whereas the middle subgraph is not an induced subgraph due to the missing edge. The edge which makes the rightmost subgraph as an induced subgraph is marked in bold in Figure 5.12.

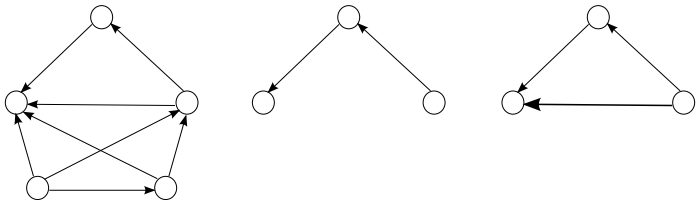


Figure 5.12: An example of an induced subgraph.

Coherence Measure Intuitively, abstracts are coherent summaries of scientific articles as they are written by authors. We hypothesize that the final summary will be coherent if we extract from the input document, only those sentences which contain the coherence patterns existing in the abstracts of the corpus. We use coherence patterns to extract sentences for creating a coherent summary. If we overlay the input document with coherence patterns and extract the sentences which constitute those patterns, then the extracted sentences are already coherent. For example, Figure 5.13 illustrates the extraction of sentences from an input document (Figure 5.13, ii) which constitute a coherence pattern (Figure 5.13, i). This may result in the final summary, consisting of sentences s_1, s_4, and s_7.

The flow diagram to extract coherence patterns is shown in Figure 5.14. We mine 3-nodes coherence patterns and 4-nodes coherence patterns from the abstracts of the corpus [2]. The corpus consists of 700 scientific articles from *PubMed*. We extract abstracts from the corpus and represent them as entity graphs. Then, we perform one-mode projection on the sentence nodes of the entity graph of each abstract. Further, we mine coherence patterns from the directed one-mode projection graphs of the abstracts. We consider the subgraphs of the one-mode projection graph of the abstract as coherence patterns.

We calculate the weight of each coherence pattern based on its frequency in the corpus as shown in Equation 5.10. We further normalize the sum of frequencies of a coherence pattern by its maximum frequency in the corpus.

$$weight(pat_u) = \frac{\sum_{k=1}^{q} freq(pat_u, g_k)}{max_{k=1}^{q} freq(pat_u, g_k)}, \tag{5.10}$$

where pat_u is the coherence pattern. q is the number of projection graphs associated with the abstracts of the scientific articles in the corpus. g_k represents the projection graph of the abstract of k^{th} scientific article in the corpus. The weights of the coherence patterns obtained in Equation 5.10 are not on the same scale. So, we normalize the weights of the patterns using the standard score $\left(\frac{x-\mu}{\sigma}\right)$, where μ is the mean and σ is the standard deviation. We then apply a sigmoid function to obtain the weights in the interval $[0, 1]$. Finally, we use the normalized weights of coherence patterns in the optimization phase (see Section 5.5).

[2] We consider human summaries of DUC 2005 to mine coherence patterns for DUC 2002 (see Chapter 6).

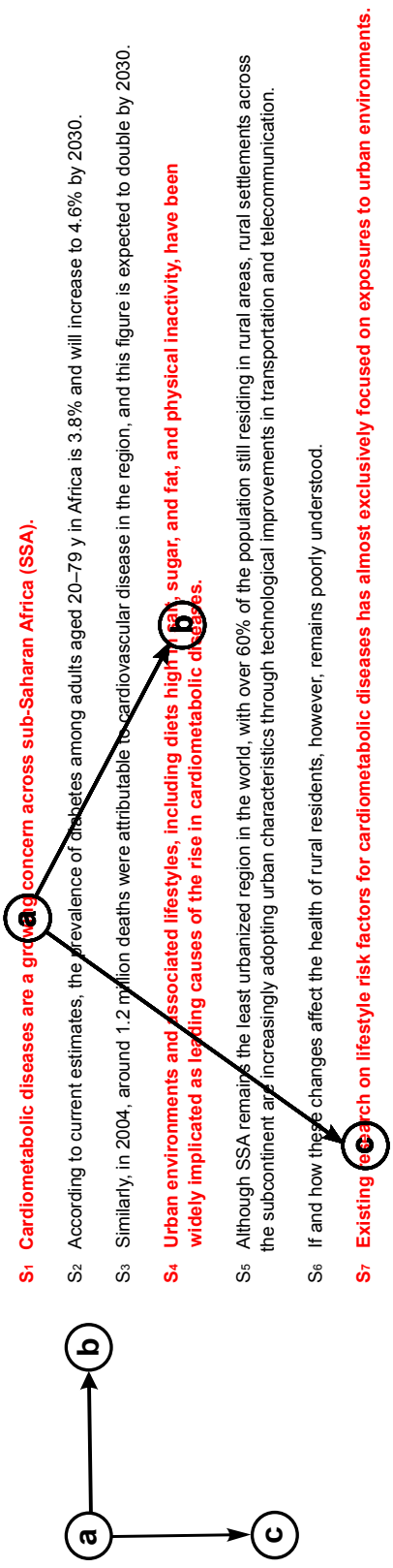


Figure 5.13: An input document is overlaid with the coherence pattern shown in left

5.5 Optimization

Greedy approach has always been a famous approach in summarization until year 2007. Then, McDonald (2007) introduced the global optimization approach for summarization. This tech-

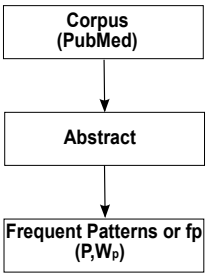


Figure 5.14: The flow diagram to extract coherence patterns.

nique is widely used by many researchers for summarization. Integer Linear Programming (ILP) has been used to obtain globally optimized summaries (Gillick et al., 2009; Nishikawa et al., 2010; Galanis et al., 2012). The problem with a greedy approach is that it always considers the local optimal result which sometimes may not lead to the best summary of a document.

In this section, we introduce the optimization phase of our approaches for the summarization of scientific articles. Since we employ two different document representations (entity graphs and topical graphs), we explain the optimization phase for both of them.

We incorporate coherence using two different approaches in the optimization phase. In the first approach, we maximize the outdegree measures (Section 5.4.1) and in the second approach, we maximize the occurrence of coherence patterns (Section 5.4.2) in the summary. In our approach, we use Mixed Integer programming (MIP) for optimization.

5.5.1 Entity Graph and Outdegree

In this work, we use entity graphs for the representation of scientific articles. The entity graphs are unweighted bipartite graphs, where one set of nodes is entities and another set of nodes is sentences. We explain the entity graphs in detail in Section 5.1.

The objective function in the optimization phase maximizes importance, non-redundancy and coherence, which are the three important factors in summarization. We compute the importance by calculating the ranks of sentences using the Hubs and Authorities algorithm. We discuss about the method to calculate the ranks of sentences in Section 5.2.

We give initial ranks to sentence nodes and entity nodes, as shown in Equations 5.11 and 5.12, to apply ranking algorithm. In the *PLOS Medicine* dataset, the title of a scientific article is self-contained (see Section 2). We use the titles while assigning initial ranks to the sentences of a scientific article. In Equation 5.11, $title$ is the title of the corresponding scientific article, and $sim(sent_i, title)$ is the cosine similarity between $title$ and sentence $sent_i$. In Equation 5.12, $tf(ent_j, article)$ is the term frequency of entity ent_j in the scientific article. $occurrence(e_j, title)$ is a function which returns 1 only if entity ent_j is present in the

title of the scientific article.

$$R[sent_i] = 1 + sim(sent_i, title) \tag{5.11}$$

$$R'[ent_j] = 1 + tf(ent_j, article) + occurrence(ent_j, title) \tag{5.12}$$

Non-redundancy is taken care of by Carbonell & Goldstein (1998) using sentence similarity. Galanis et al. (2012) relate bigram information with non-redundancy. We compute non-redundancy on the basis of entities, which is discussed in Section 5.3. We assume that the summary would be less redundant, if we include more unique entities in the summary.

In the objective function in Equation 5.13, function $f(sent_i)$ corresponds to the coherence measure of the sentence ($sent_i$). We consider coherence measure by calculating the outdegree measure and the positional information of the sentence as shown in Equation 5.5. Here, the outdegree of sentences is calculated using the unweighted projection graph.

Variables:

$$s_i \quad \& \quad e_j$$

Objective function:

$$f(X, Y) = max(\lambda_I \sum_{i=1}^{n} R(sent_i) \cdot s_i + \lambda_C \sum_{i=1}^{n} f(sent_i) \cdot s_i + \lambda_R \sum_{j=1}^{m} e_j) \tag{5.13}$$

where, $s_i \in X$ and $e_j \in Y$. s_i is a binary variable of sentence $sent_i$. e_j is a binary variable of entity ent_j. n is the number of sentences and m is the number of entities. λ_I, λ_C and λ_R are the tuning parameters of importance, coherence and non-redundancy, respectively.

$$\sum_{i=1}^{n} l_i \cdot s_i \leq l_{max}, \tag{5.14}$$

$$\sum_{j \in E_i} e_j \geq |E_i| \cdot s_i \quad \text{for } i = 1, \ldots, n, \tag{5.15}$$

$$\sum_{i \in S_j} s_i \geq e_j \quad \text{for } j = 1, \ldots, m. \tag{5.16}$$

The objective function in Equation 5.13 is subject to constraints 5.14, 5.15, and 5.16. The constraint in Equation 5.14 restricts the length of the final summary. For example, DUC2002 dataset has a length limit of 100 words. In our work, we use the *PLOS Medicine* dataset for the experiments where the length of the final summary is restricted to five sentences or 750 words (see Chapter 6). Here, l_{max} is the length limit of the summary and l_i is the word length of sentence $sent_i$.

The constraint in Equation 5.15 shows that on selection of a sentence from the input document, all the entities present in the sentence must be selected. E_i is the set of entities present

in sentence $sent_i$ and $|E_i|$ is the number of entities in s_i. If sentence $sent_i$ is selected ($s_i = 1$), then all the entities present in the sentence must also be selected, i.e. $\sum_{j \in E_i} e_i = |E_i|$ and the constraint in Equation 5.15 holds. If sentence $sent_i$ is not selected ($s_i = 0$), then some of its entities may still be selected, as it is possible that the entities appear in another selected sentence; therefore $\sum_{j \in E_i} e_i > 0$ and the constraint in Equation 5.15 holds again.

The constraint in Equation 5.16 shows that if an entity is selected then at least one sentence which contains that entity, must be selected. S_j is the set of sentences which contains entity ent_j. If entity ent_j is selected ($e_j = 1$), then at least one sentence consisting of entity ent_j must be selected, i.e. $\sum_{i \in S_j} s_i \geq 1$ and the constraint in Equation 5.16 holds. If entity ent_j is not selected ($e_j = 0$), then none of the sentences consisting of entity ent_j may be selected, i.e. $\sum_{i \in S_j} s_i = 0$ and the constraint in Equation 5.16 holds.

5.5.2 Topical Graph and Weighted Outdegree

In this approach, we use topical graphs to represent scientific articles in which the topics are one set of nodes and the sentences are the other. The topical graph is a weighted graph where the weights are calculated by computing the logarithmic sum of the probability of words present in a topic as well as in a sentence (See Section 5.1). Our intuition of using topical graph is that it increases the information density, by not restricting the transitional conditions from one sentence to another, which assists in the extraction of important sentences for the final summary.

We use integer linear programming to maximize importance, non-redundancy and coherence. The objective function in Equation 5.19, consists of three parts $f_i(X)$ for importance, $f_c(X)$ for coherence and $f_{tc}(T))$ for topical coverage with their respective weights λ. The objective function is subject to constraints in equations 5.20, 5.21, and 5.22.

Function $f_i(X)$, where $f_i(X) = \sum_{i=1}^{n} R(sent_i) \cdot x_i$, is for the importance of sentences. We consider the ranks of sentences as the importance of sentences. For this, we apply Hubs and Authorities algorithm on the weighted topical graph (see Section 5.2). The basic prerequisite for the Hubs and Authorities algorithm is to initialize the nodes in a graph. We initialize topical nodes as shown in Equation 5.17 and sentence nodes as shown in Equation 5.18. In Equation 5.18, $sim(sent_j, title)$ is the cosine similarity between sentence $sent_j$ and the title of the corresponding scientific article.

$f_c(X)$ corresponds to the coherence of a summary. We calculate the coherence measure using the weighted one-mode projection graph. Here, the coherence measure is considered by computing normalized weighted outdegree (Equation 5.7) of a sentence node in a projection graph.

We incorporate non-redundancy by considering the topical coverage which is represented by function $f_{tc}(T)$, where $f_{tc}(T) = \sum_{j=1}^{m} t_j$. We assume that if we include more unique topics or if the summary covers maximum topics from the input document than the summary is less redundant (see Section 5.3).

$$R[topic_i] = 1 \tag{5.17}$$

$$R[sent_j] = 1 + sim(sent_j, title) \tag{5.18}$$

$$Objective\ function : \max_{X,Y}(\lambda_I f_i(X) + \lambda_C f_c(X) + \lambda_R f_{tc}(T)) \tag{5.19}$$

where, X corresponds to the set of binary variables of sentences. T denotes the set of binary variables of topics in the topical graph.

$$\sum_{i=1}^{n} x_i \leq Len(summary) \tag{5.20}$$

$$\sum_{j \in T'_i} t_j \geq |Topics_{x_i}| \cdot x_i, \quad \text{for } i = 1, \ldots, n, \tag{5.21}$$

where, n is the number of sentences in the topical graph.

$$\sum_{i \in S_j} x_i \geq t_j, \quad \text{for } j = 1, \ldots, m, \tag{5.22}$$

where, m is the number of topics in the topical graph.

The constraint in Equation 5.20 is to restrict the length of the final summary. This is similar to the length restriction constraint we have discussed in Section 5.5.1.

The constraint in Equation 5.21 represents that if sentence $sent_i$ is selected ($x_i = 1$) then the topics present in sentence $sent_i$ must also be selected, i.e., $\sum_{j \in T'_i} t_j = |Topics_{x_i}|$ and the constraint holds. $|Topics_{x_i}|$ denotes the number of unique topics present in sentence $sent_i$. If the sentence $sent_i$ is not selected ($x_i = 0$) then it is possible that few topics may still be selected as they can appear in other selected sentences, i.e., $\sum_{j \in T'_i} t_j > 0$ and the constraint holds again.

The constraint in Equation 5.22 represents that if $topic_j$ is selected ($t_j = 1$) then at least one sentence containing j^{th} topic must be selected, i.e., $\sum_{i \in S_j} x_i \geq 1$ and the constraint holds. If $topic_j$ is not selected ($t_j = 0$), then none of the sentences containing j^{th} topic may be selected, i.e., $\sum_{i \in S_j} x_i = 0$ and the constraint holds again.

5.5.3 Entity Graph and Coherence Patterns

In this section, we discuss the method to integrate coherence patterns (Section 5.4.2) for coherence in the optimization phase. In this method we use entity graphs for summarizing scientific articles.

Similar to our previous approach (Section 5.5.1 and Section 5.5.2), we integrate coherence in the optimization phase using quadratic constraint programming; however, we use coherence patterns instead of outdegree measure. We discuss the method to extract coherence patterns from the corpus of scientific articles in Section 5.4.2.

In the optimization phase, we maximize importance, non-redundancy and pattern-based coherence with their respective weights λ. The objective function of our method is shown below:

$$max(\lambda_I f_I(S) + \lambda_R f_R(E) + \lambda_C f_C(P)), \tag{5.23}$$

where S is a set of binary variables for sentences, E is a set of binary variables for entities and P is a set of binary variables for coherence patterns.

The calculations of importance ($f_I(S)$) and non-redundancy ($f_R(E)$) are explained in Section 5.5.1.

On the basis of $f_I(S)$ and $f_R(E)$ we use the optimization constraints in Equations 5.14, 5.15, and 5.16 (see Section 5.5.1).

Coherence ($f_C(P)$): Intuitively abstracts are coherent summaries because they concisely summarize the contents of scientific articles. We assume that the output summary will be coherent if the connections among its sentences are similar to the coherence patterns which occur frequently in all abstracts in the corpus. We define coherence patterns as subgraphs (Mesgar & Strube, 2015) of the projection graphs of abstracts. In Figure 5.15 we overlay the projection graph with the coherence pattern from Figure 5.13, i. This results in three instances of this coherence pattern. However, we select only one since we simultaneously optimize for importance and non-redundancy.

In the objective function, $f_C(P)$ measures the coherence of the summary based on the weights of the coherence patterns occurring in it (Equation 5.24).

$$f_C(P) = \sum_{u=1}^{U} weight(pat_u) \cdot p_u, \tag{5.24}$$

where p_u is a boolean variable associated with coherence pattern pat_u. We discuss the normalized weight of a coherence pattern, $weight(pat_u)$, in Section 5.4.2.

Let $P = \{p_1, p_2, ..., p_u\}$ be the boolean variables of mined coherence patterns. The optimization phase in our method considers the pattern pat_u for summarizing the input article, if pat_u is a subgraph of the projection graph of the article. For finding the coherence pattern in

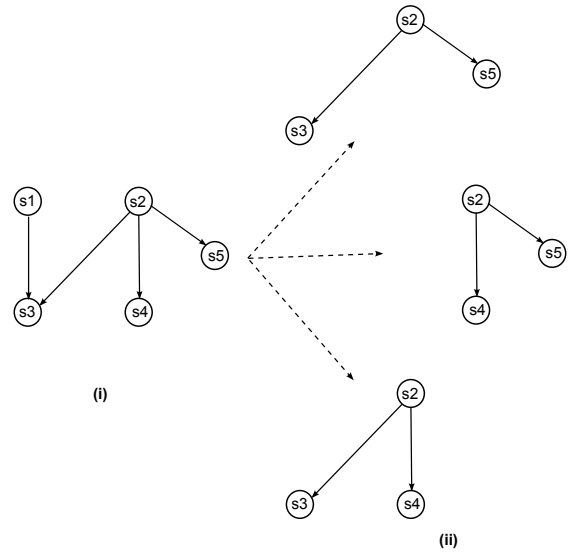


Figure 5.15: (i) A projection graph; (ii) several instances of a coherence pattern in Figure 5.13, i.

a projection graph we follow Lerouge et al. (2015). They use integer linear programming to ensure whether the given graph matches a subgraph of another given graph.

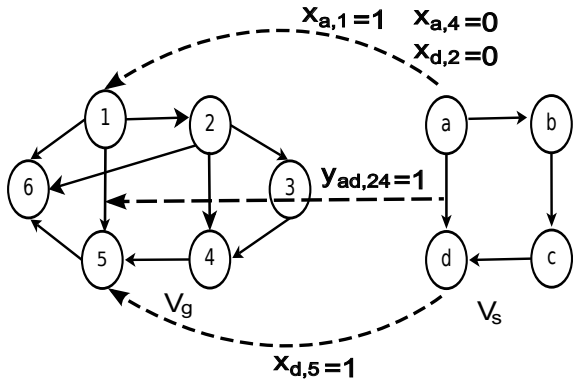


Figure 5.16: An illustration of mapping variables to overlay graph g with coherence pattern pat_u.

In order to model the graph matching problem for our approach between the projection graph $g = (V_g, E_g)$ and patterns $pat_u = (V_{pat_u}, E_{pat_u})$, two kinds of binary variables for mapping are used: $x_{i,k}$ for the node map, and $y_{ij,kl}$ for the edge map. $x_{i,k} = 1$ if vertices $i \in V_{pat_u}$ and $k \in V_g$ match. $y_{ij,kl} = 1$ if for each pair of edges $ij \in E_{pat_u}$ and $kl \in E_g$ match. Figure 5.16 illustrates the usage of these matching variables.

