TEXT SUMMARIZATION USING LATENT SEMANTIC ANALYSIS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

MAKBULE GÜLÇİN ÖZSOY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

FEBRUARY 2011

Approval of the thesis:

TEXT SUMMARIZATION USING LATENT SEMANTIC ANALYSIS

submitted by **MAKBULE GÜLÇİN ÖZSOY** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan ÖZGEN _____
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Adnan YAZICI _____
Head of Department, **Computer Engineering**

Assoc. Prof. Dr. Ferda Nur ALPASLAN _____
Supervisor, **Computer Engineering Dept., METU**

Prof. Dr. İlyas ÇIÇEKLI _____
Co-Supervisor, **Computer Eng. Dept., Hacettepe Uni.**

Examining Committee Members:

Prof. Dr. Fazlı CAN _____
Computer Engineering Dept., Bilkent University

Assoc. Prof. Dr. Ferda Nur ALPASLAN _____
Computer Engineering Dept., METU

Assoc. Prof. Dr. Cem BOZŞAHIN _____
Computer Engineering Dept., METU

Assoc. Prof. Dr. Nihan ÇIÇEKLI _____
Computer Engineering Dept., METU

Dr. Ruken ÇAKICI _____
Computer Engineering Dept., METU

Date: 02.02.2011

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name : **Makbule Gülçin ÖZSOY**

Signature :

ABSTRACT

TEXT SUMMARIZATION USING LATENT SEMANTIC ANALYSIS

Özsoy, Makbule Gülçin
M.Sc., Department of Computer Engineering
Supervisor: Assoc. Prof. Dr. Ferda Nur Alpaslan
Co-Supervisor: Prof. Dr. İlyas Çiçekli

February 2011, 69 pages

Text summarization solves the problem of presenting the information needed by a user in a compact form. There are different approaches to create well formed summaries in literature. One of the newest methods in text summarization is the Latent Semantic Analysis (LSA) method. In this thesis, different LSA based summarization algorithms are explained and two new LSA based summarization algorithms are proposed. The algorithms are evaluated on Turkish and English documents, and their performances are compared using their ROUGE scores.

Keywords: Text Summarization, Latent Semantic Analysis

ÖZ

GİZİL ANLAMSAL ANALİZ YÖNTEMİ İLE DOKÜMAN ÖZETİ ÇIKARMA

Özsoy, Makbule Gülçin
Yüksek Lisans, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi: Doç. Dr. Ferda Nur Alpaslan
Ortak Tez Yöneticisi: Prof. Dr. İlyas Çiçekli

Şubat 2011, 69 sayfa

Özet çıkarma sistemleri, kullanıcının ihtiyacı olan bilginin sıkıştırılarak sunulması ihtiyacını çözer. Literatürde doğru biçimde oluşturulmuş özetler çıkarmak için farklı yaklaşımlar mevcuttur. Bu yaklaşımların en yenilerinden biri de Gizil Anlamsal Analiz (Latent Semantic Analysis) metodudur. Bu tezde, Gizil Anlamal Analiz tabanlı farklı özetçıkarma algoritmaları açıklanmış ve iki yeni Gizil Anlamsal Analiz tabanlı algoritma sunulmuştur. Algoritmalar Türkçe ve İngilizce veri setleri kullanılarak test edilmiş ve performans sonuçları ROUGE değerleri kullanılarak karşılaştırılmıştır.

Anahtar Kelimeler: Özet Çıkarma Sistemleri, Gizil Anlamsal Analiz (Latent Semantic Analysis)

To My Family

ACKNOWLEDGMENTS

I would like to express my greatest appreciations to my supervisor Assoc. Prof. Dr. Ferda Nur Alpaslan and my co-supervisor Prof. Dr. İlyas Çiçekli, for their encouragement, guidance and valuable advices.

I would like to express my special thanks to my family and my friends for their support and patience.

In conclusion, I recognize that this research would not have been possible without the support of my employer, TÜBİTAK- BİLGEM/UEKAE.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

CHAPTER 1

INTRODUCTION

The goals of this thesis are understanding automatic text summarization approaches in literature, developing new approaches using latent semantic analysis based algorithms and applying these approaches on Turkish and English document sets to understand whether the approaches are language independent or not.

In this chapter, motivation of the research and goals of the thesis is given. At the end, the outline of the work is introduced.

1.1 Motivation

The growth in electronically available documents makes research and applications in automatic text summarization more and more important. Huge number of available documents in digital media makes it difficult to obtain the necessary information related to the needs of a user. In order to solve this issue, text summarization systems can be used. The text summarization systems extract brief information from a given document while preserving important concepts of that document. By using the summary produced, a user can decide if a document is related to his/her needs without reading the whole document. Also other systems, such as search engines, news portals etc., can use document summaries to perform their jobs more efficiently.

The aspects of a summary are defined as following, in the study of (Radev, Hovy and McKeown 2002) and (Das and Martins 2007):

- The summary can be created using single or multiple documents.

- The summary contains all necessary information and it does not include redundant information.

- The summary is short, at least shorter than the half of the original document.

Besides aspects mentioned above, there are issues that should be cared while creating summaries. The first aspect is the cohesion of the summary. Cohesion is related to the surface level structure of the text. It can be defined as grammatical and lexical structures that link text parts to each other by using pronouns, conjunctions, time references and so on. The second aspect is coherence. Coherence is about the semantic level structure of the document. It is hard to model and it needs understanding of the input text. One of the goals of automatic text summarization systems is to create cohesive and coherent summaries.

Text summarization systems can be categorized as extractive or abstractive according to the way the summary is created. In extractive summarization approaches, the goals are identifying most important concepts in the input document, and giving related sentences found in the document as an output. The summary created using these sentences may not be coherent, but gives idea about the content of the input document. In abstractive summarization approaches, first the system understands the texts and then it creates summaries with its own words. The abstractive summarization is similar to the way a person creates a summary. While creating a summary, a person uses his/her prior knowledge; but this is a challenging task for a computer system. Abstractive summarization remains as a difficult task in natural language processing.

Another categorization aspect of text summarization systems is about the number of documents used as input. The categories according to the number of input documents are single-document summarization and multi-document summarization. While a single document is used as an input to create a single summary in single-document

text summarization systems; multiple documents related to a single subject are used in multi-document text summarization systems.

Another categorization method of text summarization systems is based on the purpose of summary. The categories which are based on the purpose are named as generic summarization system and query-based summarization system. In generic text summarization systems, the summary created is about whole document. In query-based text summarization systems, the created summary is about the query asked.

Last categorization of the text summarization systems is based on the approaches used in the summarization algorithms. There are different algorithms in literature which are based on supervised or unsupervised techniques.

The first studies related to document summarization started in late fifties. These studies were based on surface level information. Then statistical approaches, more semantic oriented analysis such as lexical chains, algebraic based methods such as Latent Semantic Analysis (LSA) are used in text summarization systems.

Latent Semantic Analysis (LSA) is algebra based unsupervised method. The method is used in information retrieval for document classification, document segmentation etc. LSA has the ability to find out meaning relations among words and among sentences in the input document. In text summarization, it is used for finding out the concepts and identifying representative sentences of the concepts. Those sentences that are related to the important concepts are collected as a part of the output summary in text summarization algorithms.

Evaluation of summaries is an active research area in natural language processing. There exist different methods for the evaluation, such as using human evaluators, using precision/recall values, or using ROUGE scores (Lin 2004).

1.2 Thesis Goals

The Latent Semantic Analysis (LSA) method can extract the meaning of words and

sentences using only the input document, without any external information. It also has the ability of finding out the concepts in the input document. To perform the summarization based on LSA, first input matrix is created, then LSA related calculations are done, and lastly sentences are selected as a part of summary.

The major problems addressed in this thesis are as following:

1. Different weighting approaches give different results in summarization systems. In this thesis, different approaches for input matrix creation are explored.

2. Different approaches for sentence selection gives different results in summary. In this thesis, ways to use the information provided by the LSA method for sentence selection is explored.

3. The LSA is assumed to be language-independent. In this thesis, this assumption is explored further, and evaluations in different languages (Turkish and English) are performed.

In this thesis, we present a generic extractive text summarization system based on LSA which aims to extract summaries from single-documents. Two new summarization approaches based on LSA are proposed. The known and proposed LSA based summarization algorithms are applied on Turkish and English document sets.

1.3 Thesis Outline

The rest of the paper is organized as follows:

Chapter 2 presents the related work in document summarization. Categorization of text summarization systems, text summarization approaches in literature, and evaluation measures of the text summarization systems are explained briefly.

Chapter 3 explains the LSA approach in detail. Steps of the LSA for summarization

are given in this chapter. Different weighting schemas used in input matrix creation step is also explained.

Chapter 4 explains the existing algorithms that use different LSA approaches ((Gong and Liu 2001); (Steinberger and Jezek 2004); (Murray, Renals and Carletta 2005)). Also in this chapter, two newly proposed algorithms, Cross and Topic, are introduced and detailed information related to these algorithms is given.

Chapter 5 presents detailed information on ROUGE evaluation system. Also in this chapter, the evaluation results of the LSA based summarization algorithms using Turkish and English document sets are given and the results are discussed.

Chapter 6 presents the concluding remarks.

CHAPTER 2

RELATED WORK

Text summarization is an active research area of natural language processing. Summarization systems are basically composed of three main steps; interpretation, transformation and generation, as stated in (Jones 1999). These steps are affected from different aspects of the text summarization. Text summarization systems can be categorized according to how and why they are created, and what kind of approaches are used for creation of the summaries. The first studies about document summarization started in late fifties. Since then different methods are proposed to create better summaries. Evaluation of the summaries are done using different approaches such as using human evaluators, using precision/recall values, or using ROUGE scores.

In this chapter, detailed information related to the phases and factors of summarization systems, the categorization of text summarization systems, different summarization methods used in literature and evaluation approaches of text summarization systems are given.

2.1 Phases and Factors of Text Summarization Systems

Summary is the reduction of original text through selection and generalization of the important concepts, as stated in (Jones 1999) where the steps of summarization are given as interpretation, transformation and generation.

- In the interpretation step, input document is represented in a structured way

that the computation can be performed on it.

- In transformation step, input representation is converted into summary representation.

- In the last step, generation step, summary representation is converted into summary text.

These phases are affected from different factors of text summarization such as input, purpose, and output factors (Jones 1999).

Input factors: The features of input document can affect the resulting summary according to the following aspects:

- Structure: Structure is the organization of the given document with headers, chapters, sections, and etc. Structure of a document can be informative while creating summary.

- Scale: Scale is the length of the given document; such that document can be a long research paper, or a short news text. While long documents contain more topics and weaker co-relations, short documents contain more repeated terms about less number of topics.

- Language: Natural language used in the input document can affect the resulting summary. Summarization algorithms may or may not use language dependent information.

- Domain: The input document can be related to a specific topic, or can be more general. The summary created related to a specific topic may use world knowledge related to that topic.

- Unit: The number of documents to be used to create the document can be dif-

ferent. If a single document is used to create a single summary, the summarization systems are named single-document summarization systems. If more than one document related to a single subject is used to create a single summary, those summarization systems are named multi-document summarization system.

Purpose factors: Automatic summarization systems can create general summaries of a given text, or it can create summaries for a pre-defined task. The following aspects are related to the purpose factors of summarization systems.

- Situation: Situation is related to the context of the summary. The environment where the summary will be used; such as who uses the summary when and why; may or may not be known.

- Audience: Audience is related to the reader of the summary. If the interest of readers can be known, summaries can be created related to that subject. For example if the audience is a specific community in science, then it can be assumed that more specific information related to a single subject will be given in the summary.

- Use: Use is related to the aim for creating the summary. Summaries can be used for retrieving the source, previewing the input document, refreshing the memory about the input document which is read before and etc.

Output factors: The resulting summary can be affected from the following output aspects:

- Material: The summary of a document can be related to the all concepts mentioned in the text, or it may be related to some chosen concepts. Usually, general summarization systems intend to capture all concepts of the text. In user-focused summarization systems, like query-based summarization systems, the

summary may contain concepts related to the need of the user.

- Format: The created summary can be organized into fields, by using headings etc., or it can be organized as an unstructured text, like an abstract in a journal paper.

- Style: A summary can be informative, indicative, aggregative, or critical. Informative summaries give information about the concepts mentioned in the input document. Indicative summaries indicate what the input document is about. Aggregative summaries give supplementary information that does not exist in the input document. Critical summaries review rights and wrongs of the input document.

2.2 Categorization of Text Summarization Systems

The different aspects explained in Section 2.1 affect the resulting summary of document summarization systems. The resulting summary can have different forms, it can be produced by using different number of input documents, it can give information about whole document or it can be just an answer to a question.

Summaries can have different forms (Hahn and Mani 2000). Extractive summarization systems extract important text units; such as sentences, paragraphs, etc. from the input document. In (Das and Martins 2007), it is explained that the first concern of the extractive summarization systems is the content of the summary. Abstractive summarization approach is similar to the way that human summarizers follow, where the main concepts of a document are understood first, and then new sentences which are not seen in the original document are generated. In the study (Das and Martins 2007), it is explained that the first concern of the abstractive summarization systems is the form of the summary. Since abstractive summarization approach is more complex than the extractive summarization, most of automatic text summarization systems are extractive.

Another categorization of summarization systems are based on using single or multiple documents (Hovy and Lin 1999). While in single document summarization system a single-document is used for generating the summary, in multi-document summarization systems multiple documents on the same subject are used for the generation of a single summary.

Summarization systems can also be categorized as generic and query-based summarization systems. In generic summarization systems main topics are used to create the summary. In query-based summarization systems topics that are related to the answer of a question are used for the construction of the summary.

Another approach for categorizing document summarization systems is based on the technique that is used; namely supervised and unsupervised techniques as mentioned in the paper (Patil and Brazdil 2007). Supervised techniques use data sets that are labeled by human annotators, which is very expensive. Unsupervised approaches do not use annotated data, but they use linguistic and statistical information that are obtained from the document itself.

There are other categorization systems for document summarization (Jezek and Steinberger 2008), such as level of processing (surface-level or deeper-level), purpose of the summary (being informative, critical etc.), genre of the input documents (scientific articles, news texts etc.) and etc.

2.3 Text Summarization Approaches in Literature

There exist lots of different text summarization approaches in literature. Most of them are based on extraction of important sentences from the input text. Recently, other approaches are also proposed.

2.3.1 Extractive Summarization Methods

Extractive summarization methods try to find out the most important topics of an input document and select sentences that are related to these chosen concepts to

create the summary. In literature, there are different approaches based on surface level information, statistics, knowledge bases (ontologies, dictionaries), and so on.

2.3.1.1 Surface Level Approaches

The first study on summarization, which was conducted by (Luhn 1958), was based on frequency of the words in a document. The idea was that more frequent words are the ones that are most important. The sentences that contain these frequent words are assumed to be more important than other sentences, and are chosen to be a part of the resulting summary. After the study of the (Luhn 1958), other approaches that are based on simple, surface level features like terms from keywords/key phrases, terms from user queries and position of words/sentences are proposed.

The words/phrases that indicate importance in the given document can be used for selection of the sentences. For example, the word "significantly," or the phrase "in conclusion" can be used at the time of sentence selection for the summary. The studies (Teufel and Moens 1997) and (Edmundson 1969) are examples for the summarization systems that use keywords/key phrases as a part of their summarization systems.

Position of words/sentences can give information about the importance of that word/sentence. For example; it is observed that most of the time writers tend to write the most important content in the first sentence of the document. However this situation may change depending on the genre of the input document. It is observed that collecting only first few sentences of the given document can create successful summaries. The algorithms belonging to (Baxendale 1958), (Edmundson 1969) and (Brandow, Mitze and Rau 1995) are examples to the approaches that use position of words/sentences.

2.3.1.2 Statistical Approaches

Using statistical methods are another approach used for summarization. The most well known summarization approaches that use statistics are based on concept relevance and Bayesian classifier.

SUMMARIST project (Hovy and Lin 1999) is a well known text summarization project that uses statistical approach. In this project concept relevance information extracted from dictionaries and WordNet is used together with natural language processing methods. In this approach, a word is assumed to be occurred when other, related words are seen as well. For example, the number of occurrence of the word "automobile" is incremented when the word "car" is seen.

Another summarization application based on statistics belongs to (Kupiec, Pedersen and Chen 1995), in which Bayesian classifier is used for sentence extraction. In this approach a corpus of full texts and summaries are used for training of the system. The features used in this system are word frequency, uppercase words, sentence length, position in paragraph, and phrase structure.

2.3.1.3 Text Connectivity Based Approaches

Text connectivity is another approach in document summarization. It deals with problems of referencing to the already mentioned parts of a document. Methods that use lexical chains and Rhetorical Structure Theory are examples of the summarization systems that use text connectivity.

Lexical chains method is a well known algorithm that uses text connectivity. In this approach, semantic relations of words (synonymy, antonymy etc.) are extracted using dictionaries and WordNet. Using semantic relations lexical chains are constructed and used for extracting important sentences in a document. The algorithms belonging to (Barzilay and Elhadad 1997) and (Ercan and Çiçekli 2008) are example methods that use lexical chains for summarization.

Rhetorical Structure Theory (RST) based methods are another example that uses text connectivity for summarization. RST organizes text units into a tree like structure. Then this structure is used for summarization purposes. (Ono, Sumita and Miike 1994) and (Marcu 1997) use RST for summarization.

2.3.1.4 Graph Based Approaches

Graph based summarization approaches are another approach for text summarization. As stated in (Jezek and Steinberger 2008), the well known graph based algorithms HITS (Kleinberg 1999) and Google's PageRank (Brin and Page 1998) were developed to understand the structure of the Web. These methods are then used in text summarization.

The nodes in graph based summarization approaches represent the sentences, and the edges represent the similarity among the sentences. The similarity values are calculated using the overlapping words or phrases. The sentences with highest similarity to the other sentences are chosen as a part of the resulting summary. TextRank (Mihalcea and Tarau 2004) and Cluster LexRank (Qazvinian and Radev 2008) are two methods that use graph based approach for document summarization.

2.3.1.5 Machine Learning Based Approaches

Machine learning based approaches are also used for text summarization with the help of advances in machine learning and natural language processing. First approaches used the assumption that the features are independent. Then other approaches using dependence assumption are developed. The machine learning based summarization algorithms use techniques like Naïve-Bayes, Decision Trees, Hidden Markov Model, Log-linear Models, and Neural Networks. As stated in the paper of (Das and Martins 2007) some example studies related to machine learning based approaches belong to (Kupiec, Pedersen and Chen 1995), (Aone, et al. 1999), (Lin and Hovy 1997), (Conroy and O'leary 2001), (Osborne 2002) and (Svore, Vanderwende and Burges 2007).

2.3.1.6 Algebraic Approaches

In recent years, algebraic methods such as Latent Semantic Analysis (LSA) (Landauer, Foltz and Laham 1998), Non-negative Matrix Factorization (NMF) (Lee and Seung 1999), and Semi-discrete Matrix Decomposition (SDD) (Kolda and O'Leary 1998) are used for document summarization. Among these algorithms most well-known one is LSA, which is based on singular value decomposition (SVD). In this algorithm similarity among sentences and similarity among words are extracted. Other than summarization, LSA algorithm is also used for document clustering and information filtering.