Constraints for graph matching are as follows:

- Every node of the pattern matches at most one unique node of the graph:

$$\sum_{k \in V_g} x_{i,k} \leq 1 \quad \forall i \in V_{pat_u} \tag{5.25}$$

- Every edge of the pattern matches at most one unique edge of the graph:

$$\sum_{kl \in E_g} y_{ij,kl} \leq 1 \quad \forall ij \in E_{pat_u} \tag{5.26}$$

- Every node of the graph matches at most one node of the pattern:

$$\sum_{i \in V_{pat_u}} x_{i,k} \leq 1 \quad \forall k \in V_g \tag{5.27}$$

- A node of pattern pat_u matches a node of graph g if an edge originating from the node of pat_u matches an edge originating from the node of g:

$$\sum_{kl \in E_g} y_{ij,kl} = x_{i,k} \; \forall k \in V_g, \forall ij \in E_{pat_u} \tag{5.28}$$

- A node of pattern pat_u matches a node of graph g if an edge targeting the node of pat_u matches an edge targeting the node of g:

$$\sum_{kl \in E_g} y_{ij,kl} = x_{j,l} \; \forall l \in V_g, \forall ij \in E_{pat_u} \tag{5.29}$$

- We need a constraint to extract *induced patterns*:

$$\sum_{i \in V_{pat_u}} x_{i,k} + \sum_{j \in V_{pat_u}} x_{j,l} - \sum_{ij \in E_{pat_u}} y_{ij,kl} \leq 1 \quad \forall kl \in E_g \tag{5.30}$$

The constraints in Equations $6.32 - 6.37$ are defined to find pattern pat_u in projection graph g of the input article. However, these constraints do not ensure that the pattern is in the summary. For this, we define the constraints in Equations $6.38 - 6.40$ to assure that a pattern is selected, only if there are some sentences in the summary which constitute the pattern.

- The constraint in Equation 6.40 ensures that if sentences s_k and s_l are selected for the summary then the edge between them is selected ($z_{kl} = 1$), too.

$$s_k \cdot s_l = z_{kl} \quad \forall k, l \in V_g \tag{5.31}$$

- Pattern pat_u is present in the summary if and only if one of its instances in the projection graph is included in the summary, i.e., some of the selected sentence nodes must be present in an instance of pattern pat_u. $|V_{pat_u}|$ is the number of nodes in pattern pat_u, and $|E_{pat_u}|$ is the number of edges in pattern pat_u. This constraint is shown below:

$$\sum_{i \in v_{pat_u}} \sum_{k \in v_g} s_k \cdot x_{i,k} + \sum_{ij \in e_{pat_u}} \sum_{kl \in e_g} z_{kl} \cdot y_{ij,kl} = p_u(|V_{pat_u}| + |E_{pat_u}|) \tag{5.32}$$

- If a sentence is selected, then it has to match a node of at least one of the patterns:

$$\sum_{pat_u \in P} \sum_{i \in V_{pat_u}} x_{i,k} \geq s_k \quad \forall k \in V_g \tag{5.33}$$

It is possible that patterns with a large number of nodes are not present in the projection graph. Hence, we consider only basic patterns, i.e. 3-node and frequent patterns, i.e. 4-node patterns, in our approach.

5.6 Summary

In this chapter, we give the detailed description of the proposed approach. This approach attempts to summarize the scientific articles. The summaries produced by this approach consist of important, non-redundant and coherent information.

We explain in detail the new type of graphical document representation for the summarization task. This approach gives the higher score to the important sentences of the document using a graph-based ranking algorithm. Then, in the optimization phase, this approach attempts to maximize the three important factors of summarization.

For coherence, we incorporate structural information in our approach using coherence patterns. The inclusion of coherence patterns in our approach is implemented in collaboration with Mohsen Mesgar.

We uniquely integrate importance, non-redundancy and coherence in the optimization phase. We utilize mixed integer programming for optimization, which provides us the globally optimal solution. Hence, summaries obtained from our approach are best summaries of documents.

Chapter 6

Experiments

In this chapter, we first explain the experimental setup of our approach. We then discuss the tools used for the experiments. Finally, we discuss the results obtained by applying our proposed approach (see Chapter 5) on two different datasets: the *PLOS Medicine* dataset (Chapter 2) and the standard DUC 2002 news dataset. The evaluation is performed by using two different scores: the relevance score and the coherence score.

6.1 Preprocessing

We use the XML formatted scientific articles from *PLOS Medicine* (see Chapter 2). We extract the content of a paper excluding figures, tables and references. Then, editor's summary and authors' abstract are separated from the content for evaluation. The *PLOS Medicine* XML provides explicit full forms for each abbreviation introduced in the text. We replace abbreviations with this full form in the final summary to produce self-contained summaries. We then remove non-alphabetical characters. After this we parse articles using the Stanford parser (Klein & Manning, 2003). We perform pronoun resolution using the coreference resolution system by Martschat (2013). The preprocessing of scientific articles remains the same as above for modelling both entity graphs and topical graphs.

We use some tools for the experiments. We briefly discuss about those tools in the following section. Then, we give details of the experimental setup for our proposed approaches based on entity and topical graphs.

6.2 Tools

In this section, we give a brief description of the tools used in the experiments, which are: a gurobi optimizer, a coreference resolver and a brown coherence tool.

Gurobi Optimizer The Gurobi optimizer is a state-of-the-art mathematical programming solver. It is freely available for the research purpose on `http://www.gurobi.com`. The solver in gurobi exploits multi-processor cores and modern architectures using latest parallel computing algorithms. Gurobi consists of solvers for the following optimization problems:

- Integer Linear Programming
- Mixed-Integer Linear Programming
- Mixed-Integer Quadratic Programming
- Quadratic programming
- Quadratically Constrained Programming
- Mixed-Integer Quadratically Constrained Programming

Gurobi supports various programming and modeling languages to make it usable for people having different knowledge from different domains. We use Gurobi in the optimization phase of our approach.

Coreference Resolver Martschat (2013) introduce a graph-based coreference resolution system that represents documents graphically. In this graph representation, nodes are mentions and edges are relations between mentions. They model documents using multigraphs which allow them to have more than one relation between mentions. Then, they apply graph-clustering to perform coreference resolution. Unlike earlier approaches (Cai & Strube, 2010; Martschat et al., 2012), this system is unsupervised as it does not learn edge weights. This tool is freely available for the research purpose on `http://www.smartschat.de/software/`. We use this system to replace all pronouns in the summary with their antecedents.

Brown Coherence Tool The brown coherence tool (Elsner & Charniak, 2011) is used in our experiment to create entity grids for the document representation. Elsner & Charniak (2011) introduce a tool to build entity grid representations of documents, for the local coherence modeling. The grid models the way texts emphasize important entities. Fundamentally, it predicts that whether an entity will occur in the next sentence given its appearances in the previous sentences. In the experiments by Elsner & Charniak (2011), the entity grid model shows improvements when compared to other systems on the wall street journal dataset. This tool is freely available on `http://www.ling.ohio-state.edu/~melsner/#software`.

6.3 Experimental Setup

In this section, we provide the experimental setup of our approaches based on entity graphs and topical graphs.

6.3.1 Entity Graphs

We represent text documents as entity graphs which consist of two different sets of nodes: entities and sentences. An edge is made between an entity and a sentence, if the entity is present in the sentence. Entity graphs are unweighted bipartite graphs. The details of entity graphs are discussed in Section 5.1.1.

The entity graphs are built on the top of entity grid representations (Barzilay & Lapata, 2008). We apply the Brown coherence toolkit (Elsner & Charniak, 2011) to the articles to convert the document into an entity grid, which then is transformed into a bipartite entity graph (Guinaudeau & Strube, 2013). Entities in the bipartite graph are the head nouns of noun phrases.

6.3.2 Topical Graphs

Another representation that we use is topical graphs. We represent scientific articles of the *PLOS Medicine* dataset and news articles of the DUC 2002 dataset as topical graphs. They are bipartite graphs, which consist of two set of nodes: topics and sentences. Topics are generated by using latent dirichlet allocation (see Chapter 4). If a topic contains at least one word (not including stopwords) of a sentence, then an edge is created between the topic and the sentence in the topical graph. The edge weight is the logarithmic sum of the probabilities of common words present in the topic and the sentence. We discuss in detail about topical graphs in Section 5.1.2.

We use *gensim* to generate topics for topical graphs. We experiment with a varied number of topics like, 500, 1000, 1500 and 2000. Apart from the number of topics, we use default values of parameters in latent dirichlet allocation. For the *PLOS Medicine* dataset, we use the corpus containing bio-medical scientific articles to generate topics for topical graphs. For the DUC 2002 dataset, we use Wikipedia to generate topics for topical graphs. The corpus of bio-medical scientific articles contains $221,385$ documents and about 50 million sentences[1]. We also use Wikipedia to compare with topics from a general domain for *PLOS Medicine*.

[1] `http://www.datawrangling.com/some-datasets-available-on-the-web/`

6.3.3 Coherence Patterns

We incorporate coherence by using coherence patterns in our approach that is based on entity graphs (see Section 5.4.2). We extract frequent coherence patterns from all abstracts in the *PubMed* corpus, and generate summaries of unseen scientific articles of the *PLOS Medicine* dataset (Section 2.2). For DUC 2002 we extract coherence patterns from the human summaries of DUC 2005 (Dang, 2005). We evaluate our model on DUC 2002 to compare with the state-of-the-art systems. We use gSpan (Yan & Han, 2002) to extract all subgraphs from the projection graphs of the abstracts/human summaries. We consider subgraphs as coherence patterns (see Section 5.5.3) in our method. For the evaluation purpose, we use coherence patterns with 3 and 4 nodes, referred to as *Egraph* + CP_3 and *Egraph* + CP_4, respectively.

6.3.4 Optimization

We use Gurobi (Gurobi Optimization, Inc., 2014) to solve the mixed integer programming problem in the optimization phase. The optimization phase returns a binary value associated with each sentence. A sentence is included in the summary if its value is 1.

λ_I, λ_R, and λ_c are the tuning parameters as described in the objective function of the optimization phase (Section 5.5). We determine the best values for λ_I, λ_R, and λ_c on the development sets. $\lambda_I = 0.4$, $\lambda_R = 0.3$, and $\lambda_c = 0.3$ are the best weights for the *PLOS Medicine* development set and for the DUC 2002 development set are $\lambda_I = 0.5$, $\lambda_R = 0.2$ and $\lambda_c = 0.3$.

6.4 Evaluation Metrics

We evaluate our approaches in two ways: relevance evaluation and coherence evaluation. The relevance evaluation is done by using ROUGE scores (Lin, 2004) and the coherence evaluation is accomplished by human judgements.

In this section, we give the detailed description of the metrics employed for the relevance and coherence evaluation.

6.4.1 Relevance Evaluation

In this section, we briefly describe various versions of ROUGE metrics. *ROUGE* is a standard score for evaluation in summarization. ROUGE stands for *"Recall-Oriented Understudy for Gist Evaluation"* (Lin, 2004). It is a measure to determine the quality of summaries generated by computer by comparing them with gold-standard summaries, i.e., human summaries. This measure counts the overlapping between a system-generated summary and a human summary.

It has various versions which depend on the overlapping unit such as n-gram, word sequences and word pairs. In general, summarization systems are evaluated on the basis of three different ROUGE scores i.e. ROUGE-1, ROUGE-2 and ROUGE-SU4.

6.4.2 ROUGE-1

The ROUGE-1 score is a specific version of ROUGE-N, where N stands for the n-gram overlapping recall (Lin, 2004). In ROUGE-1, N is 1, i.e., it gives a unigram overlapping recall score between a computer generated summary and human summaries as shown in Equation 6.1.

$$\text{ROUGE-1} = \frac{\sum_{s \in \{ReferenceSummaries\}} \sum_{Unigram \in S} Count_{match}(Unigram)}{\sum_{s \in \{ReferenceSummaries\}} \sum_{Unigram \in S} Count(Unigram)}. \tag{6.1}$$

Here, reference summaries are human summaries and a candidate summary is a computer generated summary. In Equation 6.1, $Unigram$ is for one word overlapping, $Count_{match}(Unigram)$ is the maximum number of unigrams co-occurring in a candidate summary and a set of reference summaries.

6.4.3 ROUGE-2

The ROUGE-2 score gives the bigram overlapping between reference summaries and a candidate summary as shown in Equation 6.2.

$$\text{ROUGE-2} = \frac{\sum_{s \in \{ReferenceSummaries\}} \sum_{Bigram \in S} Count_{match}(Bigram)}{\sum_{s \in \{ReferenceSummaries\}} \sum_{Bigram \in S} Count(Bigram)} \tag{6.2}$$

Consider the example shown in Figure 6.1 for ROUGE-2 score.

S_1 The thief has stolen gold ornaments.

S_2 Gold ornaments were very costly.

S_3 The thief was caught by the police along with the gold ornaments.

Figure 6.1: A text example.

In the example, S_1 is a reference summary and S_2, and S_3 are candidate summaries. S_2 has only one bigram overlapping with the reference summary. Hence, the ROUGE-2 score is $\frac{1}{5}$. The ROUGE-2 score for S_3 is 0.4 because it has two bigrams overlapping with the reference summary. Hence, based on ROUGE-2 S_3 is a better candidate summary than S_2.

6.4.4 ROUGE-SU4

The ROUGE-SU4 is an extended version of ROUGE-S. ROUGE-S gives the skip-bigram overlapping between a reference summary and a candidate summary. Here, a skip-bigram means a bigram having an arbitrary space between them. In the example shown in Figure 6.1 skip-bigrams in S_1 are (*"The thief", "The has ", "The stolen", "thief has", "thief stolen", "thief gold", "thief ornaments" and so on*). The set of common skip-bigrams between S_1 and S_2 are (*"gold ornaments"*) and between S_1 and S_3 are (*"the thief","gold ornaments", "the gold", "the ornaments", "thief gold" and "thief ornaments"*). The property of ROUGE-S is that, if a candidate summary and a reference summary has some unigrams in common, but no skip-bigrams, then it will give zero as a score. Lin (2004) extends ROUGE-S to ROUGE-SU by adding unigrams, which can be achieved by adding sentence start and end markers to every sentence. ROUGE-SU4 is a variant of ROUGE-SU. ROUGE-SU4 has a fixed length of four in between a bigram.

6.4.5 Coherence Evaluation

ROUGE scores are not able to evaluate system generated summaries on the basis of coherence. Therefore, we evaluate system generated summaries by asking human subjects for the coherence assessment. The details of human coherence assessment are discussed in Section 6.8. In this section, we will discuss the kendall's coefficient of concordance and chi-square test. We utilize the kendall's coefficient of concordance to observe the agreement among the human subjects (see Section 6.8). We also test the significance of the kendall's coefficient of concordance by using chi-square test.

6.4.6 Kendall's Coefficient of Concordance (*W*)

The kendall's coefficient of concordance measures the relation among various rankings of N objects (Siegel & Castellan, 1988). *W* indicates the degree of association among the k sets of rankings of N objects. The utility of the kendall's coefficient of concordance is in interjudge reliability. The formula of *W* is as follows:

$$W = \frac{s}{\frac{1}{12}k^2(N^3-N)},$$

where, s=sum of squares of the observed deviations from the mean of R_j, i.e.,
$s = \sum \left(R_j - \frac{\sum R_j}{N}\right)^2$,
k=number of sets of rankings, i.e., number of judges,
N=number of objects ranked. The interpretation of high or significant *W* is that the judges are using essentially the same level in giving the ranks to the N objects.

6.5 Approaches for the Evaluation

We evaluate the relevance of summaries by using ROUGE scores. This evaluation metric needs at least one gold standard summary or human summary. In our *PLOS Medicine* dataset, we consider editors' summaries as gold standard summaries. We also perform experiments by considering an abstract of a scientific article as a gold summary. In this section, we will discuss about the results obtained after applying our approach discussed in Chapter 5 on the *PLOS Medicine* dataset.

We compare our proposed approaches with six other approaches, i.e., *Lead*, *Random*, *MMR*, *TextRank*, *Mead*, *CoreSc* and the different versions of our approach, i.e., *Egraph + Coh. + Pos.*, *Tgraph (2000) + Coh.*, *Egraph +* CP_3 and *Egraph +* CP_4, on *PLOS Medicine*.

Lead is a baseline approach, which takes top few sentences as the final summary. *Random* is another baseline that extracts sentences randomly from any position in the input document as the final summary. *MMR* is an approach introduced by Carbonell & Goldstein (1998), which uses a tradeoff between relevance and non-redundancy. *TextRank*, introduced by Mihalcea & Tarau (2004), represents documents graphically, which is similar to our proposed approach. Then, they apply *PageRank* (Brin & Page, 1998) on the graph to extract important sentences, i.e., higher ranked sentences, for the final summary. *Mead* employs a linear combination of three features: centroid score, position score and overlap score. The linear combination of the features is used to rank the sentences. The highest ranked sentences are added to the summary up to the required length. The centroid score gives the highest score to the most central sentence in the cluster of sentences, the position score gives a higher score to the sentences which are in the beginning of the document, and the overlap score computes the similarity between the sentences of a document. None of the features take care of the coherence of a summary, as they do not have any notion of the structure of a summary. *CoreSc* is introduced by Liakata et al. (2013), which considers discourse information while summarizing a scientific article (For details see Chapter 7).

Egraph + Coh. + Pos. is one of the variants of our approach in this thesis. It is the state-of-the-art system on the *PLOS Medicine* dataset, where documents are represented as a bipartite graph. Then, sentences are extracted on the basis of importance, non-redundancy and coherence. Another variant is *Tgraph + Coh. + Pos.*, where documents are represented as a weighted bipartite graph that contains topics and sentences. The last variants of our approach are *Egraph +* CP_3 and *Egraph +* CP_4, where documents are represented as entity graphs. Here, we consider coherence using coherence patterns (See Section 5.4.2).

6.6 Evaluation on *PLOS Medicine*

In this section, we show the ROUGE scores obtained by our approach. We begin with the Egraph model, then the Tgraph model and subsequently the structural coherence model. For the ROUGE scores comparison, we generate the summaries with two different length units: 5 sentences and 750 words [2].

We report different versions of ROUGE SU4 and ROUGE 2 scores i.e. Without Stopwords With stem (WO_{Stop} W_{Stem}), With Stopwords With Stem (W_{Stop} W_{Stem}), With Stopwords Without Stem (W_{Stop} W_{Stem}) and Without Stopwords Without Stem (WO_{Stop} WO_{Stem}). Here, without stopwords means without considering stopwords while calculating ROUGE scores and without stem means without applying porter stemmer (Porter, 2001) on summaries while calculating ROUGE scores.

In the results, we show the *Upper Bound* scores that represent maximum ROUGE scores that can be achieved in extractive summarization on the *PLOS Medicine* dataset. It is calculated by considering the whole scientific article as a summary and the corresponding editor's summary as gold standard. The *Upper Bound* scores are not very high, showing that a significant improvement in ROUGE scores on the *PLOS Medicine* dataset is difficult.

We exhibit the ROUGE scores of the approaches by considering editors' summaries and abstracts as gold summaries. However, we are more interested in the results obtained by considering editors' summaries as gold summaries. Since editors' summaries have a broader perspective and unlike abstracts, they are unbiased; we need to produce summaries similar to editors' summaries.

6.6.1 Egraph Model

The egraph model considers entity graphs (See Section 5.1.1) for the representation of the scientific articles. In this approach, we consider outdegree measure for coherence to summarize the scientific articles of *PLOS Medicine*. The approach for the summarization of scientific articles using the egraph model is described in detail in Chapter 5. This approach is referred to as *Egraph + Coh. + Pos.* In this section, we compare the ROUGE scores of the entity graph model with other baseline systems and state-of-the-art systems (See section 6.5).

The results in tables 6.1 and 6.2 are calculated on the basis of 750 words. In Table 6.1, we consider editors' summaries as gold standard summaries. In terms of ROUGE SU4 and ROUGE 2, our approach *Egraph + Coh. + Pos.* performs better than the state-of-the art systems and baseline systems as shown in both tables. *Egraph + Coh. + Pos.* shows better performance in comparison with the recent state-of-the-art system (*CoreSc*). Our approach and *CoreSc* both take care of coherence while summarizing scientific articles. However, *CoreSc* is

[2] average length of editors' summaries in the *PLOS Medicine* dataset

a domain dependent and fully supervised approach.