2.3.2 Non-Extractive Summarization Methods

Abstractive summarization methods try to fully understand the given documents, even non-explicitly mentioned topics, and generate new sentences for the summary. This approach is very similar to the way of human summarization. But practically, achieving the performance of a human summary is hard. In literature, there are approaches that create summaries in a non-extractive manner, using information extraction, ontological information, information fusion, and compression (Radev, Hovy and McKeown 2002).

As stated in the study (Radev, Hovy and McKeown 2002), predefined information types are given and summarizers find out related information to the given set. The approaches of (DeJong 1978) and (Rau and Jacobs 1991) are examples to information extraction based summarization approaches. In the same study, it is stated that in compression based summarization systems, after choosing important words or phrases, sentences for the summary are generated. The summarization approach of (Witbrock and Mittal 1999) is based on compression. There are also reduction based summarization systems which combines two or more sentences into one (Radev, Hovy and McKeown 2002). The summarization approach belonging to (Knight and Marcu 2000) is based on reduction.

2.4 Evaluation Measures

Evaluation measures are categorized in sub-categories in the paper of (Radev, Teufel, et al. 2003) and in the PhD thesis of (Steinberger 2007), which can be seen in Figure 1. Text quality based evaluation is done by human annotators who give score to each summary according to a predefined scale. Content based evaluation is done against a grand-truth summary, which is created by a human. Content based evaluations can use information of matching sentences (co-selection based evaluation) or matching words (content based evaluation). Task based evaluations measure the quality of the summary for a given task, e.g. question answering.

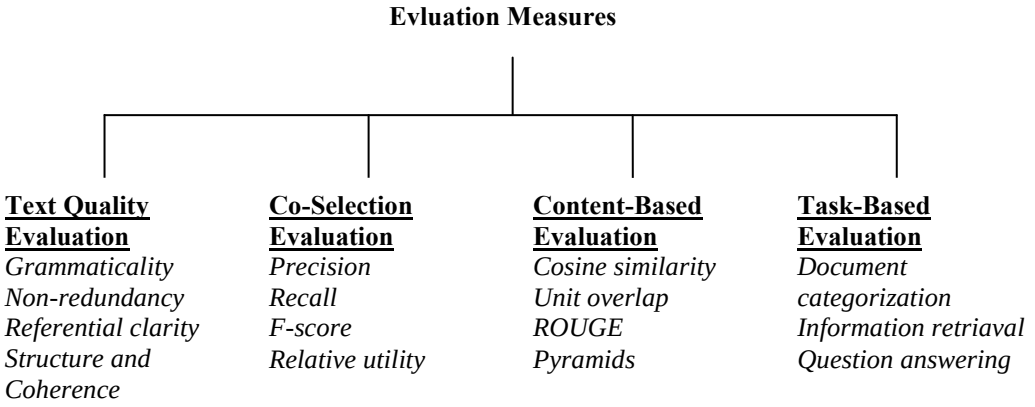

Figure 1 - Categorization of evaluation measures.

2.4.1 Text Quality Evaluation

The first category of evaluation measures is based on text quality using the aspects like grammaticality, non-redundancy, referential clarity and coherence. The text should not contain any grammatical error such as incorrect words or punctuation errors. Also the created summary should not contain redundant information and

the references in it should be clearly matched with the known objects. Good structure and the coherence are also important issues for the summary.

Text quality based evaluation is done by human evaluators. But, as explained in (Das and Martins 2007), human markings are unstable, and also it is known that using human evaluators is a time consuming evaluation method.

2.4.2 Co-Selection Evaluation

The second category is based on co-selection, where extracted summaries are compared with ideal summaries. The comparison of the summaries is done based on the selected sentences. Co-selection based evaluation may use precision, recall, f-measure values or may use relative utility.

Precision is defined in the glossary of (Baeza-Yates and Ribeiro-Neto 1999) as "an information retrieval performance measure that quantifies the fraction of retrieved documents which are known to be relevant." For text summarization, it is the division of extracted summary sentences and ideal summary sentences intersection over whole extracted summary sentences.

$$Precision = \frac{RelevantSentences \cap RetrievedSentences}{RetrievedSentences} \quad (1)$$

Recall is defined as "an information retrieval performance measure that quantifies the fraction of known relevant documents which were effectively retrieved", in the glossary of (Baeza-Yates and Ribeiro-Neto 1999). From the point of view of document summarization, it is the division of extracted summary sentences and ideal summary sentences intersection over the ideal summary sentences.

$$Recall = \frac{RelevantSentences \cap RetrievedSentences}{RelevantSentences} \quad (2)$$

F-score (F-measure) is a statistical measure that combines both precision and recall. Traditionally it is defined as the harmonic mean of precision and recall. F-score values changes in the interval of 0 and 1, where best result is 1.

$$F - score = \frac{2 * Precision * Recall}{Precision + Recall} \quad (3)$$

In literature, it is also common to use different weights for precision and recall, while calculating F-score. The weight value β is a non-negative real value. While it is set to a value larger than 1 to indicate that precision is more important, it is set to a value less than 1 to indicate that recall is more important.

$$F - score = \frac{(\beta^2 + 1) * (Precision * Recall)}{\beta^2 * Precision * Recall} \quad (4)$$

Precision and recall values may not be appropriate in some cases in text summarization. For example, from a document with five sentences [1, 2, 3, 4, 5], two different summaries are created. The first summary contains the sentences [1, 2, 5] and the other one contains the sentences [1, 4, 5]. The ideal summary contains [1, 2, 5]. While using precision and recall based evaluation, one can decide that the first summary is better than the second. But the process of summarization is also subjective, and the second summary can also be as good as the first one.

Relative utility measure is introduced by (Radev, Jing and Budzikowska 2000) to overcome the problem of the precision and recall based evaluation, as stated in the thesis of (Steinberger 2007). In this measure, ideal summary is represented

with the original sentences and their utility values. The decision of the utility values is made by human judges and is used for giving information about how important a sentence is in the given document. For example, an ideal summary for a five sentence document is given as [1/5, 2/3, 3/2, 4/3, 5/4]. The utility values for this example indicates that the first sentence is the most important sentence, the third sentence is the least important sentence, and the significance of the second and the fourth sentences are equal. So, when two different summaries collecting sentences [1, 2, 5] and [1, 4, 5] are in fact has the same evaluation score. Also, both have the highest scores that can be obtained, which indicates that both summaries are optimal.

Co-selection based evaluation can only be used with extractive summaries, where the summary is constructed using the original sentences from the input document. But in summaries, it is possible that the sentences are formatted in a different manner, by combining original sentences and/or using synonyms etc., while keeping same meaning as the original sentences.

2.4.3 Content-Based Evaluation

The third category is based on the content. In this approach, also extracted summaries are compared with ideal summaries, but this time the comparison is done using the words. Using this approach, it is possible to compare extracted and ideal summaries, even they do not share sentence. For content-based evaluations, measures such as cosine similarity, longest common subsequence, pyramids, and ROUGE scores are used.

Cosine similarity finds out the similarity of two vectors by using dot product and magnitude. In the Formula (5), A and B are vectors of attributes and θ is the cosine similarity. From the point of view of text summarization, A and B are the extracted summary and the ideal summary, respectively.

$$Similarity = \cos(\theta) = \frac{A.B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \, x \, B_i}{\sum_{i=1}^{n}(A_i)^2 \, x \sum_{i=1}^{n}(B_i)^2} \quad (5)$$

Longest Common Subsequence (LCS) based evaluation is introduced by (Radev, Teufel, et al. 2003), as stated in the PhD thesis of (Steinberger 2007). In the Formula (6), LCS finds out the length of the longest common subsequence between X and Y, which are represented as sequence of words. *length(X)* is the length of the string X and the value $edit_{di}(X, Y)$ is the edit distance between X and Y.

$$lcs(X,Y) = \frac{length(X) + length(Y) - edit_{di}(X,Y)}{2} \quad (6)$$

ROUGE N-gram co-occurrence measure is another content based evaluation method. Given multiple human judged, ideal summaries, maximum number of n-gram co-occurring between extracted and ideal summaries is calculated. The value is then divided by the total number of n-grams in ideal summaries. ROUGE-n score is a re-call based score. In the Formula (7), ROUGE-n calculation is given, where *RSS* is the reference summary set and *C* is the candidate summary and *n* is the length of the n-gram.

$$ROUGE - n = \frac{\sum_{C \epsilon RSS} \sum_{gram_n \epsilon C} Count_{match}(gram_n)}{\sum_{C \epsilon RSS} \sum_{gram_n \epsilon C} Count(gram_n)} \quad (7)$$

There are also different ROUGE scores, which are based on longest common subsequence (ROUGE-L), bigram measure that enables at most 4 unigrams inside a bi-gram component (ROUGE-SU4) and etc.

Pyramids method (Nenkova and Passonneau 2004) is a semi-automatic evaluation method which identifies summarization content units (SCUs). As explained in the PhD thesis of (Steinberger 2007), SCUs are extracted from multiple manual summaries. First, annotators identify similar sentences. Then more common SCUs are placed in higher parts of the pyramid and are given higher importance score. The extracted summaries are then compared against the ideal summaries, using the given importance scores. The need for annotation is a drawback of this evaluation method.

Content based evaluation is better than the co-selection based evaluation methods, since this approach can match two different sentences that have the same information but have different structures.

2.4.4 Task-Based Evaluation

The last category is task-based evaluation. In this evaluation approach, summaries that are created for a purpose are compared according to their performance of accomplishing the given task. Task based evaluation can use different approaches in order to evaluate the performance of the summarization system. Some of these approaches are information retrieval, question answering, and document clustering methods.

The performance of a summarization system can be understood using information retrieval approaches. The performance of the information retrieval approach using the full document and the performance of that approach using the extracted summary is compared. If the performance of the information retrieval approach does not change much, it is decided that the summarization system is successful. For more detail the paper belonging (Radev, Teufel, et al. 2003) can be used.