PLOS Medicine Editors' summaries	WO_{Stop} W_{Stem}	WO_{Stop} WO_{Stem}	W_{Stop} W_{Stem}	W_{Stop} WO_{Stem}	WO_{Stop} W_{Stem}	WO_{Stop} WO_{Stem}	W_{Stop} W_{Stem}	W_{Stop} WO_{Stem}
	ROUGE SU4				ROUGE 2			
Upper Bound	0.423	0.354	0.519	0.470	0.344	0.304	0.430	0.399
Baselines								
Lead	0.191	0.158	0.246	0.222	0.158	0.140	0.185	0.171
Random	0.140	0.113	0.169	0.153	0.102	0.088	0.125	0.116
MMR	0.183	0.149	0.240	0.215	0.141	0.125	0.171	0.157
TextRank	0.148	0.104	0.161	0.159	0.115	0.084	0.126	0.118
State-of-the-art								
Mead	0.197	0.165	0.246	0.222	0.156	0.139	0.186	0.172
CoreSc	0.193	0.153	0.230	0.219	0.135	0.125	0.170	0.149
Our Model								
Egraph + Coh. + Pos.	0.204	0.167	0.254	0.228	0.160	0.145	0.187	0.173

Table 6.1: ROUGE scores on *PLOS Medicine* with **750 words**, editors' summaries.

Furthermore, we analyse the ROUGE scores of the same approaches on the *PLOS Medicine* dataset by considering abstracts as gold standard summaries. In Table 6.2, we exhibit ROUGE SU4 and ROUGE 2 scores. It is evident from both tables that our approach performs significantly better than the state-of-the-art and the baseline systems.

PLOS Medicine Abstracts	WO_{Stop} W_{Stem}	WO_{Stop} WO_{Stem}	W_{Stop} W_{Stem}	W_{Stop} WO_{Stem}	WO_{Stop} W_{Stem}	WO_{Stop} WO_{Stem}	W_{Stop} W_{Stem}	W_{Stop} WO_{Stem}
		ROUGE SU4			ROUGE 2			
Upper Bound	0.563	0.506	0.647	0.607	0.520	0.488	0.619	0.594
Baselines								
Lead	0.260	0.227	0.330	0.300	0.230	0.217	0.277	0.260
Random	0.221	0.189	0.290	0.269	0.187	0.171	0.220	0.213
MMR	0.278	0.240	0.351	0.320	0.240	0.226	0.297	0.270
TextRank	0.196	0.167	0.261	0.236	0.167	0.150	0.210	0.196
State-of-the-art								
Mead	0.261	0.224	0.340	0.311	0.233	0.210	0.281	0.265
CoreSc	0.242	0.214	0.323	0.300	0.213	0.195	0.267	0.243
Our Model								
Egraph + Coh. + Pos.	0.270	0.240	0.350	0.321	0.241	0.224	0.290	0.280

Table 6.2: ROUGE scores on *PLOS Medicine* with **750 words**, abstract.

ROUGE scores obtained by using abstracts are better than the scores obtained by using editors' summaries as gold summaries. This is due to the fact that, abstracts and system generated summaries contain quite a lot of common words, as abstracts are written by authors.

The results in tables 6.3 and 6.4 are calculated on the basis of 5 sentences. In Table 6.3, the ROUGE scores are anticipated by considering editors' summaries as gold standard summaries, whereas the ROUGE scores in Table 6.4 are calculated on the basis of abstracts.

Systems	R-SU4	R-2
Baselines		
Lead	0.067	0.055
Random	0.048	0.031
MMR	0.069	0.048
TextRank	0.068	0.048
State-of-the-art		
Mead	0.084	0.068
CoreSc	0.080	0.065
Our Model		
Egraph + Coh. + Pos.	0.131	0.098

Table 6.3: *PLOS Medicine*, editors' summaries with **5 sentences**

Systems	R-SU4	R-2
Baselines		
Lead	0.105	0.077
Random	0.093	0.589
MMR	0.118	0.098
TextRank	0.134	0.101
State-of-the-art		
Mead	0.178	0.126
CoreSc	0.169	0.120
Our Model		
Egraph + Coh. + Pos.	0.224	0.189

Table 6.4: *PLOS Medicine*, abstracts with **5 sentences**

Here also, our approach performs substantially better than the state-of-the-art and the baseline approaches.

Among the state-of-the-art and the baseline approaches, *CoreSc* is the only approach that considers coherence while generating summaries. In *CoreSc*, coherence is incorporated by employing discourse information (For details see Chapter 7). For this, sentences in a scientific

article, have to be annotated with their discourse tags. This annotation is done automatically by the tool developed by Liakata et al. (2013). Any type of error in the annotation step will propogate to the summary generation step that will eventually produce less relevant summaries. This is one of the reasons that *CoreSc* is not performing better than our approach.

In conclusion, we can say that our approach performs better than other approaches in terms of the ROUGE scores. Our approach is semi-supervised and not particularly domain dependent.

6.6.2 Tgraph Model

The tgraph model employs topical graphs (See Section 5.1.2) to represent scientific articles. We use this graphical representation and weighted outdegree for coherence to summarize scientific articles. The detailed description of the method based on the tgraph model is in Section 5.5.2. This approach is referred to as *Tgraph + Coh.*. In this section, we compare the ROUGE scores of the tgraph model based approach with *Egraph + Coh. + Pos.* as it is the best performing approach discussed in Section 6.6.1.

In tables 6.5 and 6.6, we exhibit the results considering editors' summaries and abstracts as gold standard summaries, respectively. The results show that the tgraph model does not perform better than the egraph model. The reason for this is that topics are learned from the corpus for the topical graph, whereas entities in the entity graph are obtained from the scientific article itself. Hence, entity graphs contain information about the article to be summarised, however topical graphs consist of information from the corpus [3]. The ROUGE scores of the tgraph model, i.e., *Tgraph (n=2000) + Coh.*, are as good as *Mead* and *CoreSc* (see Tables 6.6.1).

PLOS Medicine Editors' summaries	WO_{Stop} W_{Stem}	WO_{Stop} WO_{Stem}	W_{Stop} W_{Stem}	W_{Stop} WO_{Stem}	WO_{Stop} W_{Stem}	WO_{Stop} WO_{Stem}	W_{Stop} W_{Stem}	W_{Stop} WO_{Stem}
		ROUGE SU4			ROUGE 2			
Upper Bound	0.423	0.354	0.519	0.470	0.344	0.304	0.430	0.399
Egraph + Coh. + Pos.	0.204	0.167	0.254	0.228	0.160	0.145	0.187	0.173
Tgraph (n=2000) + Coh.	0.195	0.161	0.231	0.206	0.157	0.140	0.169	0.165

Table 6.5: ROUGE scores on *PLOS Medicine* with **750 words**, editors' summaries.

Furthermore, we analyse the ROUGE scores of the summaries consisting of 5 sentences. The ROUGE SU4 and ROUGE 2 scores of the tgraph model are 0.129 and 0.095 (editors' summaries as gold standard summaries), respectively, and 0.221 and 0.179 (abstracts as gold standard summaries), respectively. In this case also, the tgraph model does not outperforms the egraph model.

[3] This corpus does not contain the scientific article to be summarised

PLOS Medicine Abstract	WO_{Stop} W_{Stem}	WO_{Stop} WO_{Stem}	W_{Stop} W_{Stem}	W_{Stop} WO_{Stem}	WO_{Stop} W_{Stem}	WO_{Stop} WO_{Stem}	W_{Stop} W_{Stem}	W_{Stop} WO_{Stem}
		ROUGE SU4			ROUGE 2			
Upper Bound	0.563	0.506	0.647	0.607	0.520	0.488	0.619	0.594
Egraph + Coh. + Pos.	0.270	0.240	0.350	0.321	0.241	0.224	0.290	0.280
Tgraph (n=2000) + Coh.	0.257	0.239	0.345	0.315	0.232	0.217	0.285	0.270

Table 6.6: ROUGE scores on *PLOS Medicine* with **750 words**, abstract.

Further, we compare the ROUGE scores of *Tgraph + Coh.* using biology journals (Table 6.7) and Wikipedia (Table 6.8) to generate topics for the topical graph. Here, the length of a summary is limited to 5 sentences. The topical graph is denser when we use biology journals to generate topics as compared to the graph generated from Wikipedia. Moreover, the ROUGE scores obtained from biology journals are better compared to the Wikipedia. Hence, we use the topical graph consists of topics generated from the biology journals. In Table 6.7, the highest ROUGE scores are obtained with 2000 topics; however, the differences are negligible for bio topics. Therefore, we use 2000 topics in the tgraph model.

Topics	R-1	R-2	R-SU4
Tgraph (n=500) + Coh.	0.279	0.090	0.125
Tgraph (n=1000) + Coh.	0.289	0.093	0.128
Tgraph (n=2000) + Coh.	0.291	0.095	0.129

Table 6.7: *PLOS Medicine*, editor's summ., Bio topic, 5 sentences

Topics	R-1	R-2	R-SU4
Tgraph (n=500) + Coh.	0.208	0.060	0.098
Tgraph (n=1000) + Coh.	0.258	0.073	0.106
Tgraph (n=2000) + Coh.	0.283	0.086	0.121

Table 6.8: *PLOS Medicine*, editor's summ., Wiki topic, 5 sentences

6.6.3 Structural Coherence Model

The result analysis in the previous two sections show that, the egraph model outperforms baseline and state-of-the-art approaches on *PLOS Medicine*. The egraph model, in Section 6.6.1, considers outdegree as a coherence measure.

The structural coherence model is based on the egraph model, which employs coherence patterns to produce better coherent summaries. In this section, we compare the ROUGE scores of the best performing approach of the previous sections, i.e., the egraph model (*Egraph + Coh. + Pos.*), and the structural coherence model (*Egraph* + CP_3 and *Egraph* + CP_4).

The results in Table 6.9 are obtained by considering editors' summaries as gold standard. *Egraph* + CP_3 and *Egraph* + CP_4 has outperformed significantly as compared to the state-of-the-art systems and baseline systems. The significance test is performed between the state-of-the-art system *Egraph + Coh. + Pos.* and *Egraph* + CP_3 on ROUGE-SU4 and ROUGE-2. The p-value obtained in the significance test is less than 0.05 on ROUGE-SU4, i.e. it is signicant at 95% level. The p-value for the ROUGE-2 score is less than 0.01, i.e. it is significant at 99% level. This shows that improvement in the ROUGE scores of *Egraph* + CP_3 is not incidental. However, *Egraph* + CP_3 and *Egraph* + CP_4 are not significantly different on both the scores.

PLOS Medicine Editors' summaries	WO_{Stop} W_{Stem}	WO_{Stop} WO_{Stem}	W_{Stop} W_{Stem}	W_{Stop} WO_{Stem}	WO_{Stop} W_{Stem}	WO_{Stop} WO_{Stem}	W_{Stop} W_{Stem}	W_{Stop} WO_{Stem}
	ROUGE SU4 (*$p_{value} < 0.05$)				ROUGE 2 (*$p_{value} < 0.01$)			
Upper Bound	0.423	0.354	0.519	0.470	0.344	0.304	0.430	0.399
Egraph + Coh. + Pos.	0.204*	0.167	0.254	0.228	0.160*	0.145	0.187	0.173
Egraph + CP_3	0.215*	0.178	0.268	0.241	0.172*	0.153	0.200	0.184
Egraph + CP_4	0.218	0.179	0.270	0.245	0.175	0.156	0.201	0.187

Table 6.9: ROUGE scores on *PLOS Medicine* with **750 words**, editors' summaries.

In Table 6.10, ROUGE scores are obtained by considering abstracts as gold standard summaries. The results in Table 6.10 show that *Egraph* + CP_3 and *Egraph* + CP_4 outperform significantly the egraph model of Section 6.6.1. The significance test is performed between *Egraph + Coh. + Pos.* and *Egraph* + CP_3 on both ROUGE-SU4 and ROUGE-2. In terms of both the scores, *Egraph* + CP_3 are significantly different from *Egraph + Coh. + Pos.*.

PLOS Medicine Abstract	WO_{Stop} W_{Stem}	WO_{Stop} WO_{Stem}	W_{Stop} W_{Stem}	W_{Stop} WO_{Stem}	WO_{Stop} W_{Stem}	WO_{Stop} WO_{Stem}	W_{Stop} W_{Stem}	W_{Stop} WO_{Stem}
	ROUGE SU4 (*$p_{value} < 0.05$)				ROUGE 2 (*$p_{value} < 0.05$)			
Upper Bound	0.563	0.506	0.647	0.607	0.520	0.488	0.619	0.594
Egraph + Coh. + Pos.	0.270*	0.240	0.350	0.321	0.241*	0.224	0.290	0.280
Egraph + CP_3	0.285*	0.249	0.362	0.333	0.251*	0.232	0.308	0.291
Egraph + CP_4	0.288	0.261	0.372	0.348	0.270	0.252	0.325	0.301

Table 6.10: ROUGE scores on *PLOS Medicine* with **750 words**, abstract.

We further analyse the results of the structural coherence model. For this, we examine the ROUGE scores of the summaries consisting of 5 sentences. Here, we observe that the

ROUGE scores of the egraph model and the structural coherence model are not significantly different. This is due to the fact that the summaries are reasonably small. Hence, for the small summaries, the structural coherence model is as good as the egraph model.

Furthermore, we apply *Egraph* + CP_3 on the dataset introduced by Liakata et al. (2013). The dataset consists of 28 scientific articles from the chemistry domain. The state-of-the-art system on this dataset is *CoreSC*, which is developed by Liakata et al. (2013). *CoreSC* considers discourse information while summarizing a scientific article. The ROUGE-1 score of *Egraph* + CP_3 (0.96) is significantly better than *CoreSC* (0.75) and *Microsoft Office Word 2007 AutoSumarize* (0.73) (García-Hernández et al., 2009), in respect of abstracts. This shows that our system performs well in other domains.

Table 6.11 shows the percentage of the sentences picked from a section for the final summary. *Lead* does not select the sentences from the whole scientific article for the final summary. However, *Egraph* + CP_3, *Egraph* + *Coh.* + *Pos.*, *CoreSc*, *TextRank* and *MMR* consist of sentences from all sections of the scientific article.

Systems	Introduction	Methods	Results	Discussion
Lead	80.59	19.41	0	0
TextRank	25.67	48.21	16.10	10.02
MMR	29.65	35.41	15.49	19.45
CoreSc	30.6	29.4	19.7	20.3
Egraph + Coh. + Pos.	31.49	40.30	15.0	13.21
Egraph + CP_3	32.50	38.5	17.67	11.33

Table 6.11: Sectional Distribution in *PLOS Medicine*

We further calculate the average number of sentences per summary obtained by *Mead* and *Egraph* + CP_3. On average *Mead* produces 17.5 sentences, *CoreSc* produces 24.5 sentences, whereas *Egraph* + CP_3 produces 27.2 sentences per summary. The possibility of longer sentences containing more topic irrelevant entities is higher than shorter sentences (Jin et al., 2010).

6.7 Evaluation on DUC 2002

The DUC 2002 single-document summarization dataset (see Section 2.3) is taken into account to investigate the results of our approach on a different domain (news articles) and with a different length of input documents. Moreover, we want to compare our system with the recent state-of-the-art systems.

Systems	R-1	R-2	R-SU4
Baselines			
Lead	0.459	0.180	0.201
DUC 2002 Best	0.480	0.228	
TextRank	0.470	0.195	0.217
LREG	0.438	0.207	
State-of-the-art			
Mead	0.445	0.200	0.210
ILP_{phrase}	0.454	0.213	
URANK	0.485	0.215	
UniformLink (k = 10)	0.471	0.201	
NN-SE	0.474	0.23	
Our Model			
Egraph + Coh. + Pos.	0.485	0.230	0.253
Tgraph + Coh.	0.481	0.243	0.242
Egraph + CP_3	0.490	0.247	0.258

Table 6.12: ROUGE scores on DUC 2002.

Table 6.12 shows the ROUGE scores of baseline and state-of-the-art approaches on DUC 2002. In this dataset, the required length of the summary is 100 words. We compare our model to previously published state-of-the-art systems on this dataset. These systems show reasonable performance on the DUC 2002 summarization task.

DUC 2002 Best is the result reported by the top performing system at DUC 2002. *LREG* is a baseline system which uses logistic regression and hand-made features (Cheng & Lapata, 2016). ILP_{phrase} is a phrase-based extraction model, which selects important phrases and combines them via integer linear programming (Woodsend & Lapata, 2010). *URANK* utilizes a unified ranking process for single-document and multi-document summarization tasks (Wan, 2010). *UniformLink (k=10)*, considers similar documents for document expansion in the single-document summarization task (Wan & Xiao, 2010). The more recent system, *NN-SE*, utilizes a hierarchical document encoder and an attention-based extractor to extract sentences from a document for a summary (Cheng & Lapata, 2016).

Our models perform better than the state-of-the-art systems and baseline systems as shown in Table 6.12. This is due to the fact that, while summarizing the documents, unlike the state-of-the-art approaches, our models do not employ noisy, external knowledge.

The results show that our models can perform better even in a different genre and with different size of input documents. Hence, our models are robust and scalable.

6.8 Human Coherence Judgement

ROUGE calculates the overlapping recall scores. It does not consider the structure of the summary; hence it cannot evaluate summary coherence. Haghighi & Vanderwende (2009), Celikyilmaz & Hakkani-Tür (2010) and Christensen et al. (2013) evaluate the overall summary quality by asking human subjects to rank system generated summaries. They only consider two candidate summaries for this experiment. We also assess the coherence of summaries by asking human subjects to rank system generated summaries on the basis of their coherence.

We asked five Natural Language Processing researchers (except ourselves) to comparatively rank the output of our system on the basis of coherence. We randomly selected ten scientific articles from *PLOS medicine*. We provide them with the output summaries of different systems for ten scientific articles (*PLOS Medicine*) in an arbitrary order to remove any bias. We used three different systems to generate summaries: *Lead* , *Egraph + Coh. + Pos.* and *TextRank*. Our human judges were asked to assign rank 1 to the best summary, rank 2 to the second best, rank 3 to the worst. By computing the average over the ranks given by all five judges we compute an overall rank: $S1$ gets an overall rank of 1.34, 2 gets 1.82, and $S3$ gets 2.84.

Unsurprisingly *Lead* performed best among the three systems. *Lead* is a baseline system, which consists of sentences from the beginning of an article, hence, they are as coherent as the original authors intended them to be. Still, the difference in average rank between *Lead* and *Egraph + Coh. + Pos.* is not very substantial. In three of our ten documents *Egraph + Coh. + Pos.* was ranked higher than *Lead* on average. The difference between *Egraph + Coh. + Pos.* and *TextRank* however is substantial.

We apply the kendall concordance coefficient (W) (Siegel & Castellan, 1988) to measure whether our human subjects agree in ranking the three systems. With $W = 0.64$ the correlation between the human subjects is relatively high. Applying the χ^2 test shows that W is significant at the 95% level. Hence, we interpret the rankings provided by our human subjects reliable and informative.

Afterwards, using the same setup as discussed above, we assess the coherence of summaries obtained by our system *Tgraph + Coh.*. The other systems participated in this experiment are *Lead* and *TextRank*. The overall rank of *TextRank* is 2.625, *Lead* is 1.675 and *Tgraph + Coh.* is 1.8. We calculated the kendall concordance coefficient (W) (Siegel & Castellan, 1988) to measure the judges' agreement. We obtain $W = 0.61$, which indicates a relatively high agreement between judges.

Our systems *Egraph + Coh. + pos.* and *Tgraph + Coh.* perform reasonably well in the coherence assessment experiments.This experiment shows that it is difficult to perform better than *Lead*. In above coherence assessment experiments, we do not consider our best performing system, *Egraph +* CP_3, in terms of ROUGE scores.

We use the same setup for the coherence assessment of the summaries obtained by *Egraph* + CP_3. We do not consider the summaries obtained by *Egraph* + CP_4 as there is no significant difference between *Egraph* + CP_3 and *Egraph* + CP_4 in terms of ROUGE scores. In this experiment, human subjects assess four systems: *Egraph* + CP_3, *Egraph* + *Coh.* + *Pos.*, *TextRank*, and *Lead*. The reason for using only these systems are:

- *Egraph* + CP_3 is similar to *Egraph* + *Coh.* + *Pos.* in terms of representation, importance, and non-redundancy.
- *Lead* is the best system in both experiments discussed above.
- *TextRank* is the baseline system.

We compute the overall average rank of a system given by the human subjects (Table 6.13).

PLOS Medicine System	Average Human Score
TextRank	3.950
Egraph + Coh. + Pos.	2.325
Egraph + CP_3	1.875
Lead	1.625

Table 6.13: The lower the value of average human scores the more coherent the summary.

As expected, *Lead* gets the best overall average rank from the human subjects. However, *Egraph* + CP_3 is close to *Lead* in terms of coherence indicating that our strategy is successful. It also performs substantially better than *TextRank* and *Egraph* + *Coh.* + *Pos.* This confirms that using coherence patterns for sentence extraction yields better coherent summaries.

We apply the kendall concordance coefficient (W) (Siegel & Castellan, 1988) to measure whether the human judges agree in ranking the four systems. With $W = 0.6725$ the correlation between the human judges is high indicating a high level of agreement. Applying the χ^2 test shows that W is significant at least at the 95% level. Hence, the rankings provided by the human judges are reliable and informative.

In appendix A, we show the summaries generated by *Egraph*+CP_3 referred to as S_1 , *Lead* referred to as S_2 and *TextRank* referred to as S_3. Summary S_1 contain small sentences as compared to summary S_2, therefore summary S_1 contain more sentences as compared to summary S_2. Although, summary S_2 is coherent as compared to summary S_1 and S_3 but it does not contain relevant information. According to human judges, our approach produces coherent summaries as compared to *TextRank*, therefore summary S_1 is better than summary S_3 in terms of coherence. Here, we can conclude that our approach produces relevant and coherent summaries.

6.9 Summary

In this chapter, we exhibit the results obtained by applying the proposed method on two datasets: *PLOS Medicine* and DUC 2002. We use two types of evaluation measures: ROUGE scores and human coherence assessment scores. Our proposed method with coherence patterns outperforms in both evaluation techniques. This shows that the summaries obtained from our approach contain relevant and coherent information. Relevance of the summaries is evaluated by using ROUGE scores and coherence by using human coherence assessment scores.

Further, our approach with coherence patterns outperforms when we use the DUC 2002 dataset. This shows that our approach is scalable and robust. Scalable because it performs effectively even with short text documents and robust as it shows substantially better results with the documents of different genre.

Chapter 7

Related Work

Summarization has always been a topic of interest for researchers in the field of natural language processing. Several approaches have been developed for the summarization task. These approaches vary on the basis of the type of summaries (generic/query-based), the type of input documents (single/multiple documents) or the type of genre of input documents (news/scientific articles).

Three broad approaches can be identified in summarization:

- Shallow Approaches: Summarization systems usually analyse the sentences of a text upto the syntactic level and do not go beyond it. This type of approach is usually applied in extractive summarization. An extractive summarization system extracts sentences as they are present in a source text. Sometimes these sentences are out of context. To solve this problem, the extracted sentences are arranged in a coherent way. Further, these extracted sentences are synthesized by using smoothing and pronoun resolution. The synthesis phase presents the sentences as a final summary in an effective manner. Although, this approach does not contain any notion of meaning of the sentence, but it is a very robust approach.

- Deeper Approaches: Summarization systems that go into the details of a text. They usually generate abstractive summaries, which involves natural language generation using semantic or discourse level representation. Since the summary is not produced by extraction, the system could generate more coherent and appropriate texts by applying several rules or machine learning techniques.

- Hybrid Approaches: This approach is commonly used in summarization. It is the fusion of previous two approaches. This approach can be carried out without analysing the deeper level of semantics of a text document. This is more suitable for multi-document summarization, where sentences are taken from different documents to produce compact and coherent text.

In this chapter, we discuss various works related to shallow and hybrid approach in detail. First, we describe various earlier approaches for summarization in detail. Then, we discuss the corpus-based approaches, discourse-based approaches, optimization-based approaches, and neural networks-based approaches.

7.1 Classical Approaches

This section describes the work done 55 years back. They are classical as they give the fundamental basis for the summarization task. These approaches utilize the surface-level features. Similar to these approaches, our proposed approach utilizes some surface-level features.

7.1.1 Luhn (1958)

Luhn (1958) introduces an approach to assign scores to sentences in a document. According to him, if the sentences achieve a higher score, they are considered important. First, he measures the importance of a word by calculating its frequency in the document. The intuition behind this is that a writer uses the important words frequently in the document. The importance of a sentence is calculated on the basis of salient words appearing in the sentence, and the position of the sentence in the document. Then, the sentences which scored highest are extracted from the document for the summary.

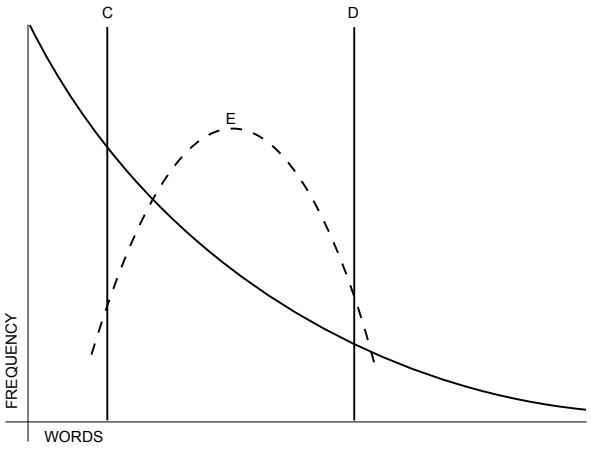


Figure 7.1: A word frequency plot.

The plot between words and their frequencies is shown in Figure 7.1. The words appearing in the highest frequency region are too common to be considered as important words for the summary. These words can be easily eliminated by comparing them with a common word list. Another way to eliminate these words is to create a threshold by establishing a high-frequency cutoff. In Figure 7.1, line "C" represents the high frequency cutoff; words appearing to its

right are considered as significant words. Line "D" in Figure 7.1 represents the low frequency cutoff, i.e., the words which appear rarely and do not have discriminatory power. Curve "E" represents the degree of discrimination of words falling between line "C" and line "D".

7.1.2 Edmundson (1969)

Edmundson (1969) introduces a summarization approach which not only considers high-frequency words but also some additional factors to summarize an article. There are three additional factors taken care of by Edmundson (1969): cue words or pragmatic words, title or heading words, and structural indicators such as positional information. The initial weights of the four factors are obtained from the corpus of 200 scientific articles. The significance score of a sentence is calculated as the linear sum of the four factors.

$$score(s_i) = w_1 H + w_2 C + w_3 T + w_4 P, \tag{7.1}$$

where H is for high-frequency words in sentence s_i, i denotes the index of a sentence, C is for cue phrases in sentence s_i, T is for title words in sentence s_i and P is for positional information of sentence s_i. w_1, w_2, w_3 and w_4 are the weights of the factors obtained from the corpus.

7.1.3 Pollock & Zamora (1975)

The approach by Pollock & Zamora (1975) has been developed for the summarization of scientific articles of chemistry. This approach is known as the Automatic Document Abstracting Method (ADAM). Pollock & Zamora (1975) show that some subject areas are more flexible than others for automatic text summarization. They indicate that including more domain knowledge while summarizing an article would lead to a better summary.

In *ADAM*, sentence selection is based on cue words, word frequency and title words. Pollock & Zamora (1975) created a Word Control List (WCL) of cue words in the field of chemistry.

ADAM creates an indicative summary of an article which assists the reader to determine whether he should read the original document or not. It mainly focuses on the rejection of sentences and has following criteria:

- **WCL**: WCL contains an ordered set of words and phrases and their two corresponding, syntactic and semantic codes. If a word having a negative code appears in a sentence, then the sentence should be rejected. For instance, cue words like *previous work* and *obvious* have negative codes whereas *this study* and *present work* have positive codes.

- **IR**: Intersentence Reference (IR), indicates that, if a sentence is rejected then other sentences referring to that sentence should be rejected.

7.1.4 Carbonell & Goldstein (1998)

Carbonell & Goldstein (1998) introduce a method to extract novel information from the document that are non-redundant and relevant to a query using the Maximal Marginal Relevance (MMR) criterion. Carbonell & Goldstein (1998) formulate the MMR criterion as follows:

$$MMR = \text{Arg max}_{S_i \in R \setminus C}[\lambda(Sim_1(S_i, Q)) - (1 - \lambda) max_{S_j \in C}(Sim_2(S_i, S_j))], \quad (7.2)$$

where, $Sim_1(S_i, Q)$ measures the similarity between sentence S_i and query Q. $Sim_2(S_i, S_j)$ measures the similarity between sentences S_i and S_j. λ is a tuning parameter. C denotes the final summary and R denotes the input text document. $R \setminus C$ denotes the set of unselected sentences in input document R.

Equation 7.2 consists of two factors: the measurement of the query relevance of the sentences which are not included in the summary and the measurement of non-redundancy of the summary by minimizing the similarity between a sentence of the input document and the sentences of the final summary.

This approach, in contrast to our approach, utilizes the greedy algorithm. It selects the local optimal solution which is not always the best solution.

7.1.5 Gong & Liu (2001)

Gong & Liu (2001) propose two types of generic summarization methods: first, using standard information retrieval methods for sentence extraction, i.e., summarization by relevance and second, using Latent Semantic Analysis (LSA) (Dumais, 2004) to obtain semantically relevant sentences, i.e., summarization by LSA.

Both type of summarization methods acquires the weighted term-frequency vector representation for each sentence in the document. The weighted term-frequency vector $W_i = [w_{1i}, w_{2i}, \cdots, w_{ni}]^T$ of sentence i is: $w_{ji} = local(a_{ji}) \cdot global(a_{ji})$, where $local$ is the frequency of term a_{ji} in sentence i and $global$ is the frequency of term a_{ji} in the document.

The relevance score of a sentence is the dot product between the term-frequency vector of the sentence and the weighted term-frequency vector of the document. The sentence with the highest relevance score is extracted for the summary. Then, the terms of the extracted sentence are eliminated from the document to avoid redundancy in the summary. Further, the weighted term-frequency vectors of the remaining sentences are calculated again. Then, second highest relevance score sentence is extracted for the summary. This process continues until the summary length reaches a predefined limit.

In summarization by LSA, Gong & Liu (2001) perform singular value decomposition (SVD) (De Lathauwer et al., 1994) on a $m \times n$ matrix D for the document, where m is

the number of terms, and n is the number of sentences. Each sentence i in the singular vector space (De Lathauwer et al., 1994) is represented by the column vector $v_i = [\nu_{i1}\nu_{i2}\cdots\nu_{ir}]$ of V^T. The sentence with the highest index value in the k^{th} right singular vector of V^T is selected for the summary, where k is initialized with 1 and incremented by one after every iteration.

7.1.6 Radev et al. (2004b)

Radev et al. (2004b) develop a centroid-based approach for summarization referred to as *MEAD*. This approach utilizes centroids of clusters, which are important to all documents in a corpus.

Radev et al. (2004b) introduce two measures in their approach: *cluster-based sentence utility* (CBSU) and *cross-sentence informational subsumption* (CSIS). CBSU assigns scores to the sentences, on the basis of the degree of relevance (0 to 10) to the topic of the cluster. The lowest CBSU score denotes that the sentence is not relevant to the cluster, while the highest denotes that the sentence is essential to the cluster. CSIS exhibits that some sentences in a document repeat the information present in other sentences. CSIS is utilized for reducing redundancy in the summary.

MEAD gives the score to the sentences of a document based on several features listed below:

- **Centroid**: Cosine similarity between the sentence and the centroid of a cluster.
- **SimWithFirst**: Cosine similarity between the sentence and the first sentence of a document.
- **Length**: If the length of the sentence is greater than a threshold, the value of this feature is 1, else it is 0.
- **RealLength**: Number of words in the sentence.
- **Position**: Position of the sentence in the document.
- **KeyWordMatch**: String match of words in the sentence with any keyword in a predefined list of keywords.
- **LexPageRank**: PageRank of the sentence.

MEAD selects highest scored non-redundant sentences for the summary. Non-redundancy is taken care of by calculating the word overlap between two sentences.

7.2 Corpus-based Approaches

In this section, we discuss about the approaches that use corpus to compute term statistics and to determine the relevant features for the summarization task. These approaches also belong to the category of surface-level approach. In our approach, we also use the corpus to compute the statistics of subgraphs (see Section 5.4.2).

7.2.1 Kupiec et al. (1995)

Kupiec et al. (1995) consider summarization as a classification problem. They calculate the probability of a given sentence to determine whether the sentence should be included in the summary. The sentences are ranked on the basis of their probability, and the top ranked sentences are selected for the summary. The probability of a sentence is calculated by using Baye's rule as follows:

$P(s \in S|F_1, F_2, \cdots F_k) = \frac{P(F_1,F_2,\cdots F_k|s\in S)P(s\in S)}{P(F_1,F_2,\cdots F_k)}$,

where S is a summary, s is a sentence, and F_k denotes a k^{th} feature.

This approach needs a training corpus with labelled sentences, which is expensive to acquire. Kupiec et al. (1995) obtained the corpus from *Engineering information Co.* which produces the abstracts of scientific articles.

The features employed by Kupiec et al. (1995) for the classification task are:

- **Sentence Length**: Sentences shorter than a certain threshold should not be included in the summary. This feature is "true" if the sentence length is greater than the threshold, else it is "false".

- **Fixed-Phrase**: Kupiec et al. (1995) compile a list of 26 indicator phrases, for example, "this letter...", "In conclusion..." etc. This feature is true if a sentence contains at least one of the indicator phrases.

- **Paragraph**: In this feature sentences are distinguished on the basis of their occurrence in a paragraph. The sentences appearing at the start, middle and end of a paragraph are referred to as *paragraph-initial*, *paragraph-final*, and *paragraph-medial*, respectively.

- **Thematic Word**: Thematic words are the most frequent content words. First, the score of a sentence is computed on the basis of thematic word frequencies. Then, the value of this feature is calculated. This feature is true if a sentence has a high score.

- **Uppercase Word**: In general, proper names are important entities appearing in the uppercase format in a document. This feature exploits this property and computes the value with the following constraints on the uppercase words:

- They should not appear at the start of a sentence,
- They should occur frequently,
- and must not be an abbreviation of measurements.

7.2.2 Myaeng & Jang (1999)

Myaeng & Jang (1999) develop an automatic summarization approach that considers lexical and statistical information computed from a corpus. The approach is divided into two sections, the training section and the summary generation section.

Similar to the previous work, sentence selection is based on the probability of a sentence, which is further based on the appearance of certain features that are likely to be present in a summary. The features used in this approach are position, keywords, centrality, title similarity and text component. The frequencies of these features are obtained from the training corpus.

A probability value is calculated separately for every feature. The final probability of a sentence is computed by the Dampster-Shafer combination rule that combines the probability values obtained from each feature.

In the final step, redundant sentences are eliminated from the set of candidate sentences. The redundancy of sentences is taken care of by calculating the similarity between all the pairs of sentences in the candidate set. If the similarity score of two sentences is greater than a certain threshold, then the sentence with the lower rank is eliminated.

7.2.3 Aone et al. (1999)

DimSum is a system developed by Aone et al. (1999), which consists of two components: *Summarization server* and *Summarization client*. *Summarization server* is a feature extractor, whereas *Summarization client* is a feature combiner. This approach does not only consider frequency based features but also linguistic features in order to obtain the domain and the structural information.

Aone et al. (1999) create a baseline database by using a name tagger. This database contains names of people, entities, and places. The database considers multi-word names like "Barack Obama" and simultaneously attempts to disambiguate the semantic types of names by considering contextual words, for instance, the word "Bill" is used separately from "Bill Clinton".

Further, Aone et al. (1999) calculate a feature using domain knowledge by considering a large corpus of the same domain. Then, another feature is computed by considering the discourse structure by identifying the relationships (name aliases, morphological variation and synonyms (Halliday & Hasan, 1976)) between sentences.

In this approach, the sentences are extracted on the basis of scores computed from different combinations of features. The features are merged by using two combiners: the batch feature combiner and the trainable feature combiner. In the batch feature combiner, Aone et al. (1999) experiment with different combinations of features and select the best combination to extract sentences for a summary. The trainable feature combiner is developed by using Baye's rule, where *DimSum* probabilistically learns the best combination of features from the human extracted summaries.

7.2.4 Lin & Hovy (2000)

SUMMARIST provides extractive summaries using robust NLP techniques with world knowledge. Lin & Hovy (2000) divide summarization into three different tasks: topic identification, interpretation and generation. The flow diagram of SUMMARIST is shown in Figure 7.2.

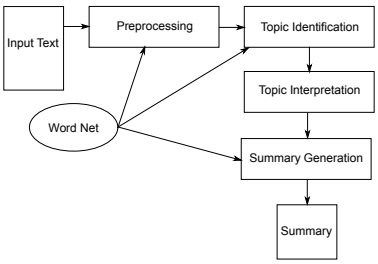


Figure 7.2: A flow diagram of SUMMARIST.

The identification of topics is based on the position of sentences (Luhn, 1958; Edmundson, 1969), cue phrases (Baxendale, 1958), frequency of words and discourse segmentation (Marcu, 1997b). Further, the topic interpretation is accomplished by combining topics, which leads to a topic consisting of similar words. Then, *SUMMARIST* extracts the sentences containing the topic words for the extractive summaries.

7.2.5 Reeve et al. (2007)

Reeve et al. (2007) introduce a method to summarize biomedical texts using the concept of lexical chains, referred to as *BioChainSumm*. They use domain specific concepts to obtain important sentences for the summary. This approach uses the UMLS MetaMap Transfer application (United States National Library of Medicine, 2005b) to identify the concepts in the source texts.

The concepts are discovered from the corpus. Then, these are used to identify the concepts in the source texts. Further, the strongest concept chain is identified by using the scoring function (Equation 7.3). The sentences are extracted for the summary on the basis of the

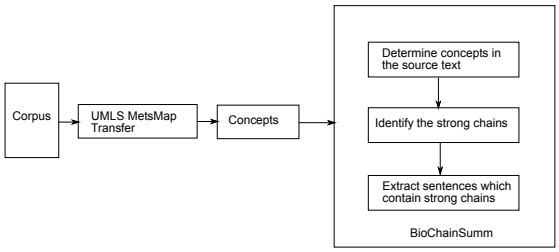


Figure 7.3: The architecture of *BioChainSumm*.

number of strong concept chains they contain. The architecture of *BioChainSumm* is shown in Figure 7.3.

$$Score(\text{concept-chain}) = freq(\nu) \cdot count(unique), \tag{7.3}$$

where, $freq(\nu)$ is the frequency of the frequent concept ν in a concept-chain. $count(unique)$ is the number of unique concepts in a concept-chain.

7.2.6 Toutanova et al. (2007)

PYTHY is an approach based on a log-linear ranking model which maximizes the measure of sentence importance. Toutanova et al. (2007) give scores (Equation 7.4) to the sentences in a document cluster based on the following features: relative frequency of content words in the cluster, frequency of words in the topic, sentence length, sentence position, bigram or multigram frequency, a binary feature for the presence of a verb in the sentence and inverse document frequency.

$$score(sent) = \sum_{i=1 \cdots K} w_k f_k(sent), \tag{7.4}$$

where, f_k represents the k^{th} feature of sentence $sent$ and w_k represents the weight of the k^{th} feature. The feature weights are learned using pair-wise ranking training criteria. The objective function is shown in Equation 7.5.

$$max(\sum_{sent_i > sent_j} log(\frac{e^{\sum w_k f_k(sent_i)}}{e^{\sum w_k f_k(sent_i)} + e^{\sum w_k f_k(sent_j)}})) \tag{7.5}$$

PYTHY deals with non-redundancy by calculating the score of a sentence as shown in Equation 7.4. Every time a best scored sentence is included in the summary, the scores of the remaining sentences are re-calculated by giving discount to some features in the score.