Similar to information retrieval approach, question answering approaches can be used for summarization evaluation. This time, reading only the input text or only the summary text, human judges replies some multiple choice questions. The correct result ratios are used to evaluate the summarization system. For more detail the paper

(Morris, Kasper and Adams 1992) can be used.

Document categorization is also used for summarization evaluation. For this purpose document corpuses with category labels are used. Categorization by human judges or automatic classifiers is performed using original document, the extracted summaries and randomly created summaries. While the results with original documents set the upper bound, the summaries created by choosing random sentences set the lower bound. Using precision and recall values, the extracted summaries can be compared with the results of using original documents or randomly created summaries.

CHAPTER 3

LATENT SEMANTIC ANALYSIS

Latent Semantic Analysis (LSA) is an algebraic-statistical method which extracts hidden semantic structures of words and sentences. It is an unsupervised approach, which does not need any training or external knowledge. LSA uses context of the input document and extracts information such as which words are used together and which common words are seen in different sentences. If the number of common words between sentences is high, it means that the sentences are more semantically related.

LSA method has the ability to represent the meaning of words, and meaning of sentences at the same time. Meaning of a sentence is decided using the word it contains, and meaning of words are decided using the sentences that contains the word. For finding out the interrelations between sentences and words, an algebraic method named Singular Value Decomposition (SVD) is used. Besides having capability of modeling relationships among words and sentences, SVD has the capability of noise reduction, which helps to improve accuracy.

In order to see how LSA can represent the meaning of words and sentences an example is given below.

Example:

Three sentences are given as an input to LSA:

d0: "The man walked the dog"

d1: "The man took the dog to the park"

d2: "The dog went to the park"

After performing the calculations we get the resulting graphic:

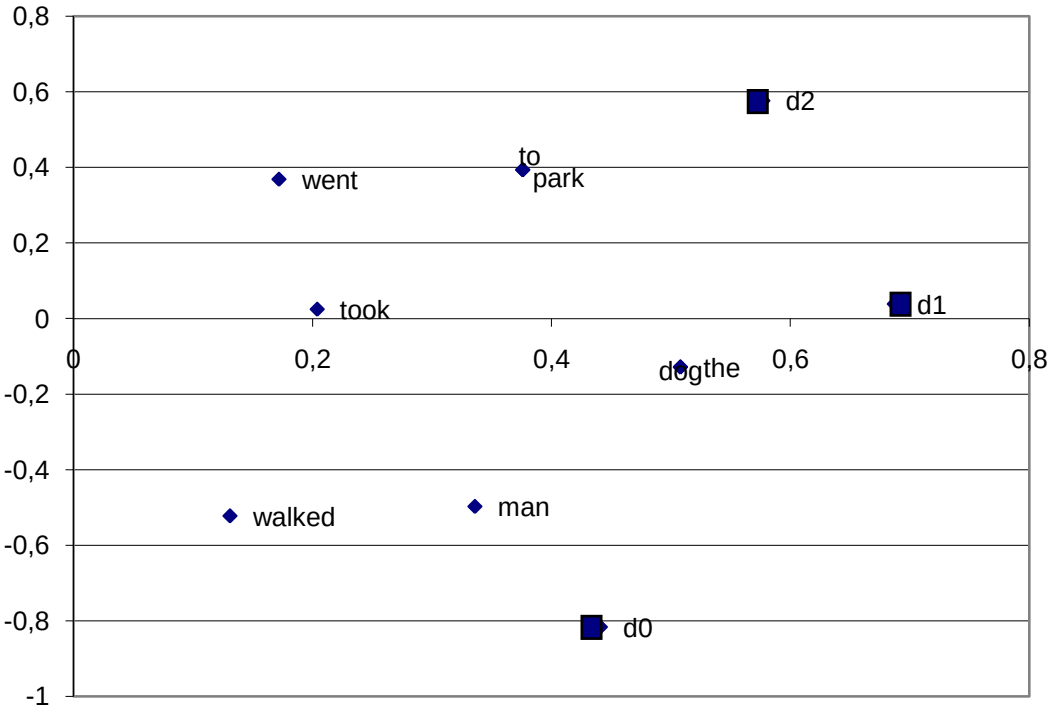

Figure 2 - LSA can represent the meaning of words and sentences

From the Figure 2, we can see that d1 is more related to d2 than d0; and the word "walked" is related to the word "man"; but not so much related to the word "park". These kinds of analysis can be made by using LSA and input data, without any external knowledge.

The summarization algorithms that are based on LSA method usually contain three main steps:

1. *Input Matrix Creation:* The input document is represented in a matrix form to perform the calculations.

2. *Singular Value Decomposition (SVD):* SVD is an algebraic method that can model relationships among words/phrases and sentences.

3. *Sentence Selection:* Using the results of SVD different algorithms use different approaches to select important sentences.

LSA has several limitations: The first is that it does not use information of word order, syntactic relations, and morphologies. These kinds of information can be necessary for finding out the meaning of the words and the meaning of the texts. The second limitation is that it uses no world knowledge, but just the information that exists in input document. The third limitation is related to the performance of the algorithm. With larger and more inhomogeneous data the performance decreases sharply. The decrease in performance is related to the usage of SVD, which is a very complex algorithm.

3.1 Input Matrix Creation

An input document needs to be represented in such a way that a computer can understand and perform calculations on. This representation is usually a matrix representation, where columns are sentences and rows are words/phrases. The cells are used to represent the importance of words in sentences. For filling out the cell values, different approaches can be used. Since all words are not seen in all sentences, most of the time the created matrix is sparse.

3.1.1 Creation of the Matrix

The first step of input matrix creation is to create the matrix in the form of *terms* x *sentences*. Assuming there are *m* terms and *n* sentences, the matrix A with size of *m* x *n* is created, which is $A = [A_1, A_2, A_n]$. Each column A_i represents weighted term vector of sentence *i* of the input document. The terms can be words/phrases that have been seen in the sentences, or they can be preprocessed before the creation of the matrix.

The effect of the way of input matrix creation is high for summarization, since it affects the resulting matrices calculated with Singular Value Decomposition (SVD), the second step of LSA. SVD is a complex algorithm and increase in size of input matrix degrades the performance. In order to reduce matrix size, the rows of the matrix- the words- can be reduced by preprocessing approaches like stop word removal, using roots of words only, using phrases instead of words and etc. These preprocessing approaches are mostly language dependent.

3.1.1.1 Segmentation

The smallest unit that will be extracted from the original document for the summary should be decided before performing summarization. The smallest unit can be a paragraph, a sentence, or a phrase. Most common extractive summarization approach is sentence level extraction.

In order to find out the boundaries of the phrases/sentences, the input text is segmented into tokens (words). This step is not a trivial one, since there are irregularities in natural languages. For most of the languages white spaces and punctuation marks are used as boundary markers. As stated in the PhD thesis of (Hassel 2007), segmentation is much more difficult task for languages without word-boundary markers, such as Chinese and Japanese, but there are more works based on statistics to solve the problem, e.g. Chinese word segmentation (Luk 1994).

3.1.1.2 Stemming

In documents a word can be seen in different formats, such as plural vs. singular, present vs. past tense, etc. Most of the time these words have the same meaning and treating them differently is unnecessary. In order to use these words as the same token (concept), stemmers are used.

Stemmers are tools that reduce the original word forms into roots (stems) of these words. Stemmers are necessary to represent different word forms in a single format and to reduce memory usage for storing the words. Also, smaller list of words make it easier to perform calculations. As a result of performing stemming, document representation (input matrix) is less noisy and more dense.

The efficiency of a stemmer is important while performing further calculations. Sometimes stemmers can do over-stemming such that two words are given the same stem, while it should not be. For example, the words “experience” and “experiment” are two different words, which should not be stemmed into the same root. But stemmers can find out their root as “experi”. Another stemming problem is related to under-stemming such that two words should have been stemmed into the same word, but have not been. For example, “run” and “ran” can be found as two different stems, instead of one.

In this thesis, for Turkish documents, Zemberek Morphological Analyzer (Zemberek 1999), and for English documents, Porter Stemmer (Porter Stemmer 2000) are used for performing stemming.

3.1.1.3 Stopword Filtering

Input documents usually contain words that do not add information but are necessary for syntactical formation, such as words like “the”, “is”, etc. Since these words are less usefull and less informative, they introduce noise into the document representation (input matrix). In order to get rid of these kind of words, a stopword

removal step is used.

Stopword removal is done using predefined, human-made list of words. The words in the list are not used while creating the input matrix. Since a predefined list is used, this approach is language dependent. Instead of using these kinds of lists, a frequency threshold can be used. If a word is seen more/less frequently than predefined threshold, that word can be considered as stopword. But decision of threshold is another issue to be considered.

In this thesis predefined lists of stop words in Turkish (Stop Words List-Turkish 2010) and in English (Stop Words List-English 2010) are used.

3.1.2 Weighting Cell Values

The cell values of matrix can change the results of SVD. The cell values represent the importance of words in sentences. There are different approaches to fill out the cell values. These approaches are as follows:

1. *Frequency of word:* The cell is filled out with the frequency of the word in the sentence.

2. *Binary Representation:* The cell is filled out with 0/1 according to the existence of word in the sentence.