7.3 Discourse Structure-based Approaches

While previous sections focus on surface-level approaches, this section focuses on the structure-based approaches. This section gives the details of the approaches that exploits the discourse

structure of the documents. Similar to this, we incorporate the structural feature for coherence in our approach.

7.3.1 Barzilay & Elhadad (1997)

Barzilay & Elhadad (1997) develop a technique to produce summaries by considering lexical chains. A lexical chain is a model of the topic progression in a document. The architecture of the technique is shown in Figure 7.4.

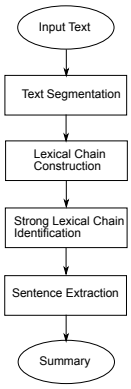


Figure 7.4: The flow diagram of a method introduced by Barzilay & Elhadad (1997)

The text segmentation process in Figure 7.4 is executed to identify sentence boundaries in a document. Further, lexical chains are computed by using knowledge sources such as WordNet, a part-of-speech tagger, and a shallow parser. The strong lexical chains are identified by calculating the score as shown in Equation 7.6.

$$Score(Chain) = Length(Chain) \cdot HomogeneityIndex(Chain), \tag{7.6}$$

where $Length$ represents the number of members in a $Chain$.

$$HomogeneityIndex(Chain) = 1 - \frac{Unique(Chain)}{Length(Chain)} \tag{7.7}$$

where, $Unique(Chain)$ represents the number of unique members in $Chain$. Afterwards, the strong chains are identified if they satisfy the "Strength Criterion":
$Score(Chain) > Average(Scores) + 2 \cdot StandardDeviation(Scores)$.
Then, the sentence is extracted for the summary if it contains the first occurrence of a chain member in the document.

7.3.2 Marcu (1997a)

Marcu (1997a) develops a psycholinguistic method, which utilizes the discourse structure of a document, for summarization. Moreover, he proposes an evaluation method for discourse-

based summarization technique.

In this approach, discourse structure of a text is identified by the rhetorical parsing algorithm (Marcu, 1997b). Then, the discourse structure is utilized to assign scores to textual units of the document. The assigned scores are evaluated by a psycholinguistic experiment showing that scores and important textual units correlate. The top scoring textual units are considered for the summary. Finally, Marcu (1997a) evaluates the importance of a summary by asking human judges. Human judges give score to the summary on a three-point scale: 2 is given to an important and concise summary, 1 is given to a moderate summary, and 0 is given to an unimportant summary. The results of this experiment shows that the discourse theories are capable of producing better summaries of text documents.

7.3.3 Strzalkowski et al. (1998)

Strzalkowski et al. (1998) describes an approach which utilizes the structure of the written text. According to their empirical study, there are certain regularities in the organization of textual units and style of writing known as the Discourse Macro Structure (DMS), which can be exploited to produce a coherent summary. Using the DMS, the approach produces two types of summaries: indicative and informative.

According to Strzalkowski et al. (1998), the DMS of scientific articles is more complex than news articles, for instance, for scientific articles, the DMS is *Introduction-Method-Results-Discussion-Conclusion* and for news articles, it is *Background-Main News*.

The approach works on the paragraph level and produces a summary consisting of whole paragraphs. Strzalkowski et al. (1998) presume that paragraphs are self-contained and express a single thought or matter.

The paragraph is selected on the basis of the assigned score. The score (Equation 7.8) is computed by considering the following criteria:

- Words or phrases occur frequently in a paragraph.
- Words or phrases occur in a title of a paragraph.
- Noun phrases as a subject phrase in a paragraph.
- Words or phrases occurring in few paragraphs.
- Paragraphs appear in the beginning of the news document.
- Proper names appear in a paragraph.
- Cue phrases appear in a paragraph.

$$Score(para) = \frac{1}{F(abs(len_{para} - len_{summary}))} \sum_{h} w_h c_h \tag{7.8}$$

where, F is a normalization function with an argument $abs(len_{para} - len_{summary})$. c_h is value of one of the above criterions; w_h is the weight, that shows the effectiveness of the criterion. The summary is generated by maximizing the score, while maintaining the length constraint.

7.3.4 Boguraev & Kennedy (1999)

Unlike earlier approaches, Boguraev & Kennedy (1999) use phrasal level information instead of sentence level information for a summary. Phrasal units of documents, referred to as the topic stamps, are determined by using linguistically intensive approaches (anaphora resolution, phrasal analysis). The topic stamps are presented in such a way that they reflect the global context of the document known as the capsule overview.

The capsule overview is a representation of a document, obtained by applying a data reduction technique on the document. This representation is not a summary as it does not contain sentences. This approach does not focus on the summary generation, but rather on the identification of the topic stamps and the production of the capsule overview of a document. Moreover, the approach is domain and genre independent. The tasks involved in obtaining the capsule overview of a document are:

- Text pre-processing
- Linguistic analysis
- Discourse segmentation
- Extended phrasal analysis
- Anaphora resolution
- Calculation of discourse salience
- Topic stamp identification

The time stamps and the capsule overview can be utilized further to produce summaries.

7.3.5 Teufel & Moens (1999)

Teufel & Moens (1999) develop an approach for the summarization of scientific articles. The approach is based on the argumentative zoning of the text document. The argumentative zoning describes the rhetorical roles of the sentences in a text document. The argumentative roles are defined as *GOAL*, *ACHIEVEMENT*, *BACKGROUND*, *METHOD*, etc.

First, this approach separates the relevant and irrelevant sentences of a document, where relevant sentences carry rhetorical roles. Then, the identification of rhetorical roles of relevant sentences is done by classifying them according to one of the seven rhetorical roles. The rhetorical roles considered in this approach are as follows:

- BACKGROUND
- TOPIC/ ABOUTNESS
- RELATED WORK
- PURPOSE/PROBLEM
- SOLUTION/METHOD
- RESULT
- CONCLUSION/CLAIM

The rhetorical roles are considered as one of the features of the classifier, which gives probability scores to sentences. On the basis of this score, sentences are selected for the summary. The evaluation is based on co-selection between the gold standard sentences and the automatically extracted sentences. The results show that the rhetorical roles are useful features for the summary generation.

7.3.6 Ercan & Cicekli (2008)

Ercan & Cicekli (2008) consider the lexical cohesion structure in the text to extract the significant sentences for a summary. Ercan & Cicekli (2008) use lexical chains to analyze the structure and find topics in the text.

The algorithm for summarization consists of the following steps:

- Sentence Detection: Parser is used to detect the sentence boundaries.
- Part of Speech Tagging: The MaxEnt Part of Speech tagger is used to find POS tags.
- Noun Phrase Detection: Chunker is used to find the noun phrases.
- Lexical Chaining: The lexical chains are made by using Galley & McKeown (2003)'s algorithm.
- Filtering Weak Lexical chains: The lexical chains filtering is accomplished by using the strength score of lexical chains which is introduced by Barzilay & Elhadad (1997).

- Clustering Lexical Chains: Lexical chains that occur together are considered to be in the same cluster. These chains make a set of topics that are related to each other.
- Extracting Sequences with Respect to Clusters: For each cluster, sequences are formed on the basis of the presence of lexical chain member of the cluster in a sentence.
- Sentence Selection: The first sentence of each sequence is extracted for the summary.

7.3.7 Louis et al. (2010)

Louis et al. (2010) describe a classification approach for summarization. They describe two important sets of discourse features: structural and semantic. They analyse the predictive power of these features in comparison to non-discourse features. They conclude that discourse features are more robust indicators of important contents in the text documents.

Structural features are obtained from RST trees (Mann & Thompson, 1988). These features capture the importance of the text segment on the basis of its position in the global structure of the text. Semantic features identify the type of relation between two sentences; however, they do not contain structural information of the text. These features are computed by using penn discourse tree bank annotations (Prasad et al., 2008). Finally, these features are utilized in the classification task to identify important sentences from the text. Results show that using both features for summarization substantially improves the accuracy.

7.3.8 Contractor et al. (2012)

Contractor et al. (2012) perform scientific article summarization using argumentative zoning. The argumentative zones of sentences are utilized as features to extract the sentences for the summary.

The method has two main steps: classification and clustering. The classification step is used to identify initial candidate sentences for the summary. The clustering step groups similar sentences into one cluster, which helps to remove redundancy from the candidate set of sentences.

To train the classifier, the sentences in the abstracts of the corpus are considered as positive labeled instances, whereas the sentences in the main text are considered as the unlabeled data, i.e., they can have either positive or negative labels. The training of the classifier using positive and unlabeled data is done by Elkan & Noto (2008). The features used are: verbs, tf-idf values, citation sentences, argumentative zones, and position of sentences.

In the clustering step, similar sentences are grouped together on the basis of their argumentative zones. Then, K-means is applied to cluster the sentences within each argumentative zone labels. The centroid sentence from each cluster is considered for the final summary.

7.3.9 Christensen et al. (2013)

Christensen et al. (2013) introduce a novel approach which extracts sentences on the basis of salience and coherence. They use graphical representation of documents, referred to as G-FLOW, where nodes correspond to sentences and edges correspond to the discourse relations among sentences. If there is an edge between sentences s_i and s_j then sentence s_i can be placed next to sentence s_j in a summary to make it coherent.

Christensen et al. (2013) create an edge between two sentences if there exists a discourse relation, such as discourse cues, deverbal nouns, co-reference, event/entity combination, and more. They give weights to edges which depend on the type of discourse relation between the sentences, for instance, if the edge exists due to the discourse markers, then the edge weight will be two. They also incorporate negative weights to the edges. An edge between two sentences contains a negative weight, if a sentence contains a deverbal noun reference, a discourse marker, or a co-reference mention that is not satisfied by another sentence.

The summary is generated by maximizing the objective function shown in Equation 7.9. The objective function is subject to a summary length constraint.

$$max(Sal(S) + \alpha Coh(S) - \beta|S|), \tag{7.9}$$

where, $Sal(S)$ corresponds to the salience of sentences in the summary. Salience of a sentence is defined as the sum of the salient features with their weights. $Coh(S)$ corresponds to the coherence of the summary. Coherence is defined as the sum of the edge weights between the sentences of the summary shown in Equation 7.10.

$$Coh(S) = \sum_{i=1}^{|S|-1} weight_{+}(s_i, s_{i+1}) + \lambda weight_{-}(s_i, s_{i+1}), \tag{7.10}$$

where, $weight_{+}$ is for the positive edge weights, $weight_{-}$ corresponds to the negative edge weights, and λ is a tradeoff coefficient. $|S|$ is used to avoid small sentences in the summary.

Similar to the proposed approach, Christensen et al. (2013) represent documents graphically using discourse relations. Here, the graph is a general graph in contrast to our graphical representation. Unlike Christensen et al. (2013), our proposed approach considers the overall structure of the summary.

7.3.10 Liakata et al. (2013)

The approach developed by Liakata et al. (2013) exploits the automatically produced scientific discourse annotations (CoreSc) for the summarization of scientific articles. First, the scientific articles are annotated using the CoreSc scheme. This scheme utilizes the 11 content-based concepts such as *Result*, *Methodology*, *Hypothesis* etc. Then, Liakata et al. (2013) observe

the sequence of CoreSc categories in abstracts to give the skeleton of the final summary. The CoreSc categories found in the scientific article is also considered while creating the final summary. They utilize the 28 scientific articles of chemistry for the evaluation purpose.

Similar to this, we utilize the structure of abstracts to generate the final summary. We evaluate our approach on the dataset created by Liakata et al. (2013) (see Chapter 6).

7.3.11 Jha et al. (2015)

Jha et al. (2015) introduce a discourse-based approach to generate coherent and readable summaries of scientific articles. In this approach, they combine a content model and a discourse model for the summarization of scientific articles.

In the content model, Jha et al. (2015) find the subtopics of scientific articles and transitions between them using hidden markov models. These subtopics are considered for the guidance of the summarizer to produce coherent summaries.

In the discourse model, Jha et al. (2015) introduce *Minimum Independent Discourse Contexts (MIDC)* to avoid the problem of disconnectivity among the sentences of a summary in extractive summarization. They calculate the *MIDC* of each sentence based on the discourse rules. These rules are activated by coreference chains, explicit discourse relations, and entity links between sentences.

Firstly, *surveyor* finds the subtopics using the content model; then, the LexRank algorithm is applied to find the most salient sentence in each subtopic. Afterwards, it calculates the *MIDC* of the most central sentence of each subtopic. If *MIDC* of the sentence does not exceed the maximum allowed length, it is included in the summary.

In contrast to the proposed approach, Jha et al. (2015) utilizes greedy algorithm to extract sentences for a summary. Similar to Christensen et al. (2013), this approach does not consider the overall structure of the summary.

7.4 Graph-based Approaches

In this section, we briefly describe the approaches that use graph representations for input documents. Similarly, our approach represents documents graphically. However, we use bipartite graph for the representation that has not been used in the summarization task.

7.4.1 Mihalcea & Tarau (2004)

TextRank is a graph-based ranking model for summarization introduced by Mihalcea & Tarau (2004). In this approach, documents are represented graphically, where nodes represent

sentences. The graph is a fully connected directed graph with edge weights as the cosine similarity (on tf-idf) between the corresponding sentences (Figure 7.5). The direction of edges corresponds to the occurrence of sentences in the input document.

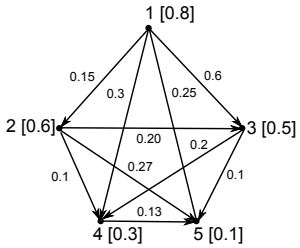


Figure 7.5: A graphical representation of a document.

Mihalcea & Tarau (2004) apply the PageRank ranking algorithm (Brin & Page, 1998) to the graph of a document to obtain ranks of the sentences. Here, the ranks are considered as, the importance of sentences, i.e., higher-ranked sentences are more important.

Let $G = (V, E)$ be a directed fully connected graph, where V is the set of vertices and E is the set of edges. The rank of a vertex v_i is calculated as follows:

$$R(v_i) = (1 - d) + d \times \sum_{j \in indegree(v_i)} \frac{1}{|outdegree(v_j)|} R(v_j), \tag{7.11}$$

where, d is a damping factor which can be set between 0 and 1. $indegree(v_i)$ is the set of vertices pointing to vertex v_i and $outdegree(v_j)$ is the set of vertices that are pointed by vertex v_j.

Top ranked sentences are considered for the summary. In Figure 7.5, a document with 5 sentences is represented graphically, where ranks of sentences are written in square braces.

7.4.2 Radev et al. (2004b)

The approach by Radev et al. (2004b) is a graph-based approach (*LexRank*). The sentence importance is based on the eigenvector centrality in a graph of a document.

The document is represented graphically, where nodes are sentences and edges connect the sentences. Unlike *TextRank*, the graph is undirected. The edge weights in the graph are calculated by cosine similarity between the corresponding sentences.

Radev et al. (2004a) set a certain threshold for the edge weight to prune weak edges in the graph. Then, the modified PageRank algorithm (Equation 7.12) is applied on the graph to obtain importance score. The top ranked sentences are extracted for the summary.

$$R(v_i) = \frac{d}{N} + (1 - d) \sum_{u_j \in adj[v_i]} \frac{R(u_j)}{deg(u_j)}, \tag{7.12}$$

where, d is the damping factor and N is the number of nodes in a graph. $adj[v_i]$ is the set of adjacent nodes of vertex v_i and $deg(u_j)$ calculates the number of edges to connect vertex u_j with other vertices of the graph.

7.4.3 Nastase (2008)

Nastase (2008) develops an approach for query-based multi-document summarization. She expands the query to understand it properly using encyclopedic knowledge of Wikipedia. For the query expansion, she extracts all the open-class words and named entities from the topic and expands them using Wikipedia articles which contain these topic words in their titles. The query is also expanded by using hypernyms, hyponyms and antonyms in WordNet.

Further, Nastase (2008) represents the document and the expanded query graphically, where nodes correspond to lemmatized words, and named entities and edges represent the grammatical dependency relations. Then, she extracts a subgraph which consists of all the open class words and named entities in the expanded query. In the extracted subgraph, each edge corresponds to the grammatical relation in a sentence of a text document. Further, Nastase (2008) gathers all the sentences in the subgraph and gives them scores based on the topic coverage and the number of edges covered by them. Top scored, non-redundant sentences are extracted for the summary, where non-redundancy is taken care of by a simple lexical overlap.

7.5 Topic Modelling-based Approaches

In this thesis, we propose an approach that uses topic modelling for the representation of documents (see Section 5.1.2). So, we provide the brief details of the approaches that use topic modelling for the summarization task in this section.

7.5.1 Lawrie et al. (2001)

The approach developed by Lawrie et al. (2001) does not generate the summary, however, it helps in finding the topic terms and constructing the relations between them, which can be later used to produce better summaries.

The topic terms have a high conditional probability in comparison to other terms in the vocabulary. This conditional probability measures the predictive power of a term and is utilized to build a probabilistic language model of the vocabulary.

The topic terms of the summary must have maximal predictive power and coverage of the vocabulary. To determine the topic terms for the summary, the probabilistic language model is transformed into a graph, and then graphical algorithms are utilized to find the topic terms.

7.5.2 Haghighi & Vanderwende (2009)

Haghighi & Vanderwende (2009) develop a generative probabilistic model for extractive multi-document summarization. They introduce two approaches for summarization based on topic modelling: *TOPICSUM* and *HIERSUM*.

TOPICSUM consists of three type of topic distributions: *BACKGROUND*, *CONTENT* and *DOCSPECIFIC*.

- **BACKGROUND**: This distribution is used for the stopwords.
- **CONTENT**: This distribution is used for the words which contain important information.
- **DOCSPECIFIC**: This distribution is used for document specific words which are local to one document.

After computing these topics, *TOPICSUM* draws a distribution over the topics: *BACKGROUND*, *CONTENT* and *DOCSPECIFIC* for each sentence of a document. Then, Haghighi & Vanderwende (2009) extract sentences using the KL-divergence with the *CONTENT* distribution of each document set.

HIERSUM considers two types of content distributions: the general content distribution and the specific content distribution. The general content distribution chooses words which consistently occur throughout a document as well as in many documents . The specific content distribution chooses words which occur in several documents but are concentrated in a small number of sentences. Using these distributions, *HIERSUM* can extract two types of summaries: general summaries by using the KL-divergence with general content distribution and topic specific summaries by using the-KL divergence with specific content distribution.

7.5.3 Guo et al. (2015)

Guo et al. (2015) introduces an approach for summarizing scientific articles using information structure in them. They use Argumentative Zoning (AZ) scheme (Teufel & Moens, 2002) for the information structure.

Guo et al. (2015) apply latent dirichlet allocation to determine the information structure of sentences in the document. They define topics as a list of features rather than a bag-of-words. The features used in topic modelling are: citation, tables, figures, personal and possessive pronoun, conjunction, adjective, adverb, modal, tense, and voice.

Guo et al. (2015) uses graph clustering techniques to find the information conveyed by the topics, to each sentence. They represent the document graphically having nodes as sentences and the edge weights as the topics shared between the corresponding sentence nodes.

Then, they apply the graph clustering technique (Dhillon et al., 2007) and extract the central sentences of the clusters for a summary.

Similar to the proposed approach, Guo et al. (2015) utilizes topical information to represent the documents graphically. Topics in this approach are list of predefined features. Unlike Guo et al. (2015), our proposed approach takes care of coherence while extracting important but non-redundant sentences for a summary.

7.6 Citation-based Approaches

This section gives the brief descriptions of citation-based approaches. These approaches are only for scientific articles. In this thesis, our proposed approach also focuses on the summarization of scientific articles.

7.6.1 Qazvinian & Radev (2008)

Qazvinian & Radev (2008) develop an approach for summarizing scientific articles using citations. The citation-based approach is based on exploring others' perspective of the target article's contribution.

Qazvinian & Radev (2008) use the ACL Anthology Network (Joseph & Radev, 2007) to produce citation summaries. They collect five clusters of scientific papers, where the scientific articles in each cluster belong to one topic. They create a citation network of an article, where nodes are citation sentences of the target article. This graph is a fully connected undirected weighted graph, where edge weights correspond to the similarity between the two citation sentences. A high similarity score between two citation sentences shows that they share the same facts. Then, a graph clustering method is applied to obtain the communities in the citation summary network.

After building the network and forming the communities, Qazvinian & Radev (2008) extract the sentences for the final summary using two different methods: *Cluster Round-Robin (C-RR)* and *Cluster Lexrank (C-lexrank)*. In *C-RR*, they extract the sentences in order of their occurrence in the clusters. They start with the largest cluster to extract the first sentence, and then the first sentences from the remaining cluster, then the second sentences of every cluster, and so on until the summary length limit is attained. In *C-lexrank*, Qazvinian & Radev (2008) apply LexRank (Erkan & Radev, 2004) on each cluster. Then, they extract the most central sentence from every cluster, and if the summary length limit is not attained, the second most salient sentence from every cluster is extracted, and so on.

7.6.2 Teufel et al. (2006)

Teufel et al. (2006) develop an approach to automatically classify the citation sentences on the basis of their rhetorical function, which is useful for summarizing scientific articles. They introduce four categories of citation sentences on the basis of their functions: *Weak*, *Contrast*, *Positive* and *Neutral*.

Teufel et al. (2006) use classification algorithms to automatically classify citation sentences. For this, they ask annotators to assign citation functions to sentences of the training data. The features used are: cue phrases, cues identified by annotators, verb tense, voice and modality.

Teufel et al. (2006) hypothesize that the citation functions of the sentences can be helpful in creating coherent summaries of scientific articles. They assume that the annotation of citation functions is similar to the discourse structure of scientific articles.

7.6.3 Abu-Jbara & Radev (2011)

Abu-Jbara & Radev (2011) introduce an approach to produce coherent and readable summaries of scientific articles using citation sentences. In this approach, coherent citation summaries are achieved by identifying reference scopes of citations.

The scope of a reference is defined as the shortest fragment of the citation sentence which is grammatically sound. The shortest fragment must contain the citation of the target paper. To find the fragment, Abu-Jbara & Radev (2011) use link grammar parser (Temperley et al., 1991), which is not trained on the citation sentences. Thus, Abu-Jbara & Radev (2011) replace the references with the tags, for instance, target reference is replaced by the *TREF* tag. Finally, the extracted fragment is used for summarizing the target scientific article.

Abu-Jbara & Radev (2011) apply a sentence filtering technique in which they classify sentences into two classes, the suitable sentences and the unsuitable sentences, using the support vector machine algorithm. Afterwards, they identify the citation functions of the suitable sentences: *Background*, *Problem-statement*, *Method*, *Results*, and *Limitations*. Then, the clustering of the sentences is done within each citation functional category. After clustering, *LexRank* is applied within each cluster to obtain the ranks of the sentences. Then, the sentences are selected for the summary in an order: first, based on their category, second, based on the size of their cluster and then based on their ranks.

The citation functional categories are ordered as *Background*, *Problem-statement*, *Method*, *Results* and then *Limitations*. In each functional category, clusters are ordered according to the number of sentences in the clusters and in each cluster, sentences are ordered according to their LexRank values as shown in Figure 7.6. Eventually, they apply the postprocessing to the selected sentences for the summary using reference scope to obtain better coherent summaries.

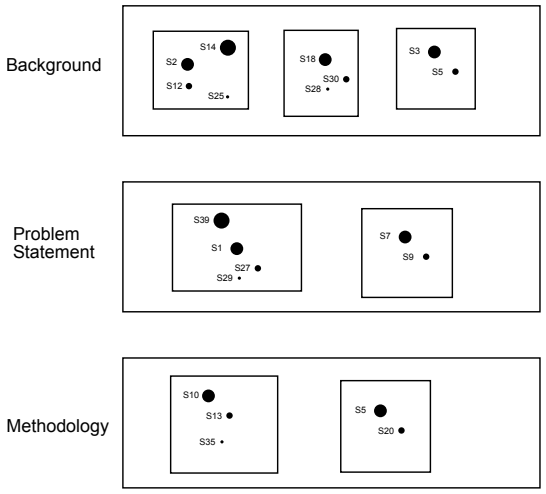


Figure 7.6: An example of sentence selection.

7.6.4 Xu et al. (2015)

Xu et al. (2015) develop an approach to summarize the main contribution of a scientific article using its citations. First, they automatically identify the keywords from the citation sentences of the target scientific article; then the keywords are exploited to identify the citation sentences that best capture the main contribution of the target scientific article.

To identify the keywords, Xu et al. (2015) calculate the probability of observing the k citing sentences of the citation summary of scientific article d, containing word w:

$$P(X = k|N, K, n) = \frac{\binom{K}{k}\binom{N-K}{n-k}}{\binom{N}{n}}, \tag{7.13}$$

where X is the number of citing sentences in the citation summary of d that consists of word w. K is the number of sentences in the citation summary of d. N is the total number of sentences in the citation summaries of the articles belonging to collection C. n is the total number of citing sentences which contain word w. Afterwards, they use the hypergeometric test to obtain a p-value which identifies the salience of word w in characterising the main contribution of d.

Further, Xu et al. (2015) use generative probabilities to translate the salience of words (p-value) into a discriminative language model (KPLM). This model gives high probabilities to the keywords which can serve as the characteristics of the main contribution of the paper. Then, Xu et al. (2015) select the set of citing sentences that captures the salient keywords to create a summary.

7.6.5 Cohan & Goharian (2015)

Cohan & Goharian (2015) propose an approach for summarizing the scientific articles using citation sentences and the discourse structure of an article. The approach consists of four steps: extraction of citation-contexts, grouping of citation-contexts, ranking of sentences within a group, and selection of sentences for the summary. The article which contains a citation sentence is referred to as "reference article".

To extract the citation-contexts, Cohan & Goharian (2015) create a vector space model of the citation sentence and rank the sentences on the basis of the similarity between the citation sentence and other sentences in the reference article. The highest ranked sentences are considered as the citation-contexts.

Then, Cohan & Goharian (2015) build a graph of the highest ranked sentences, where nodes are sentences and edge weights are the similarity score between the corresponding sentences. To group the citation-contexts, Cohan & Goharian (2015) find the subgraphs having high and low intra-connectivity. The subgraph is considered as the group of citation-contexts.

After grouping the citation-contexts, Cohan & Goharian (2015) consider a group as a graph and calculate the centrality of nodes. The centrality of nodes can be measured in various ways: nodes degree, eigenvectors, betweenness and closeness. The sentence selection for the summary generation is accomplished by two approaches: the iterative approach and the greedy approach. The iterative approach selects the top ranked sentences from all groups until the summary length limit is reached. The greedy approach is similar to Maximal Marginal Relevance (MMR) (Carbonell & Goldstein, 1998) in which sentences are selected from a group using the formulae:

$$MMR(sent_i) = \lambda Sim_1(sent_i, D) - Sim_2(sent_i, summary), \tag{7.14}$$

where, sim_1 and sim_2 are the similarity measures.

7.7 Integer Programming-Based Approach

In this section, we describe the approaches that utilize the integer programming technique. In our proposed approach, we also use integer programming to obtain the best sentences for the final summary.

7.7.1 McDonald (2007)

McDonald (2007) introduces an approach for summarization, based on global inference algorithms. In this approach, McDonald (2007) maximizes two factors: relevance and non-redundancy. The formulation of the summarization inference problem is shown below:

Objective function:

$$Summ = \text{arg max}(f(Summ)) \tag{7.15}$$

$$= \text{arg max}_{Summ \subseteq Doc} \sum_{sent_i \in Summ} Relevance(sent_i) - \sum_{sent_i, sent_j \in Summ, i<j} Redundancy(sent_i, sent_j) \tag{7.16}$$

Subject to:

$$\sum_{sent_i \in Summ} length(sent_i) \leq length(Summ) \tag{7.17}$$

The constraint in Equation 7.16, $Relevance(sent_i)$ measures the importance of sentence $sent_i$ and
$Redundancy(sent_i, sent_j)$ measures the redundancy between sentences $sent_i$ and $sent_j$.
The constraint in Equation 7.17 represents a length constraint.

The global inference problem is NP-Hard, i.e., it cannot be solved in polynomial time, however, there are some approximations, such as greedy and dynamic programming, which can be utilized to solve it nevertheless. McDonald (2007) compares solutions of the global inference problem using approximation techniques with the exact solution. McDonald (2007) uses integer linear programming to obtain the exact solution. He found that a dynamic programming algorithm yields optimal accuracy and scaling properties, as compared to both a greedy algorithm and an exact algorithm that uses Integer Linear Programming.

7.7.2 Galanis et al. (2012)

Galanis et al. (2012) present a method for extractive summarization using integer linear programming and Support Vector Regression (SVR). They maximize the importance and the diversity in a summary using integer linear programming.

The support vector regression model calculates the importance of sentences in a document. In the training phase of SVR, the target score of a sentence in a document in the training dataset is the average of ROUGE-2 and ROUGE-SU4 of the sentence. The ROUGE scores are obtained when the sentence is compared to the gold summary of the document. The features employed in the SVR model are:

- Sentence position in the document.
- Named entities in the sentence.
- Levenshtein distance of the sentence with the topic.
- Word overlap between the topic and the sentence.

- Sum of the content word frequencies.

Galanis et al. (2012) model the objective function of integer linear programming which maximizes the sentence importance and number of unique bigrams. The objective function of the integer linear programming model is as follows:

$Objective \quad function$:

$$max(\lambda_1 \cdot importance(Summ) + \lambda_2 \cdot diversity(Summ)), \tag{7.18}$$

where, λ_1 and λ_2 are tradeoff parameters, which are tuned on the development dataset.

In contrast to the proposed approach, Galanis et al. (2012) do not incorporate coherence factor in their model. Moreover, their algorithm is fully supervised.

7.7.3 Hirao et al. (2013)

Hirao et al. (2013) introduce an approach for single-document summarization based on a tree knapsack problem (Chapter 3). The approach considers the rhetorical relations between textual units of a document while summarizing the document to obtain a coherent summary.

The rhetorical structure theory (Mann & Thompson, 1988) based discourse tree (RST-DT) does not explicitly determine the parent-child relationship, so it is difficult to formulate the summarization task as a tree knapsack problem. Thus, Hirao et al. (2013) propose rules to transform a rhetorical structure theory-based discourse tree, into a dependency-based discourse tree (DEP-DT).

Then, Hirao et al. (2013) formulate the summarization task in order to search for the optimal rooted subtrees. The ILP formulation to solve the summarization task is shown below:

$Objective \quad function$:

$$max(\sum_{i=1}^{N} \frac{\sum_{w \in W_{st_i}} tf(w, doc)}{depth(st_i)} y_i) \tag{7.19}$$

$Subject \quad to$:

$$\sum_{i=1}^{N} length(y_i) \leq L_{max} \tag{7.20}$$

$$y_{parent_i} \geq y_i \quad \forall i \tag{7.21}$$

$$y_i \epsilon [0, 1] \quad \forall i \tag{7.22}$$

The objective function in Equation 7.19 maximizes the significance score of rooted subtrees obtained from a dependency-based discourse tree. y_i is a binary variable of rooted subtree

st_i. $tf(w, doc)$ calculates the frequency of word w in document doc. N is the number of rooted subtrees.

The constraint in Equation 7.20 ensures that the length of a summary must not exceed the maximum allowed length L. The constraint in Equation 7.21 exhibits that a summary is a rooted subtree of dependency-based discourse tree.

7.7.4 Gorinski & Lapata (2015)

Gorinski & Lapata (2015) develop an approach for movie-script summarization using integer linear programming. In the approach, they use graphical representations of the movie scripts. The approach selects the chain of scenes by optimizing diversity, importance and logical progression.

Gorinski & Lapata (2015) represent a movie-script as a weighted bipartite graph, where one set of nodes is scenes (S) from a movie script and the other set is characters (C) (Figure 7.7). The edge weights are of two types: one is from character to scene and another is from scene to character, hence the edges have two directions as shown in Figure 7.7. The edge weight for the 'character to scene' transition is the probability of a character being central in the scene. The edge weight for the 'scene to character' transition is the ratio of the number of interactions of a character in a specific scene, to the total number of interactions of the character in the script.

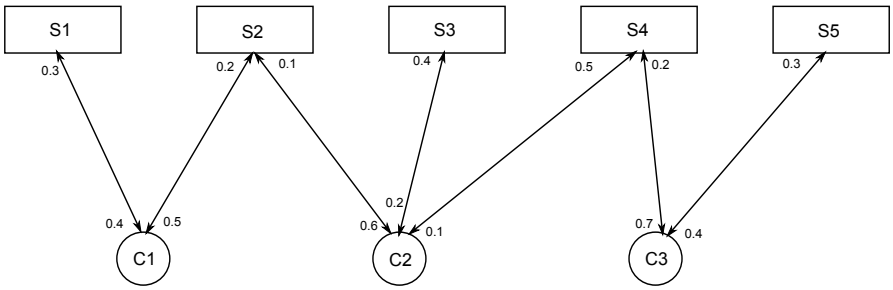


Figure 7.7: A bipartite graph of a movie script.

Then, Gorinski & Lapata (2015) formulate the scene extraction model as follows:

$$argmax(\lambda_I Imp(S) + \lambda_d div(S) + \lambda_p Prog(S)), \tag{7.23}$$

where, $Imp(S)$ refers to the importance of scenes in a chain, which is the ratio of the main characters to support characters in the scene. $div(S)$ represents the diversity in the scenes of a chain by including dissimilar scenes. $Prog(S)$ is responsible for selecting a chain consisting of coherent scenes. This progression function is modeled in terms of the strength of characters in the selected scene and influence of the same characters in the next scene. The objective

function is subject to a constraint which ensures that the summary length must not exceed the maximum allowed length.

Similar to the proposed approach, Gorinski & Lapata (2015) use bipartite graph representations. However, the bipartite graph in this approach consists of bidirectional edges, whereas our graph contains unidirectional edges. In contrast to this approach, our method takes into account the whole structure of the final summary.

7.7.5 Schluter & Søgaard (2015)

Schluter & Søgaard (2015) introduce an unsupervised approach for summarization using coverage maximization. They use same the integer linear programming formulation as Gillick et al. (2009) but in addition to bigrams they use new concepts.

Schluter & Søgaard (2015) use three new syntactic and semantic concepts to produce better summaries. They define new concepts as: named entities, syntactic dependencies and semantic frames. Schluter & Søgaard (2015) hypothesize that a summary should contain persons, organizations, and locations mentioned in the input document. Moreover, the summary should contain the salient semantic frames present in the input document. The objective function shown below is subject to the summary length constraint:

$$(1-\lambda)\sum_{i}^{N_b} w_{b_i} B_i + \lambda \sum_{j}^{N_c} w_{c_j} C_j, \tag{7.24}$$

where, B_i is the binary variable of bigram b_i and w_{b_i} is the importance weight of bigram b_i. λ is a tuning factor set to 0.5, i.e., giving equal weight to bigrams and syntactic and semantic concepts. C_j is the binary variable of concept c_j and w_{cj} is the importance weight associated with concept c_j. N_b and N_c are the number of bigrams and new concepts in the input document, respectively.

7.7.6 Yogatama et al. (2015)

Yogatama et al. (2015) introduce an extractive summarization algorithm by maximizing semantic volume. They represent each sentence as low-dimensional vectors in a distributed semantic space using singular value decomposition.

In the approach, a subset of sentences is selected for the summary whose convex hull maximizes volume in the semantic space, as shown in Figure 7.8. Convex hull is the smallest convex set which contains all the points in set X. In Figure 7.8, sentences are represented in the low-dimensional semantic space with two vectors $y1$ and $y2$. If the maximum length of the summary is five sentences, then sentences $s1$, $s2$, $s4$, $s5$, and $s7$ are selected for the summary as they are maximizing the volume.

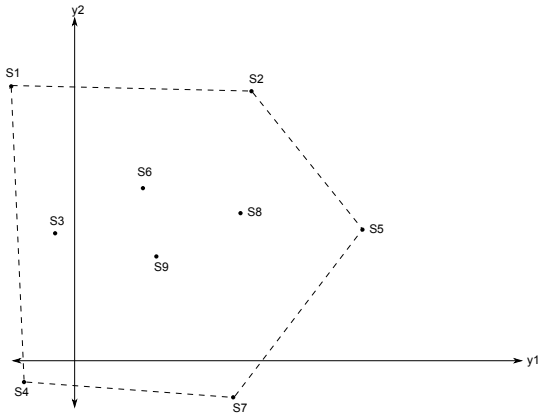


Figure 7.8: A toy example of convex hull.

Yogatama et al. (2015) formulate the selection of sentences for the summary by maximizing semantic volume. Objective function:

$$\arg\max(Volume(\psi(Summ))) \tag{7.25}$$

Subject to:

$$Length(Summ) \leq L \tag{7.26}$$

The objective function in Equation 7.25, $\psi(Summ)$ represents the vectors of sentences in summary $Summ$ and function $Volume$ calculates the volume of sentences in the semantic space. The constraint in Equation 7.26 ensures that the length of the summary must not exceed the maximum allowed length L.

7.8 Neural Network-based Approaches

In this section, we briefly describe the approaches that use neural networks. These approaches are not completely related to our approach; however, they are the new trend in the summarization task.

7.8.1 Liu et al. (2012)

Liu et al. (2012) develop a framework for query-based multi-document summarization, based on a deep learning model. This framework has three parts: extraction of content, generation of summary, and reconstruction validation. These parts are tightly connected to produce a summary with all the important information of the documents.

In the content extraction step (shown in Figure 7.9), the feature vector of each document is given as an input to the deep neural network. The feature vector consists of term frequencies of the vocabulary words in the document. The deep neural network in the content extraction phase, consists of four hidden layers: filtration of the accidental words, detection of keywords, and extraction of candidate sentences for the summary. Liu et al. (2012) use Restricted Boltzmann Machines (RBMs) as building blocks for these hidden layers.

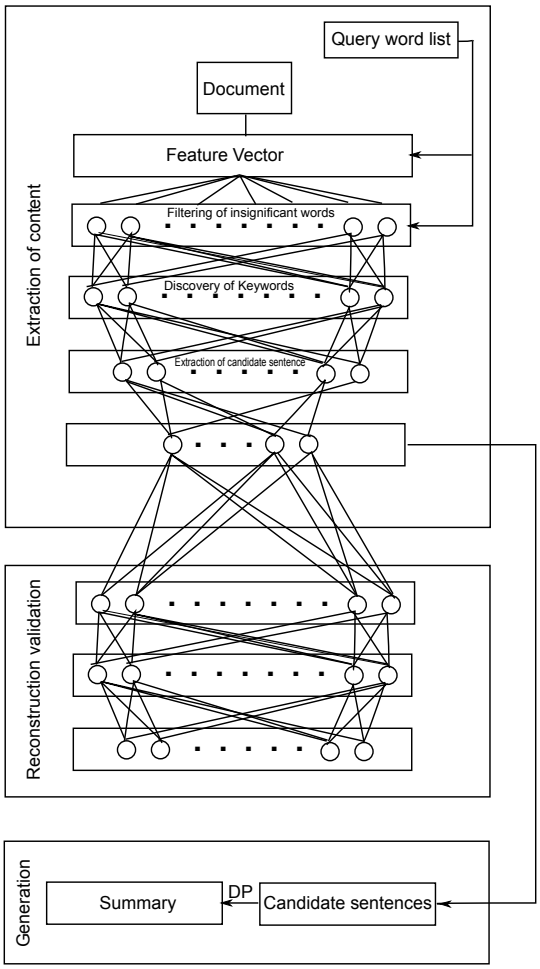


Figure 7.9: The framework of the approach by Liu et al. (2012).