3. *Tf-Idf (Term Frequency–Inverse Document Frequency):* The cell is filled out with tf-idf value of the word. When the word is more frequent in the sentence but less frequent in the whole document the tf-idf value is higher. The higher tf-idf value indicates that the word is much more representative for that sentence than others. Tf-idf is calculated as:

$$Tf - idf = tf - idf \tag{8}$$

The term frequency *(tf)* value is calculated using the Formula (9). The *n (i, j)* is the number of occurrences of the considered word *i* in sentence *j*, and $\sum_k n$ *(k, j)* is the sum of number of occurrences of all words in sentence *j*.

$$tf(i,j) = \frac{n(i,j)}{\sum_{k \epsilon AllWordsInj} n(k,j)} \quad (9)$$

The inverse document frequency *(idf)* value is calculated using the Formula (10). In this formula the $|D|$ is the total number of sentences in the input text, and d_i is the number of sentences where the word *i* appears

$$idf = \log \frac{|D|}{d_i} \quad (10)$$

4. *Log Entropy*: The cell is filled with log-entropy value of the word, which gives information on how informative the word is in the sentence. It is computed as follows:

$$p(i,j) = \frac{f_{ij}}{gf_i} \quad (11)$$

$$sum = \sum_j p(i,j) \log_2 p(i,j) \quad (12)$$

$$global(i) = 1 + \left(\frac{sum}{\log_2 n}\right) \quad (13)$$

$$local(i,j) = \log_2(1 + f(i,j)) \quad (14)$$

$$log - entropy = gloabl * local \quad (15)$$

In the formulas from the Formula (11) to Formula (15), the *p (i, j)* is the probability of word *i* that is appeared in sentence *j*, gf_i is the total number of times word *i* occurs in all sentences, *f (i, j)* is the number of times word *i* appeared in sentence *j*, and *n* is the number of sentences in the document.

5. *Root Type:* The cell is filled with frequency of the word if its root type is noun; otherwise the cell value is set to 0.

6. *Modified Tf-Idf:* This approach proposed in order to eliminate noise from the input matrix. The cell values are set to tf-idf scores first, and then the words that have scores less than or equal to the average of the row is set to 0.

3.2 Singular Value Decomposition

Singular Value Decomposition (SVD) is an algebraic method that can model relationships among words/phrases and sentences. In this method, the given input matrix A is decomposed into three new matrices as follows (Figure 3):

$$A = U\Sigma V^T \quad (16)$$

A: Input matrix (m x n)
U: Words x Extracted Concepts (m x n)
∑: Scaling values, diagonal descending matrix (n x n)
V: Sentences x Extracted Concepts (n x n)

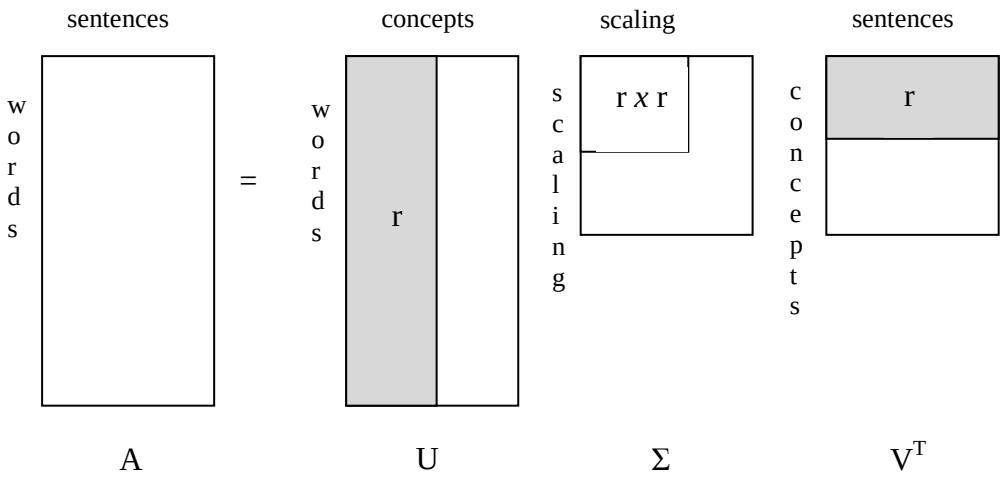

Figure 3 - Singular Value Decomposition

SVD has the capability of mapping *m*-dimensional term vector space into *r*-dimensional singular vector space. The mapping reveals latent semantic structure of the input document. *r* linearly independent vectors represent the topics (concepts) of the input document. *r* value is the rank and is equal to or less than the value of *min (m, n).*

Decision of the singular vectors is based on co-occurrence of the words. If the words are co-occurred in different parts of the document, they are considered as related to each other. The magnitude of singular vectors gives information about the importance of the concept. The sentences related to the concepts, by containing the related words, are projected along the singular vectors. The sentence with the highest index value is the most representative sentence about that singular vector (concept).

The reduction in dimension, *r* value, is important; since it can affect the later computation performance. As stated in (Deerwester, et al. 1990), the value of r should be

able to fit the real structure of the input data, while not having noisy data, such as containing unimportant information. The proper way of deciding the *r*-value is still an open question in literature.

The well-known drawback of SVD is that it is time consuming. Once new terms/sentences are added to the initial matrix, SVD calculation has to be performed again. To overcome this problem (Deerwester, et al. 1990) suggests locating new term/sentence at the centroid of the terms/sentences, respectively. Another drawback of SVD is related to polysemy. After the SVD calculations, every word is represented as a point in space. If a word has multiple entirely different meanings, that word represented as the average of the different meanings in the singular vectors space. This may cause problems in the later steps of the document analysis. (Deerwester, et al. 1990) suggests detecting the different meanings of the words and categorizing them in different places in the space.

3.3 Sentence Selection

Using the results of SVD, different algorithms use different approaches to select sentences. The algorithms aim to find out which sentences are more representative of the input document than other sentences. The details of these algorithms are described in Chapter 4 of this thesis.

CHAPTER 4

TEXT SUMMARIZATION USING LSA

The algorithms in the literature that use LSA for text summarization, perform the three steps explained in Chapter 3; namely input matrix creation, singular value decomposition, and sentence selection. In the sentence selection step, algorithms follow different approaches which will be detailed in Section 4.1. Also, new methods that are proposed in this thesis will be explained in Section 4.2.

4.1 Sentence Selection Approaches in Literature

There are various algorithms that use LSA for document summarization. In this paper three of them will be explained in detail.

4.1.1 Gong and Liu (2001)

The algorithm of Gong and Liu is one of the main studies conducted in LSA based text summarization. After representing the input document in matrix and doing calculations of SVD, V^T matrix, matrix of *extracted concepts x sentences,* is used for selecting the important sentences. In V^T matrix, row order indicates the importance of the concepts such that the first row represents the most important concept extracted. The cell values of this matrix show the relation between the sentence and the concept. A higher cell value indicates that the sentence is more related to the concept.

In the approach of Gong and Liu, one sentence is chosen from the most important concept, and then second sentence is chosen from the second most important concept; and this process continues until all predefined number of sentences are collected. The number of sentences to be collected is given as a parameter.

In the Example given in Chapter 3, three sentences were given, and the SVD calculations were done accordingly. The resulting V^T matrix, with its rank set to two, is as in Figure 4.

V^T matrix (r = 2)			
	Sent0	Sent1	Sent2
Con0	0,457	0,728	0,510
Con1	-0,770	0,037	0,637

Figure 4 - V^T matrix. From each row, sentence with highest score is chosen until predefined number of sentences are collected.

In Figure 4, the example V^T matrix which is calculated based on the example given in Chapter 3 is given. First, the concept *con0* is chosen, and then the sentence *sent1* is chosen, since it has the highest cell value in that row.

The approach of Gong and Liu has some disadvantages that are defined by (Steinberger and Jezek 2004). The first disadvantage is that the number of sentences to be collected is the same as the reduced dimension. If the given predefined number is large, sentences from less significant concepts are chosen. The second disadvantage is related to choosing only one sentence from each concept. Some concepts, especially important ones, can contain sentences that are highly related to the concept, but do not have the highest cell value. The last disadvantage is that all chosen

concepts are assumed to be in the same importance level, which may not be true.

4.1.2 Steinberger and Jezek (2004)

The approach of Steinberger and Jezek starts with input matrix creation and SVD calculation. After these steps, sentence selection step is applied which differs from the approach of Gong and Liu. The approach of Steinberger and Jezek uses both V and $\sum$ matrixes for sentence selection.

In this approach, length of each sentence vector, represented by the row of V matrix, is used for sentence selection. The calculation of the length of the sentence *i* is calculated as follow:

$$Length = \sqrt{\sum_{j=1}^{n} V_{ij} * \Sigma_{jj}} \qquad (17)$$

The dimension of new space, *n*, is given as a parameter to the approach of Steinberger and Jezek. The concepts whose indexes less than or equal to the given dimension are used for length calculations. $\sum$ matrix is used as a multiplication parameter in order to give more emphasis on the most important concepts. The sentence with the highest length value is chosen to be a part of the resulting summary.

Using the results of the example given in Chapter 3, calculated length values are given in Figure 5. The given dimension size is two for this example. Since the sentence *sent1* has the highest length, it is extracted first as a part of the summary.

Length scores	
Sent0	1,043
Sent1	1,929
Sent2	1,889

Figure 5 - Length scores. Sentence with highest length score is chosen.

The main purpose of this algorithm is to create a better summary, by getting rid of disadvantages of Gong and Liu summarization algorithm. In Steinberger and Jezek approach sentences that are related to all important concepts are chosen, while allowing collection of more than one sentence from an important concept.

4.1.3 Murray, Renals and Carletta (2005)

The first two steps of the LSA algorithm are executed before sentence selection step, as in the previous algorithms. In this approach, V^T and $\sum$ matrices are used for sentence selection.

In this approach more than one sentence can be collected from the topmost important concepts, placed in the first rows of the V^T matrix. Decision of how many sentences will be collected from each concept is made by using $\sum$ matrix. The value is decided by getting percentage of the related singular value over the sum of all singular values, for each concept.