To incorporate the query information in the content extraction step, Liu et al. (2012) assign higher weights to those words in the feature vector, which occur in the query as well as in the document. Then, reconstruction validation tries to find the best local optimum parameters for the content extraction step. This is accomplished by using backpropogation algorithm.

Afterwards, the summary generation step applies a dynamic programming algorithm to select the best sentences for the summary, from the candidate sentences obtained from the content extraction phase. The dynamic programming algorithm maximizes the sentence im-

portance in the summary subject to the length constraint. Sentence importance looks at the important words present in the sentence. Important words are determined from the query and the hidden layers of the content extraction step.

7.8.2 Cao et al. (2015)

Cao et al. (2015) introduce a novel approach (*PriorSum*) for summarization based on the idea of summary prior. The idea is to define a measure that computes the suitability of a sentence for the summary without considering its context.

Cao et al. (2015) use the approach developed by Carbonell & Goldstein (1998), where the sentence ranking method gives scores to the sentences. Here, sentence ranking method is based on document-dependent and document-independent features.

PriorSum applies enhanced convolutional neural network (CNN) to discover document-independent features. The enhanced CNN uses multiple filters for variable window sizes and two max-over-time pooling operations to obtain the summary prior representation. The document dependent features of the sentence are: position , average frequencies of the words, and average cluster frequency values of the words. In the end, both features are combined for the regression framework (Li et al., 2007) to estimate the importance of sentences (Equations 7.27 and 7.28). For the training purpose, *PriorSum* calculates the ROUGE-2 scores of sentences of documents in the training data and considers them as importance scores.

$$F = [x_i, x_d] \tag{7.27}$$

$$t' = W \times F \tag{7.28}$$

where, W is the set of weights for feature set F. x_i is the set of independent features and x_d is the set of dependent features. t' is the importance measure of the sentences. Finally, *PriorSum* selects the best sentences for the final summary on the basis of the importance measure.

7.8.3 Kobayashi et al. (2015)

Kobayashi et al. (2015) develops a summarization approach based on embedding distributions. A word embedding is a parameterized function which maps words to high-dimensional vectors. They produce a summary by maximizing the submodular function based on the word embeddings (Bengio et al., 2003). Submodular functions have a diminishing returns property which makes them appropriate for many other applications.

Kobayashi et al. (2015) propose a submodular objective function which is based on the embedding distribution. The idea is that if two embedding distributions, S and C, are similar then each embedding in S is near to every embedding in C. The formulation of the submodular

objective function is shown in Equation 7.29.

$$F(C) = -\sum_{s \epsilon D} \sum_{w \epsilon s} h(Dist(w, C)), \tag{7.29}$$

where, C is a summary and h is a non-decreasing scaling function. $Dist(w, C)$ measures the distance between word w of sentence s and summary C.

7.9 Summary

In this chapter, we have discussed the works that are related to our approach. We have focused on the approaches: corpus-based approach, discourse-based approach, citation-based approach, graph-based approach, optimization-based approach, topic modelling-based approach, and neural-network based approach.

Chapter 8

Discussion

Summarization has always been an interesting task for the researchers in the field of natural language processing. An automatic text summarization approach takes large documents as an input and automatically gives the gist of those documents. It has several applications in the medical domain, movie reviews, books and various other fields. The approaches for summarization depend on various factors like, type of input documents, type of summaries, genre of documents etc. However, the general framework of the approaches would be the same.

We introduce a new dataset of scientific articles from the bio-medical domain in this thesis. This dataset has several advantages which are discussed in Chapter 2. Unlike other datasets, we do not need humans to create gold-standard summaries for evaluation, as each article is accompanied with its editor's summary. We consider editor's summaries as human summaries and attempt to create system-generated summaries similar to them.

In the introduction of this thesis, we indicate that a summarization approach has to focus on the three factors: importance, non-redundancy and coherence. The previous methods do not consider these factors simultaneously; therefore, we aim to build an approach which produces a summary containing important but non-redundant and coherent information. These three factors are tightly integrated in our approach. This is a novel graph-based approach for summarizing scientific articles. We represent scientific articles graphically and apply graph-based techniques to find the best subset of sentences from the input document. We extract the sentences for the summary by optimizing the factors globally.

We evaluate our systems on the basis of relevance and coherence. The relevance of summaries is measured by using ROUGE scores. Since ROUGE scores do not consider coherence of summaries, we perform human judgement experiments to evaluate coherence. These experiments show that our approach performs substantially better than the state-of-the-art and baseline systems. This supports our claim that the summaries produced by our system contain important, non-redundant and coherent information.

Further, we experiment with the DUC 2002 dataset to evaluate the performance of our

system on different genres of text articles. The results show that our approach performs considerably well as compared to the other state-of-the-art systems and baseline systems. Thus, we can say that our system is robust and scalable as it performs reasonably well with small text articles of different genres.

8.1 Future Work

In this section, we indicate some future research directions which could be explored to improve the proposed approach.

8.1.1 Domain Dependent

There are some methods that incorporate domain information and focus on the discourse structure of the scientific articles to summarize them (Teufel & Moens, 2002; Reeve et al., 2007; Contractor et al., 2012; Guo et al., 2015). They show that including the structural information of scientific articles improve the results of the summarization system. However, in the proposed approach, we do not consider domain knowledge for the summarization of scientific articles. It needs to be investigated that incorporating structural information in the proposed method will improve the quality of the summaries.

In addition, there are some available medical resources such as Medical WordNet (Smith & Fellbaum, 2004) and Onto Builder (Herre, 2015), that may be incorporated in the proposed approach as external knowledge. For instance, Medical WordNet can be used in the formation of entity graphs or topical graphs.

In brief, domain knowledge helps to incorporate rich linguistic information of scientific articles, leading to better summaries, in terms of relevance and coherence.

8.1.2 Evaluation Metrics

Although the evaluation metrics used in the experiments are quite standardized, they still need to be improved. The drawback of ROUGE scores is that they do not evaluate summary on the basis of their coherence. This score cannot distinguish between coherent and non-coherent summaries. In this thesis, we overcome this problem by asking humans to rank the summaries on the basis of their coherence. However, the human coherence assessment is not an efficient way of evaluating summaries. It will be interesting to do the coherence assessment automatically, for instance, one can compare the structure of human summaries and system generated summaries.

In principal, ROUGE scores calculate the overlapping score between a system generated summary and human summaries. For overlapping, ROUGE only considers string matching.

Thus, it does not consider an overlap between the two words, which are similar in meaning. The semantic representations of words, such as word vectors, can be one of the solutions to this problem. It will be interesting to know the scores of state-of-the-art-results using semantic representation.

List of Figures

List of Tables

Bibliography

Abu-Jbara, Amjad & Dragomir Radev (2011). Coherent citation-based summarization of scientific papers. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers),* Portland, Oreg., 19–24 June 2011, pp. 500–509.

Anzai, Yuichiro (2012). *Pattern Recognition & Machine Learning*. Elsevier.

Aone, Chinatsu, Mary Ellen Okurowski, James Gorlinsky & Bjornar Larsen (1999). A trainable summarizer with knowledge acquired from robust NLP techniques. In Inderjeet Mani & Mark T. Maybury (Eds.), *Advances in Automatic Text Summarization*, pp. 71–80. Cambridge, Mass.: MIT Press.

Barzilay, Regina & Michael Elhadad (1997). Using lexical chains for text summarization. In *Proceedings of the ACL Workshop on Intelligent and Scalable Text Summarization, Madrid, Spain, July 1997*, pp. 10–17.

Barzilay, Regina & Mirella Lapata (2008). Modeling local coherence: An entity-based approach. *Computational Linguistics*, 34(1):1–34.

Baxendale, Phyllis B (1958). Machine-made index for technical literature: an experiment. *IBM Journal of Research and Development*, 2(4):354–361.

Bengio, Yoshua, Réjean Ducharme, Pascal Vincent & Christian Jauvin (2003). A neural probabilistic language model. *journal of machine learning research*, 3(Feb):1137–1155.

Blei, David, Lawrence Carin & David Dunson (2010). Probabilistic topic models. *IEEE signal processing magazine*, 27(6):55–65.

Blei, David M & John D Lafferty (2006). Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pp. 113–120, ACM.

Blei, David M & John D Lafferty (2009). Topic models. *Text mining: classification, clustering, and applications*, 10(71):34.

Blei, David M, Andrew Y Ng & Michael I Jordan (2003). Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Boguraev, Branimir & Christopher Kennedy (1999). Salience-based content characterisation of text documents. In Inderjeet Mani & Mark T. Maybury (Eds.), *Advances in Automatic Text Summarization*, pp. 99–110. Cambridge, Mass.: MIT Press.

Borko, Harold & Charles L Bernier (1975). Abstracting concepts and methods.

Brin, Sergey & Lawrence Page (1998). The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117.

Cai, Jie & Michael Strube (2010). End-to-end coreference resolution via hypergraph partitioning. In *Proceedings of the 23rd International Conference on Computational Linguistics,* Beijing, China, 23–27 August 2010, pp. 143–151.

Cao, Ziqiang, Furu Wei, Sujian Li, Wenjie Li, Ming Zhou & Houfeng Wang (2015). Learning summary prior representation for extractive summarization. *Proceedings of ACL: short papers*, pp. 829–833.

Carbonell, Jaime G. & Jade Goldstein (1998). The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval,* Melbourne, Australia, 24–28 August 1998, pp. 335–336.

Casella, George & Edward I George (1992). Explaining the gibbs sampler. *The American Statistician*, 46(3):167–174.

Celikyilmaz, Asli & Dilek Hakkani-Tür (2010). A hybrid hierarchical model for multi-document summarization. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics,* Uppsala, Sweden, 11–16 July 2010, pp. 815–824.

Cheng, Jianpeng & Mirella Lapata (2016). Neural summarization by extracting sentences and words. pp. 484–494.

Chib, Siddhartha & Edward Greenberg (1995). Understanding the metropolis-hastings algorithm. *The american statistician*, 49(4):327–335.

Christensen, Janara, Mausam, Stephen Soderland & Oren Etzioni (2013). Towards coherent multi-document summarization. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* Atlanta, Georgia, 9–14 June 2013, pp. 1163–1173.

Cohan, Arman & Nazli Goharian (2015). Scientific article summarization using citation-context and articleâs discourse structure. *Empirical Methods in Natural Language Processing (EMNLP â15)*.

Contractor, Danish, Yufan Guo & Anna Korhonen (2012). Using argumentative zones for extractive summarization of scientific articles. In *Proceedings of the 24th International Conference on Computational Linguistics,* Mumbai, India, 8–15 December 2012, pp. 663–678.

Daneš, František (Ed.) (1974). *Papers on Functional Sentence Perspective*. Prague: Academia.

Dang, Hoa Trang (2005). Overview of DUC 2005. In *Proceedings of the 2005 Document Understanding Conference held at the Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing,* Vancouver, B.C., Canada, 9–10 October 2005.

De Lathauwer, L, B De Moor, J Vandewalle & Blind Source Separation by Higher-Order (1994). Singular value decomposition. In *Proc. EUSIPCO-94, Edinburgh, Scotland, UK*, Vol. 1, pp. 175–178.

Dhillon, Inderjit S, Yuqiang Guan & Brian Kulis (2007). Weighted graph cuts without eigenvectors a multilevel approach. *IEEE transactions on pattern analysis and machine intelligence*, 29(11):1944–1957.

Dumais, Susan T (2004). Latent semantic analysis. *Annual review of information science and technology*, 38(1):188–230.

Edmundson, H.P. (1969). New methods in automatic extracting. *Journal of the Association for Computing Machinery*, 16(2):264–285. Reprinted in *Advances in Automatic Text Summarization*, Mani, I. and Maybury, M.T. (Eds.), Cambridge, Mass.: MIT Press, 1999, pp.21-42.

Elkan, Charles & Keith Noto (2008). Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* Las Vegas, Nev., 24–27 August 2008, pp. 213–220.

Elsner, Micha & Eugene Charniak (2011). Extending the entity grid with entity-specific features. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers),* Portland, Oreg., 19–24 June 2011, pp. 125–129.

Ercan, Gonenc & Ilyas Cicekli (2008). Lexical cohesion based topic modeling for summarization. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pp. 582–592, Springer.

Erkan, Güneş & Dragomir R. Radev (2004). LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22:457–479.

Galanis, Dimitrios, Gerasimos Lampouras & Ion Androutsopoulos (2012). Extractive multi-document summarization with integer linear programming and support vector regression. In *Proceedings of the 24th International Conference on Computational Linguistics,* Mumbai, India, 8–15 December 2012, pp. 911–926.

Galley, Michel & Kathleen McKeown (2003). Improving word sense disambiguation in lexical chaining. In *IJCAI*, Vol. 3, pp. 1486–1488.

García-Hernández, René Arnulfo, Yulia Ledeneva, Griselda Matías Mendoza, Ángel Hernández Dominguez, Jorge Chavez, Alexander Gelbukh & José Luis Tapia Fabela (2009). Comparing commercial tools and state-of-the-art methods for generating text summaries. In *Proceedings of Advances in Artificial Intelligence, 8th Mexican International Conference on Artificial Intelligence,* Guanajuato, Mexico, 9-13 November 2009, pp. 92–96.

Gillick, Daniel, Korbinian Riedhammer, Benoit Favre & Dilek Hakkani-Tür (2009). A global optimization framework for meeting summarization. In *Proceedings of the 2009 IEEE International Conference on Acoustics, Speech, and Signal Processing,* Taipei, Taiwan, 19–24 June 2009, pp. 4769–4772.

Gong, Yihong & Xin Liu (2001). Generic text summarization using relevance measure and latent semantic analysis. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,* New Orleans, Louis., 9–12 September 2001, pp. 19–25.

Gorinski, Philip John & Mirella Lapata (2015). Movie script summarization as graph-based scene extraction. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies,* Denver, Col., 31 May – 5 June 2015, pp. 1066–1076.

Griffiths, Thomas L, Mark Steyvers, David M Blei & Joshua B Tenenbaum (2004). Integrating topics and syntax. In *Advances in neural information processing systems*, pp. 537–544.

Guinaudeau, Camille & Michael Strube (2013). Graph-based local coherence modeling. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers),* Sofia, Bulgaria, 4–9 August 2013, pp. 93–103.

Guo, Yufan, Roi Reichart & Anna Korhonen (2015). Unsupervised declarative knowledge induction for constraint-based learning of information structure in scientific documents. *Transactions of the Association for Computational Linguistics*, 3:131–143.

Gurobi Optimization, Inc. (2014). *Gurobi Optimizer Reference Manual*.

Haghighi, Aria & Lucy Vanderwende (2009). Exploring content models for multi-document summarization. In *Proceedings of Human Language Technologies 2009: The Conference of the North American Chapter of the Association for Computational Linguistics,* Boulder, Col., 31 May – 5 June 2009, pp. 362–370.

Halliday, M. A. K. & Ruqaiya Hasan (1976). *Cohesion in English*. London, U.K.: Longman.

Harris, Zellig S (1954). Distributional structure. *Word*, 10(2-3):146–162.

Herre, Heinrich (2015). Persistence, change, and the integration of objects and processes in the framework of the general formal ontology. In Vesselin Petrov & Adam C. Scarfe (Eds.), *Dynamic Being*, pp. 337–354. Cambridge Scolar Publishing.

Hirao, Tsutomu, Yasuhisa Yoshida, Masaaki Nishino, Norihito Yasuda & Masaaki Nagata (2013). Single-document summarization as a tree knapsack problem. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing,* Seattle, Wash., 18–21 October 2013, pp. 1515–1520.

Jha, Rahul, Reed Coke & Dragomir Radev (2015). Surveyor: A system for generating coherent survey articles for scientific topics. In *Proceedings of the 29th Conference on the Advancement of Artificial Intelligence,* Austin, Texas, 25–30 January 2015, pp. 2167–2173.

Jin, Feng, Minlie Huang & Xiaoyan Zhu (2010). A comparative study on ranking and selection strategies for multi-document summarization. In *Proceedings of Coling 2010: Poster Volume,* Beijing, China, 23–27 August 2010, pp. 525–533.

Joseph, Mark Thomas & Dragomir R Radev (2007). Citation analysis, centrality, and the acl anthology. *Ann Arbor*, 1001:48109–1092.

Klein, Dan & Christopher D. Manning (2003). Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics,* Sapporo, Japan, 7–12 July 2003, pp. 423–430.

Kleinberg, Jon M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632.

Kobayashi, Hayato, Masaki Noguchi & Taichi Yatsuka (2015). Summarization based on embedding distributions. *Proceedings of the 2015 EMNLP*, pp. 1984–1989.

Kupiec, Julian, Jan Pedersen & Francine Chen (1995). A trainable document summarizer. In *Proceedings of the 18th ACM-SIGIR Conference,* Seattle, Wash., 1995, pp. 68–73.

Lawrie, Dawn, W Bruce Croft & Arnold Rosenberg (2001). Finding topic words for hierarchical summarization. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 349–357, ACM.

Lerouge, Julien, Pierre Le Bodic, Pierre Héroux & Sébastien Adam (2015). GEM++: A tool for solving substitution-tolerant subgraph isomorphism. In C.-L. Liu, B. Luo, W.G. Kropatsch & J. Cheng (Eds.), *Graph-Based Representations in Pattern Recognition*, pp. 128–137. Heidelberg, Germany: Springer.

Li, Sujian, You Ouyang, Wei Wang & Bin Sun (2007). Multi-document summarization using support vector regression. In *Proceedings of DUC*, Citeseer.

Liakata, Maria, Simon Dobnik, Shyamasree Saha, Colin Batchelor & Dietrich Rebholz-Schuhmann (2013). A discourse-driven content model for summarising scientific articles evaluated in a complex question answering task. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing,* Seattle, Wash., 18–21 October 2013, pp. 747–757.

Lin, Chin-Yew (2004). ROUGE: A package for automatic evaluation of summaries. In *Proceedings of the Text Summarization Branches Out Workshop at ACL '04,* Barcelona, Spain, 25–26 July 2004, pp. 74–81.

Lin, Chin-Yew & Eduard Hovy (2000). The automated acquisition of topic signatures for automatic summarization. In *Proceedings of the 18th International Conference on Computational Linguistics,* Saarbrücken, Germany, 31 July – 4 August 2000, pp. 495–501.

Liu, Yan, Sheng-hua Zhong & Wenjie Li (2012). Query-oriented multi-document summarization via unsupervised deep learning. In *AAAI*.

Louis, Annie, Aravind Joshi & Ani Nenkova (2010). Discourse indicators for content selection in summarization. In *Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pp. 147–156, Association for Computational Linguistics.

Luhn, H.P. (1958). The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2:159–165.

Mani, Inderjeet (2001). *Automatic summarization*, Vol. 3. John Benjamins Publishing.

Mani, Inderjeet & Mark T. Maybury (Eds.) (1999). *Advances in Automatic Text Summarization*. Cambridge, Mass.: MIT Press.

Mann, William C & Sandra A Thompson (1988). Rhetorical structure theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

Marcu, Daniel (1997a). From local to global coherence: A bottom-up approach to text planning. In *Proceedings of the 14th National Conference on Artificial Intelligence,* Providence, R.I., 27–31 July 1997, pp. 629–635.

Marcu, Daniel (1997b). The rhetorical parsing of natural language texts. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and of the 8th Conference of the European Chapter of the Association for Computational Linguistics,* Madrid, Spain, 7–12 July 1997, pp. 365–372.

Martschat, Sebastian (2013). Multigraph clustering for unsupervised coreference resolution. In *51st Annual Meeting of the Association for Computational Linguistics: Proceedings of the Student Research Workshop,* Sofia, Bulgaria, 5–7 August 2013, pp. 81–88.

Martschat, Sebastian, Jie Cai, Samuel Broscheit, Éva Mújdricza-Maydt & Michael Strube (2012). A multigraph model for coreference resolution. In *Proceedings of the Shared Task of the 16th Conference on Computational Natural Language Learning,* Jeju Island, Korea, 12–14 July 2012, pp. 100–106.

McDonald, Ryan (2007). A study of global inference algorithms in multi-document summarization. In *Proceedings of the European Conference on Information Retrieval,* Rome, Italy, 2-5 April 2007.

Mesgar, Mohsen & Michael Strube (2014). Normalized entity graph for computing local coherence. In *Proceedings of TextGraphs-9: Graph-based Methods for Natural Language Processing, Workshop at EMNLP 2014,* Doha, Qatar, 29 October 2014, pp. 1–5.

Mesgar, Mohsen & Michael Strube (2015). Graph-based coherence modeling for assessing readability. In *Proceedings of STARSEM 2015: The Fourth Joint Conference on Lexical and Computational Semantics,* Denver, Col., 4–5 June 2015, pp. 309–318.

Mihalcea, Rada & Paul Tarau (2004). TextRank: Bringing order into texts. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing,* Barcelona, Spain, 25–26 July 2004, pp. 404–411.

Moon, Todd K (1996). The expectation-maximization algorithm. *IEEE Signal processing magazine*, 13(6):47–60.

Myaeng, S.H. & D.-H. Jang (1999). Development and evaluation of a statistically-based document summarization system. In Inderjeet Mani & Mark T. Maybury (Eds.), *Advances in Automatic Text Summarization*, pp. 61–70. Cambridge, Mass.: MIT Press.

Nastase, Vivi (2008). Topic-driven multi-document summarization with encyclopedic knowledge and spreading activation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 763–772, Association for Computational Linguistics.

Nishikawa, Hitoshi, Takaaki Hasegawa, Yoshihiro Matsuo & Genichiro Kikui (2010). Opinion summarization with integer linear programming formulation for sentence extraction and ordering. In *Proceedings of the 23rd International Conference on Computational Linguistics,* Beijing, China, 23–27 August 2010, pp. 910–918.

Page, Lawrence, Sergey Brin, Rajeev Motwani & Terry Winograd (1998). *The PageRank citation ranking: Bringing order to the web*. Technical Report: Stanford University, Stanford, Cal.

Parveen, Daraksha, Mohsen Mesgar & Michael Strube (2016). Generating coherent summaries using coherence patterns.

Parveen, Daraksha, Hans-Martin Ramsl & Michael Strube (2015). Topical coherence for graph-based extractive summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing,* Lisbon, Portugal, 17–21 September 2015, pp. 1949–1954.

Parveen, Daraksha & Michael Strube (2014). Multi-document summarization using bipartite graphs. In *Proceedings of TextGraphs-9: Graph-based Methods for Natural Language Processing, Workshop at EMNLP 2014,* Doha, Qatar, 29 October 2014, pp. 15–24.

Parveen, Daraksha & Michael Strube (2015). Integrating importance, non-redundancy and coherence in graph-based extractive summarization. In *Proceedings of the 24th International Joint Conference on Artificial Intelligence,* Buenos Aires, Argentina, 25–31 July 2015, pp. 1298–1304.

Pollock, Joseph J & Antonio Zamora (1975). Automatic abstracting research at chemical abstracts service. *Journal of Chemical Information and Computer Sciences*, 15(4):226–232.

Porter, Martin F (2001). *Snowball: A language for stemming algorithms*.

Prasad, Rashmi, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind K Joshi & Bonnie L Webber (2008). The penn discourse treebank 2.0. In *LREC*, Citeseer.

Qazvinian, Vahed & Dragomir R. Radev (2008). Scientific paper summarization using citation summary networks. In *Proceedings of the 22nd International Conference on Computational Linguistics,* Manchester, U.K., 18–22 August 2008, pp. 689–696.

Radev, Dragomir, Timothy Allison, Sasha Blair-Goldensohn, John Blitzer, Arda Celibi, Stanko Dimitrov, Elliott Drabek, Ali Hakim, Wai Lam, Danyu Liu, Jahna Otterbacher, Hong Qi, Horacio Saggion, Simone Teufel, Michael Topper, Adam Winkel & Zhu Zhang (2004a). MEAD – a platform for multidocument multilingual text summarization. In *Proceedings of the 4th International Conference on Language Resources and Evaluation,* Lisbon, Portugal, 26–28 May 2004.

Radev, Dragomir R., Hongyan Jing, Ma
lgorzata Styś & Daniel Tam (2004b). Centroid-based summarization of multiple documents. *Information Processing and Management*, 40:919–938.

ramsl, Hans-martin (2015). Levaraging topic models for graphical local coherence representation.

Reeve, Lawrence H, Hyoil Han & Ari D Brooks (2007). The use of domain-specific concepts in biomedical text summarization. *Information Processing & Management*, 43(6):1765–1776.

Schluter, Natalie & Anders Søgaard (2015). Unsupervised extractive summarization via coverage maximization with syntactic and semantic concepts. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, Vol. 2, pp. 840–844.

Siegel, Sidney & N. John Castellan (1988). *Nonparametric Statistics for the Behavioral Sciences* (2nd ed.). New York: McGraw-Hill.

Smith, Barry & Christiane Fellbaum (2004). Medical wordnet: a new methodology for the construction and validation of information resources for consumer health. In *Proceedings of the 20th international conference on Computational Linguistics*, p. 371, Association for Computational Linguistics.

Strzalkowski, Tomek, Jin Wang & Bowden Wise (1998). A robust practical text summarization. In *Proceedings of the AAAI Symposium on Intelligent Text Summarization*, pp. 26–33.

Teh, Yee Whye, Michael I Jordan, Matthew J Beal & David M Blei (2012). Hierarchical dirichlet processes. *Journal of the american statistical association*.

Temperley, Davy, Daniel Sleator & John Lafferty (1991). *Link grammar*.

Teufel, Simone & Marc Moens (1999). Argumentative classification of extracted sentences as a first step towards flexible abstracting. *Advances in automatic text summarization*, 155:1–171.

Teufel, Simone & Marc Moens (2002). Summarizing scientific articles: Experiments with relevance and rhetorical status. *Computational Linguistics*, 28(4):409–445.

Teufel, Simone, Advaith Siddharthan & Dan Tidhar (2006). Automatic classification of citation function. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing,* Sydney, Australia, 22–23 July 2006, pp. 103–110.

Toutanova, Kristina, Chris Brockett, Michael Gamon, Jagadeesh Jagarlamudi, Hisami Suzuki & Lucy Vanderwende (2007). The PYTHY summarization system: Microsoft Research at DUC 2007. In *Proceedings of the 2007 Document Understanding Conference held at the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics,* Rochester, N.Y., 26–27 April 2007.

Wallach, Hanna M (2006). Topic modeling: beyond bag-of-words. In *Proceedings of the 23rd international conference on Machine learning*, pp. 977–984, ACM.

Wan, Xiaojun (2010). Towards a unified approach to simultaneous single-document and multi-document summarizations. In *Proceedings of the 23rd international conference on computational linguistics*, pp. 1137–1145, Association for Computational Linguistics.

Wan, Xiaojun & Jianguo Xiao (2010). Exploiting neighborhood knowledge for single document summarization and keyphrase extraction. *ACM Transactions on Information Systems*, 28(2):8 pages.

Wolfram, Dietmar (2003). *Applied informetrics for information retrieval research*. Greenwood Publishing Group.

Woodsend, Kristian & Mirella Lapata (2010). Automatic generation of story highlights. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp. 565–574, Association for Computational Linguistics.

Xu, Han, Eric Martin & Ashesh Mahidadia (2015). Extractive summarisation based on keyword profile and language model. *NAACL*.

Yan, Xifeng & Jiawei Han (2002). gSpan: Graph-based substructure pattern mining. In *Proceedings of the International Conference on Data Mining,* Maebashi City, Japan, 9–12 December 2002, pp. 721–724.

Yogatama, Dani, Fei Liu & Noah A Smith (2015). Extractive summarization by maximizing semantic volume. In *Conference on Empirical Methods in Natural Language Processing*.

Appendix A

Summary S_1 of a *PLOS Medicine* article (*journal.pmed.1001622*) by *Egraph*+$\mathbf{CP_3}$

S_1 Globally, suicide is amongst the leading causes of premature mortality; in 2010, it was the fifth leading cause of death in women and the sixth in men among individuals aged 15-49 y.

S_2 Time trends in suicide rates may be influenced by a number of factors including socio-economic changes, the prevalence of mental illness or distress, and certain types of media reporting; there is growing evidence that changes in the popularity and availability of lethal suicide methods could also have a marked impact on time trends in overall suicide rates.

S_3 Previous studies of method availability and suicide have mostly focused on the impact of restricting access to methods, such as detoxification of domestic gas, bans on sales of toxic pesticides and legal changes in firearms regulations.

S_4 In 1998-2000 there was a rapid rise in suicide by carbon monoxide poisoning from the inhalation of barbecue charcoal gas in Hong Kong and Taiwan.

S_5 Suicides by this method used to be very rare, but within 5 y charcoal burning became the second most common method of suicide in these two countries.

S_6 We used data from eight East/Southeast Asian countries Hong Kong, Taiwan, Japan, the Republic of Korea, Singapore, Malaysia, the Philippines, and Thailand to investigate time trends in charcoal-burning suicide across different countries and the association between changes in charcoal-burning suicide and overall suicide rates for the years 1995-2011.

S_7 We also examined sex - and age-specific time trends to identify the demographic groups showing the greatest increases in charcoal-burning suicide rates across different countries.

S_8 Specifically, the objectives of this analysis were to investigate (i) time trends and regional patterns of charcoal-burning suicide throughout East/Southeast Asia during the

period 1995-2011 and (ii) whether any rises in use of this method were associated with increases in overall suicide rates.

S_9 Sex - and age-specific trends over time were also examined to identify the demographic groups showing the greatest increases in charcoal-burning suicide rates across different countries.

S_{10} The World Health Organization Mortality Database contained suicide data for nine of countries countries, but only six had method-specific data available.

S_{11} Data for the 3-y period 1995-1997 prior to 1998, when the first widely publicised suicide by charcoal burning occurred were used to assess the baseline rates.

S_{12} There is no specific code for charcoal-burning suicide in the International Classification of Diseases.

S_{13} We did not have access to data from other countries to investigate this further.

S_{14} In joinpoint regression analysis, suicide time trends are characterised by contiguous linear segments and " join points " points at which trends change.

S_{15} Crude rates for Singapore were modelled similarly, as age-specific suicide data were unavailable.

S_{16} We calculated incidence rate ratios assuming a linear change in rates.

S_{17} Charcoal-burning suicide rates increased in all countries over the study period, but the magnitude of the rise varied by country.

S_{18} However, magnitude should be noted that these estimates are sensitive to baseline rates.

S_{19} Similar declines in other methods of suicide in Hong Kong and Taiwan were also observed after 2003.

S_{20} Combined numbers of charcoal-burning suicides for the five study countries reached a peak in 2009 (n = 6,759).

S_{21} Similarly, in Taiwan, the rise in charcoal-burning suicide in 2000-2006 was related to an increase in overall suicide rate over the same period.

S_{22} There was no evidence for an association of time trends in the rate of charcoal-burning suicide with changes in the overall suicide rate in Singapore.

S_{23} In countries with a rise in the charcoal-burning suicide rate, the timing, scale, and sex/age pattern of the increase varied by country.

S_{24} Our data showed that the increases in charcoal-burning suicide were associated with various levels of changes in overall suicide rates across the East/Southeast Asian countries studied.

S_{25} In contrast , Singapore had a much smaller rise in charcoal-burning suicide than other countries did.

S_{26} There are several limitations to this study.

S_{27} Our main analyses included both suicides and deaths coded as undetermined intent, and findings were similar when data only for certified suicides were used.

S_{28} Suicide estimates in countries five countries are considered to be reliable according to the rating scheme of the World Health Organization.

S_{29} Suicide statistics are subject to under-reporting and misclassification in Malaysia, the Philippines, and Thailand, where the quality of suicide registration is not satisfactory.

S_{30} In Taiwan, the rise and fall of charcoal-burning suicide did not seem to be associated with economic conditions.

S_{31} Our results have several implications for international and regional suicide prevention strategies.

Summary S_2 of a *PLOS Medicine* article (*journal.pmed.1001622*) by *Lead*

S_1 Globally, suicide is amongst the leading causes of premature mortality ; in 2010, it was the fifth leading cause of death in women and the sixth in men among individuals aged 1549 y.

S_2 Time trends in suicide rates may be influenced by a number of factors including socio-economic changes, the prevalence of mental illness or distress, and certain types of media reporting ; there is growing evidence that changes in the popularity and availability of lethal suicide methods could also have a marked impact on time trends in overall suicide rates.

S_3 Previous studies of method availability and suicide have mostly focused on the impact of restricting access to methods, such as detoxification of domestic gas, bans on sales of toxic pesticides and legal changes in firearms regulations.

S_4 However, many suicides using these methods had already occurred before the implementation of restrictions, highlighting the potential importance of surveillance for the emergence of new suicide methods at an early stage to enable public health action to prevent an increase of suicide by new methods.

S_5 In 1998-2000 there was a rapid rise in suicide by carbon monoxide poisoning from the inhalation of barbecue charcoal gas in Hong Kong and Taiwan.

S_6 Suicides by this method used to be very rare, but within 5 y charcoal burning became the second most common method of suicide in these two countries.

S_7 Although cases of charcoal-burning suicide have been reported in other neighbouring East/Southeast Asian countries such as China, Japan, Macao, Malaysia, Singapore, and the Republic of Korea, to the best of our knowledge, there has been no systematic investigation of regional patterns and time trends in the use of this method and the association between time trends in charcoal-burning suicide and overall suicide rates in affected countries.

S_8 We used data from eight East/Southeast Asian countries Hong Kong, Taiwan, Japan, the Republic of Korea, Singapore, Malaysia, the Philippines, and Thailand to investigate time trends in charcoal-burning suicide across different countries and the association between changes in charcoal-burning suicide and overall suicide rates for the years 1995-2011.

S_9 We also examined sex - and age-specific time trends to identify the demographic groups showing the greatest increases in charcoal-burning suicide rates across different countries.

S_{10} Our overall aim was to establish what can be learned from the changing incidence of charcoal-burning suicide in this region to inform the prevention of the future emergence

of novel suicide methods.

S_{11} Specifically, the objectives of this analysis were to investigate (i) time trends and regional patterns of charcoal-burning suicide throughout EastSoutheast Asia during the period 1995-2011 and (ii) whether any rises in use of this method were associated with increases in overall suicide rates.

S_{12} Sex - and age-specific trends over time were also examined to identify the demographic groups showing the greatest increases in charcoal-burning suicide rates across different countries.

S_{13} The study used only aggregate secondary data that were available openly ; no identifiable personal data were used in the study.

S_{14} Ethical approval was thus not required.

S_{15} To investigate time trends in charcoal-burning suicide in EastSoutheast Asia we first systematically identified countries with data available in the World Health Organization WHO Mortality Database, which provides the most comprehensive standardised national mortality statistics for countries around the world.

S_{16} Figure 1 shows a flow chart summarising how we identified data for the study countries.

S_{17} In brief, we first identified 19 countries that were classified as in the EastSoutheast Asia region by the United Nations eight in East Asia and 11 in Southeast Asia.

S_{18} The WHO Mortality Database contained suicide data for nine of these countries, but only six had method-specific data available.

S_{19} We then extracted complete method-specific suicide data by sex, age 5-y bands, and year for Japan and the Republic of Korea for the period 19952011, and for the years available for Hong Kong 2001-2011, Malaysia 2000-2008, the Philippines 1995-2003, 2008, and Thailand 1995-2000, 2002-2006.

S_{20} Data for the 3-y period (1995-1997) prior to 1998, when the first widely publicised suicide by charcoal burning occurred were used to assess the baseline rates.

S_{21} We then supplemented the WHO data by extracting relevant suicide data from the national death registers for Hong Kong (1995â2011) and Taiwan (1995â2011), as well as from published mortality statistics for Singapore (1996â2011), although only sex- and method-specific, but not age-specific, data were available for Singapore.

Summary S_3 of a *PLOS Medicine* article (*journal.pmed.1001622*) by *TextRank*

S_1 Some evidence of method substitution was found in Japanese males – the rise in charcoal-burning suicide was accompanied by a fall in suicide by other methods Figure 2.

S_2 There was no evidence for an association of time trends in the rate of charcoal-burning suicide with changes in the overall suicide rate in Singapore.

S_3 A rise in charcoal-burning suicides was first seen in Hong Kong 1999, followed by Singapore 2000, Taiwan 2001, Japan 2003, and the Republic of Korea 2008, although the evidence for a definite starting year for Singapore was limited because of relatively small suicide numbers.

S_4 The WHO Mortality Database also provided population data; when these were incomplete or unavailable, relevant data were extracted from the United Nations population database.

S_5 Suicide estimates in these five countries are considered to be reliable according to the rating scheme of the WHO.

S_6 Negative binomial regression models were used because there was evidence for over-dispersion in the Poisson regression analyses of the data.

S_7 Charcoal-burning suicide rates increased in all countries over the study period, but the magnitude of the rise varied by country.

S_8 There are several limitations to this study.

S_9 In contrast, the quality of suicide data for countries with incomplete time series Malaysia, the Philippines, and Thailand is thought to be relatively poor.

S_{10} Annual rates of changes in charcoal-burning suicide rates did not differ by sex/age group in Taiwan and Hong Kong, whilst people aged 1524 y in Japan and people aged 25-64 y in the Republic of Korea tended to have the greatest rates of increase.

S_{11} Malaysia, the Philippines, and Thailand also use different languages.

S_{12} The economic recession, which led to an increased number of people troubled by debt problems, may have had some role in the increase in charcoal-burning suicide.

S_{13} Although there was no indication of a marked rise in charcoal-burning suicide rate in these three countries, a very slight rise in Malaysia was observed the rate increased from 006 per 100,000 in 2000 to a peak of 030 per 100,000 in 2003.

S_{14} Males aged 25-44 y tended to show the highest rates compared to other sex/age groups, except that Japanese males aged 4564 y had rates similar to those of males aged 25-44 y In contrast, females aged 65+ y tended to have the lowest rates, except in the Republic of Korea, where females aged 45-64 y had the lowest rates.

S_{15} Furthermore, although economic slowdowns may be accompanied by rises in suicide, the impact is not likely to be method-specific ie, affecting only trends in charcoal-burning suicide but not suicide using other methods.

S_{16} Third, we did not include data for some East/Southeast Asian countries where cases of charcoal-burning suicide were also reported recently, such as China and Macao; detailed method-specific data for suicide were unavailable for these countries.

S_{17} Our results have several implications for international and regional suicide prevention strategies.

S_{18} Time trends in suicide rates may be influenced by a number of factors including socio-economic changes, the prevalence of mental illness or distress, and certain types of media reporting; there is growing evidence that changes in the popularity and availability of lethal suicide methods could also have a marked impact on time trends in overall suicide rates.

S_{19} In Hong Kong, charcoal-burning suicides emerged in 1998-1999, following the Asian economic crisis in 1997-1998, which was shown to have a strong impact on Hong Kong's economy and suicide patterns.

S_{20} This may be related to the characteristics of the initial cases.

S_{21} Similarly, in Taiwan, the rise in charcoal-burning suicide in 2000-2006 was related to an increase in overall suicide rate over the same period.

S_{22} In the Republic of Korea, the increase in charcoal-burning suicide was quite recent and was not associated with a rise in the overall suicide rate, as the magnitude of increase was relatively small.

S_{23} The sequence started by comparing the model with zero join points i.e., a straight line with no change in trend and that with one join point, and it ended when there was no statistical evidence that more joint points fit the data better or when reaching the maximum number of join points allowed.

S_{24} The starting and peak years were identified in the joinpoint regression analyses and by visual inspection of the graphs of time trends in charcoal-burning suicide.