V^T matrix (r = 2)			
	Sent0	Sent1	Sent2
Con0	0,457	0,728	0,510
Con1	-0,770	0,037	0,637

Figure 6 - V^T matrix. From each row, sentences with higher scores are chosen until calculated number of sentences are collected.

In Figure 6, the V^T matrix of the Example given in Chapter 3 is given. From the calculations of $\sum$ matrix, it is observed that collecting one sentence from the first row is sufficient, but for demonstration purposes two sentences will be collected from the Figure 6. So, from *con0* the sentences *sent1* and *sent2* are selected as a part of the summary.

The approach of Murray, Renals and Carletta solves the problems of Gong & Liu's approach of selecting single sentence from each concept, even the concept is very important. In this approach, more than one sentence can be chosen even they do not have the highest cell value in the row of the related concept. Also, the reduced dimension has not to be the same as the number of sentences in the resulting summary.

4.2 Proposed Sentence Selection Algorithms

The analysis of input documents released that some of the sentences, especially the ones in introduction and conclusion parts, belong to more than one concept at the same time. It is observed that these sentences may cause noisy information. In order to understand the effect of these sentences in LSA based summarization systems *Cross* method is proposed.

Another concern is related to the extracted concepts using SVD. These concepts can be subtopics of other topics, or can be main topics. In order to observe if extracted concepts are main topics or subtopics, *Topic* method is proposed. Using this method,

main topics are extracted and sentences are selected from main topics.

4.2.1 Cross Method

Cross method is an extension to the approach (Steinberger and Jezek 2004). In this approach input matrix creation and SVD calculation steps are executed as in other approaches and then the V^T matrix is used for sentence selection purposes. Between the SVD calculation step and the sentence selection step, there exists a pre-processing step.

The aim of pre-processing step is to remove overall effect of sentences that are related to the concept somehow, but not one of the most significant sentences for that concept. For each concept, which is represented by the rows of the V^T matrix, the average sentence score is calculated. Then the cell values which are less than or equal to the average score are set to zero.

After preprocessing, the steps of Steinberger and Jezek approach are followed with a modification. In our Cross approach, the total length of each sentence vector, which is represented by a column of the V^T matrix, is calculated. Then, the longest sentence vectors are collected as a part of the resulting summary.

V^T matrix (r = 2)				
	Sent0	Sent1	Sent2	Avg.
Con0	~~0,457~~	0,728	~~0,510~~	0,565
Con1	~~-0,770~~	0,037	0,637	-0,021

Figure 7 - V^T matrix after preprocessing.

In Figure 7, an example V^T matrix is given after the preprocessing is executed. For the preprocessing step; first the average score for each concept is calculated, and then the cell values less than this average is set to zero.

V^T matrix (r = 2)			
	Sent0	Sent1	Sent2
Con0	0	0,728	0
Con1	0	0,037	0,637
Length	0	0,765	0,637

Figure 8 - V^T matrix and length scores

In Figure 8, the length scores calculated by adding up the concept scores with values after the preprocessing step. In this example matrix, *sen1* has the highest length score, so it has been chosen to be part of the summary.

4.2.2 Topic Method

Topic method is similar to the other approaches in following the steps of summarization approaches based on LSA. In this step first the input documents are represented in a matrix form, and then SVD calculation is done. After these steps, a preprocessing step is followed before selecting the sentences for the summary. For preprocessing and the sentence selection steps the V^T matrix is used.

The main idea in topic method is to find out the main-concepts and sub-concepts. The resulting concepts extracted from the SVD calculations are known to be topics of the input document. But these topics can be sub-topics of other extracted topics. In this approach, after deciding the main topics which may be a group of subtopics, the

sentences are collected as a part of the summary from the main topics.

The preprocessing step of this approach starts with a similar way of Cross approach's pre-processing step. First, average sentence score is calculated for each concept using the row of V^T matrix. Then the cell values less than this score are set to zero. This step removes sentences that are not highly related to the concept, leaving only the most important sentences related to that concept. In Figure 9, an example V^T matrix after preprocessing is given.

V^T matrix (r = 2)				
	Sent0	Sent1	Sent2	Avg.
Con0	~~0,457~~	0,728	~~0,510~~	0,565
Con1	~~-0,770~~	0,037	0,637	-0,021

Figure 9 - V^T matrix after preprocessing

After the first step of preprocessing, the step of finding out main topics comes. For this step, a concept x concept matrix is created. This new matrix is created by finding out concepts that has common sentences, and setting the new cell values to the total of common sentence scores. In Figure 10, example *concept* x *concept* matrix based on the V^T matrix in Figure 9 is given.

	Con0	Con1
Con0	1,456	0,765
Con1	0,765	1,348

Figure 10 - New concept x concept matrix

After the creation of *concept x concept* matrix, the strength of each concept is calculated. For this calculation, each concept is thought to be a node and cell values are thought to be similarity values, the edge scores, among nodes. So, the newly created *concept x concept* matrix can be thought as a graph representation of the concepts of the input document.

	Strength
Con0	2,221
Con1	2,113

Figure 11 - Strength values

For each concept, the strength value is calculated by getting the cumulative of the cell values for each row of the *concept x concept* matrix. The concept with the highest strength value is chosen as the main topic of the input document. A higher strength value indicates that the concept is much more related to the other concepts, and it is one of the main topics of the input text. In Figure 11, calculated strength values can be seen. Since *con0* has the highest strength value, it is chosen to be the main topic.

V^T matrix (r = 2)			
	Sent0	Sent1	Sent2
Con0	0	0,728	0
Con1	0	0,037	0,637

Figure 12 - V^T matrix after preprocessing

After these steps, the sentences are collected from the pre-processed V^T matrix following the approach of Gong and Liu. As explained before, a single sentence is collected from each concept until predefined numbers of sentences are collected. In Topic method, instead of topmost concepts of V^T matrix, the chosen main concepts are used for sentence selection. In Figure 12, *sen1* is chosen from *con0*, since that sentence has the highest cell value.

CHAPTER 5

EVALUATION

Evaluation of the LSA based summarization approaches are conducted on Turkish and English datasets. The evaluations are based on ROUGE evaluation system.

In the Section 5.1, general information about ROUGE evaluation system is given. In the Section 5.2.1 and in the Section 5.4.2, information about the Turkish datasets and the English datasets are presented, respectively. In the same sections, the evaluation results for the datasets are given. In the Section 5.2.3, the results of LSA based summarization approaches are compared against other summarization approaches. Lastly in the Section 5.2.4 analysis of the evaluation results are discussed.

5.1 ROUGE Evaluation Approach

Recall-Oriented Understudy for Gisting Evaluation (ROUGE) is a set of metrics introduced in (Lin and Hovy 2003) and (Lin 2004).As stated in (Das and Martins 2007) it is used as a standard of automatic evaluation of document summarization. The ROUGE evaluation approach is based on n-gram co-occurrence, longest common subsequence and weighted longest common subsequence between the ideal summary and the extracted summary.

The n-gram based ROUGE score, ROUGE-N score, is based on comparing n-grams in the ideal summaries and the reference summary. This score is computed as in the Formula (18):

$$ROUGE - N(s) = \frac{\sum_{r \epsilon R} \langle \Phi_{\mathrm{n}}(r), \Phi_{\mathrm{n}}(\mathrm{s}) \rangle}{\sum_{r \epsilon R} \langle \Phi_{\mathrm{n}}(r), \Phi_{\mathrm{n}}(\mathrm{r}) \rangle} \quad (18)$$

In the Formula (18), $R = \{r_1, r_2.., r_n\}$ is the set of ideal (reference) summaries, *s* is the extracted summary. The Φ_n *(d)* value is the representation of the n-grams in document *d*, whose values are set to 1 if the n-gram in that index seen in the document *d* and is set to 0 otherwise. From the Formula (18), it is obvious that ROUGE-N is based on recall statistics.

The longest common subsequence (LCS) based ROUGE score, ROUGE-L score, is based on the idea that longer LCS value between the ideal and extracted summary sentences indicates that the sentences are more similar. The ROUGE-L score is calculated as in the Formula (21)(19):

$$R_{LCS}(s) = \frac{\sum_{i=1}^{u} LCS(r_i, s)}{\sum_{i=1}^{u} |r_i|} \quad (19)$$

$$P_{LCS}(s) = \frac{\sum_{i=1}^{u} LCS(r_i, s)}{|s|} \quad (20)$$

$$ROUGE - L(s) = \frac{(1 + \beta^2) R_{LCS} P_{LCS}}{R_{LCS} + \beta^2 P_{LCS}} \quad (21)$$

The |*x*| value, which is used in precision and recall calculations in the Formula (19) and the Formula (20), is the length of the sentence. The LCS(x, y) value is the length of the LCS between *x* and *y*. The β value is used for weighting the precision and recall scores. The ROUGE-L score is based on F-measure. Another ROUGE measure is based on using weights on Formula (21), to penalize non-consecutive subsequence

matches. This measure is named ROUGE-W.

Another ROUGE score, ROUGE-S, is based on ordered pairs of words that are common between the ideal and the extracted summaries. This score is calculated using the Formula (24):

$$R_s(s) = \frac{\sum_{i=1}^{u} \langle \Psi_2(r_i), \Psi_2(s) \rangle}{\sum_{i=1}^{u} \langle \Psi_2(r_i), \Psi_2(r_i) \rangle} \quad (22)$$

$$P_S(s) = \frac{\sum_{i=1}^{u} \langle \Psi_2(r_i), \Psi_2(s) \rangle}{\langle \Psi_2(s), \Psi_2(s) \rangle} \quad (23)$$

$$ROUGE - S(s) = \frac{(1+\beta^2) R_S P_S}{R_S + \beta^2 P_S} \quad (24)$$

The Formula (22) and the Formula (23) use $\Psi_2(d)$, which is the binary vector representation of ordered pairs of words. The values of the vector are set to if the pair exists in the document *d*, and it is set to 0 if it does not.

Table 1 – Correlations between rouge scores and human judge scores (all summarizers including human ones are included).

Score	Correlation
ROUGE-1	0.92465
ROUGE-2	0.80044
ROUGE-L	0.92269

In (Das and Martins 2007), it is stated that the ROUGE-2 is the best evaluation approach among ROUGE-N based approaches in terms of correlation between human judge scores and ROUGE scores. In the PhD thesis of (Steinberger 2007), it is explained that when all summarizers including human ones are included, correlation between humans and ROUGE-1 and ROUGE-L is strong. Table 1 gives related correlation values. When only system summarizers are considers, it is stated in the PhD thesis of (Steinberger 2007) that ROUGE-2 score correlates best with human judges (Table 2).

Table 2 – Correlations between rouge scores and human judge scores (only system summarizers).

Score	Correlation
ROUGE-1	0.90317
ROUGE-2	0.96119
ROUGE-L	0.91143

While performing the evaluation of the text summarization algorithms explained in this thesis, ROUGE-1, ROUGE-2, ROUGE-3, and ROUGE-L results are obtained, but discussions are made using only ROUGE-L results. Other ROUGE results behave in a similar manner as ROUGE-L score.

5.2 Evaluation Results

Different LSA approaches are executed on Turkish and English datasets using different input matrix creation methods. In order to make the resulting matrix smaller in size, stemming and stop word removal are applied. All the summaries created have length of 10% of the input document.

5.2.1 Evaluation Results for Turkish Datasets

Four different sets of Turkish documents are used for the evaluation of summarization approaches.

The first two sets of articles are scientific articles, related to different areas such as medicine, sociology, psychology. Each dataset contains fifty articles. The articles in the second dataset are longer than the articles in the first set. The evaluation is done by using abstracts of the input documents, which are human-generated summaries. The sentences of the abstracts may not match with the original sentences of the input document. The statistics about these data sets are given in Table 3.

Table 3 – Statistics of datasets Dataset1 and Dataset2

Statistics	Dataset1	Dataset2
Number of documents	50	50
Sentences per document	89,7	147,3
Words per document	2302,2	3435
Words per sentence	25,6	23,3

Using the Dataset1 and the Dataset2, different LSA approaches are executed using different input matrix creation methods. The evaluation of these two datasets is done using ROUGE approach. The ROUGE-L F-score values of DS1 can be found in the Table 4 and the ROUGE-L F-score values of DS2 can be found in the Table 5.

Table 4 – ROUGE-L f-measure scores for the data set DS1

ROUGE-L f-measure scores for the data set DS1		LSA Based Text Summarization Algorithms				
		G&L	S&J	MRC	Cross	Topic
Input Matrix Creation Methods	frequency	0,236	0,250	0,244	0,302	0,244
	binary	0,272	0,275	0,274	0,313	0,274
	tf-idf	0,200	0,218	0,213	0,304	0,213
	logentropy	0,230	0,250	0,235	0,302	0,235
	root type	0,283	0,282	0,289	0,320	0,289
	mod. tf-idf	0,195	0,221	0,223	0,290	0,223

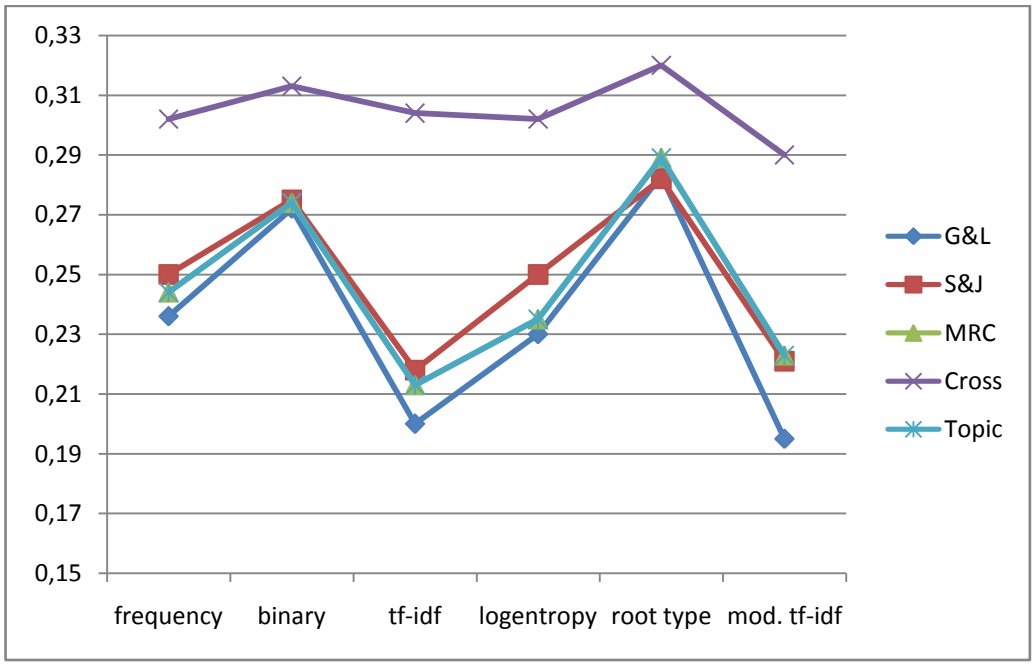

Figure 13 - ROUGE-L f-measure scores for the data set DS1
(Input matrix creation x Sentence selection alg.)

From the Table 4 and Table 5, it has been observed that Cross method has the highest score among the LSA based summarization approaches. Topic method has achieved better results than the approach of (Gong and Liu 2001), and has achieved same results as the approach of (Murray, Renals and Carletta 2005).

Table 5 – ROUGE-L f-measure scores for the data set DS2

ROUGE-L f-measure scores for the data set DS2		LSA Based Text Summarization Algorithms				
		G&L	S&J	MRC	Cross	Topic
Input Matrix Creation Methods	frequency	0,256	0,251	0,259	0,264	0,259
	binary	0,191	0,220	0,189	0,274	0,189
	tf-idf	0,230	0,235	0,227	0,266	0,227
	logentropy	0,267	0,245	0,268	0,267	0,268
	root type	0,194	0,222	0,197	0,263	0,197
	mod. tf-idf	0,234	0,239	0,232	0,268	0,232

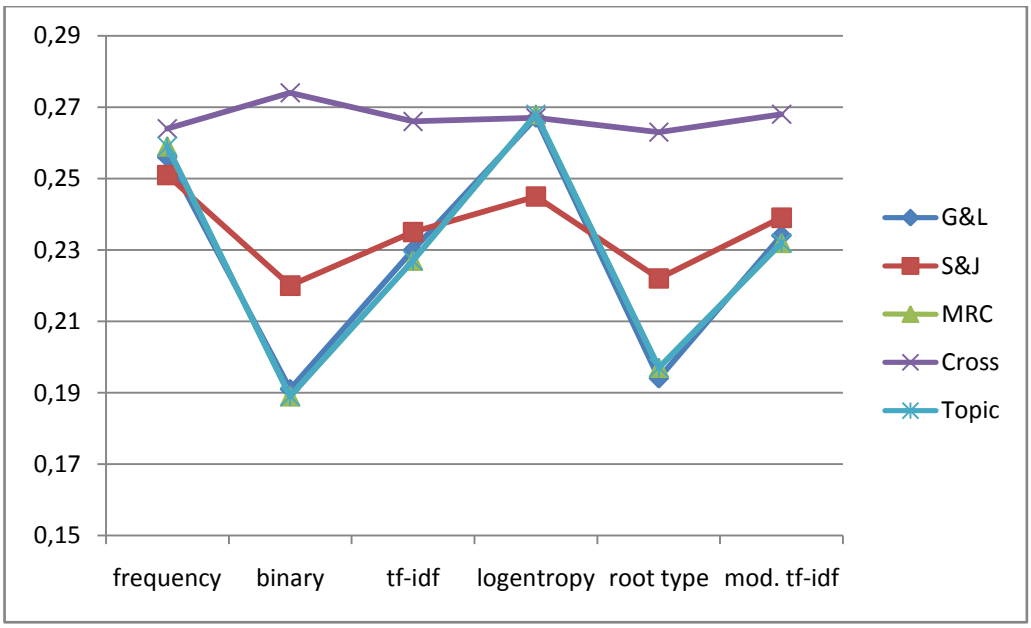

Figure 14 - ROUGE-L f-measure scores for the data set DS2
(Input matrix creation x Sentence selection alg.)

The third dataset, Dataset3, is composed of news texts in Turkish. It contains 120 texts. The number of sentences in news texts is usually less than scientific documents. This is also true for the third dataset whose number of sentences is not as high as scientific papers. The evaluation results of this dataset can be found in Table 6.

Table 6 – ROUGE-L f-measure scores for the Dataset3 (News Dataset)

ROUGE-L f-measure scores for the Dataset3		LSA Based Text Summarization Algorithms				
		G&L	S&J	MRC	Cross	Topic
Input Matrix Creation Methods	frequency	0.347	0.341	0.358	0.422	0.347
	binary	0.479	0.437	0.465	0.432	0.465
	tf-idf	0.303	0.295	0.313	0.389	0.305
	logentropy	0.350	0.316	0.359	0.225	0.350
	root type	0.501	0.449	0.470	0.454	0.471
	mod. tf-idf	0.319	0.291	0.339	0.387	0.320

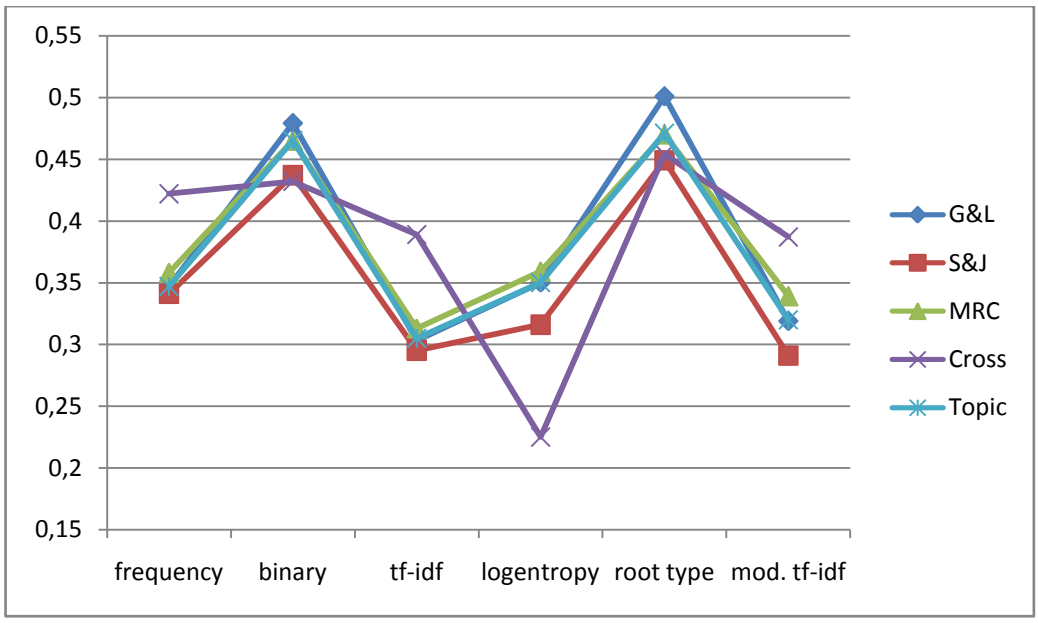

Figure 15 - ROUGE-L f-measure scores for the data set DS2
(Input matrix creation x Sentence selection alg.)

The results of the news dataset, Dataset3, show that Cross method does not work well with shorter documents. Topic method results are nearly the same as the results of the (Gong and Liu 2001) approach.

The fourth dataset in Turkish is a new dataset, which is used for the first time. The dataset, Dataset4, is composed of scientific articles in Turkish, related to different areas of science, such as sociology, biology, computer science and etc. The number of documents in this dataset is 153. The ROUGE-L F-score results can be seen in Table 7.

Table 7 – ROUGE-L f-measure scores for the new dataset of Turkish scientific articles (DataSet4)

ROUGE-L f-measure scores for the Dataset4		LSA Based Text Summarization Algorithms				
		G&L	S&J	MRC	Cross	Topic
Input Matrix Creation Methods	frequency	0,219	0,192	0,218	0,225	0,217
	binary	0,134	0,155	0,133	0,219	0,133
	tf-idf	0,212	0,203	0,205	0,231	0,205
	logentropy	0,232	0,191	0,232	0,229	0,231
	root type	0,232	0,191	0,232	0,229	0,231
	mod. tf-idf	0,211	0,202	0,203	0,230	0,202

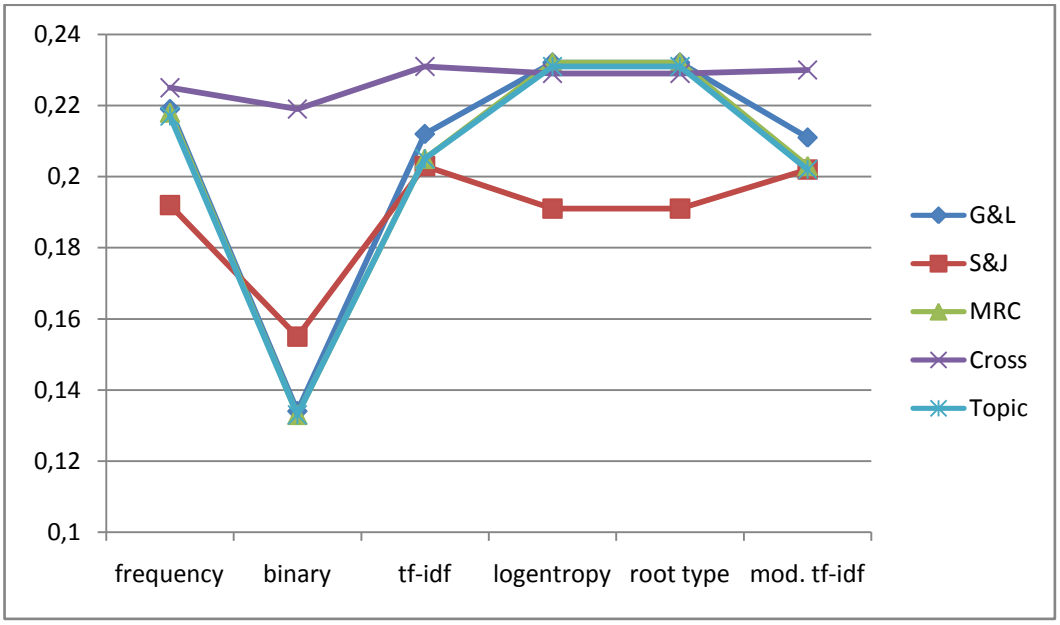

Figure 16 - ROUGE-L f-measure scores for the data set DS4
(Input matrix creation x Sentence selection alg.)

The results of the new dataset of Turkish scientific papers, Dataset4, show that Cross method works better than all other approaches. The results of Topic method are nearly the same as (Murray, Renals and Carletta 2005) approach, as observed in the first two datasets.

5.2.2 Evaluation Results for English Datasets

The datasets that are used for the evaluation of the LSA based summarization approaches are Duc2002 (Duc2002 Dataset 2002), Duc2004 (Duc2004 Dataset 2004) and Summac (Summac Dataset 2000) datasets. All datasets are used for single document summarization. There were tasks defined in the Duc datasets that limit the output summary size. Instead of limiting the summaries to the given sizes, the summary length is given as 10% of the original document.

The first dataset is Duc2002 dataset, which defines three different tasks. Since task-1 is related to single document summarization, this task is chosen. In this task, nearly

600 newspaper texts ("sixty sets of approximately 10 documents each") were given as input. The ROUGE-L F-scores and precision scores are given in Table 8.

Table 8 – ROUGE-L scores for Duc2002-Task1, a) F-measure scores

ROUGE-L f-measure scores for the Duc2002-Task1		LSA Based Text Summarization Algorithms				
		G&L	S&J	MRC	Cross	Topic
Input Matrix Creation Methods	frequency	0.093	0.095	0.098	0.179	0.097
	binary	0.234	0.224	0.230	0.196	0.230
	tf-idf	0.094	0.098	0.109	0.177	0.108
	logentropy	0.103	0.098	0.104	0.176	0.104
	root type	0.103	0.098	0.104	0.176	0.104
	mod. tf-idf	0.094	0.098	0.119	0.177	0.118

Table 9 – ROUGE-L scores for Duc2002-Task1 b) Precision scores

ROUGE-L precision scores for the Duc2002-Task1		LSA Based Text Summarization Algorithms				
		G&L	S&J	MRC	Cross	Topic
Input Matrix Creation Methods	frequency	0.323	0.287	0.315	0.324	0.315
	binary	0.304	0.299	0.301	0.323	0.301
	tf-idf	0.312	0.295	0.304	0.327	0.303
	logentropy	0.356	0.307	0.347	0.333	0.347
	root type	0.356	0.307	0.347	0.333	0.347
	mod. tf-idf	0.312	0.294	0.307	0.331	0.307

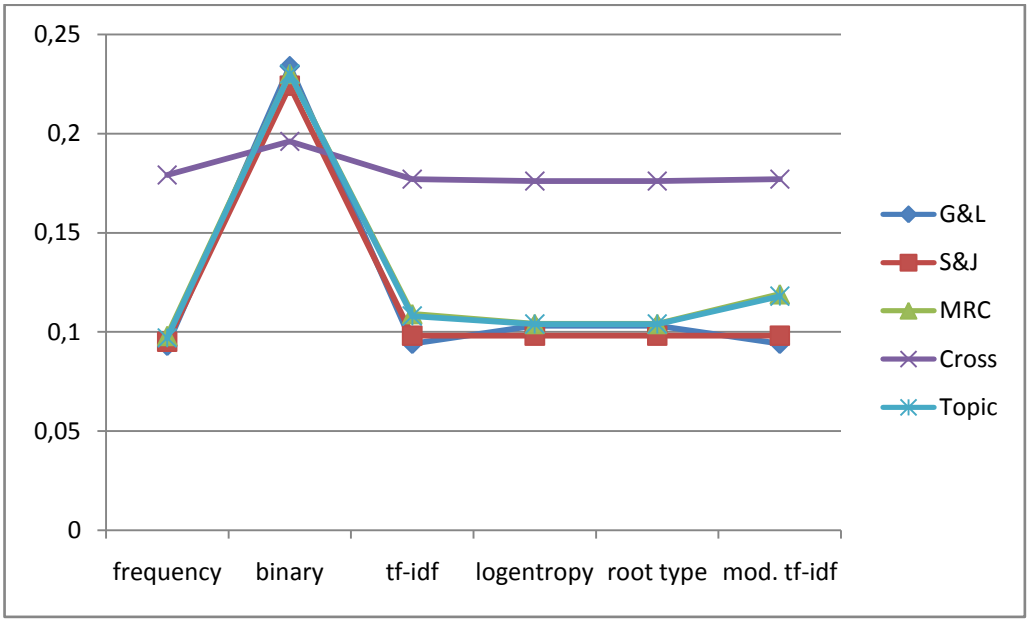

Figure 17- ROUGE-L f-measure scores for the data set Duc2002-Task1 (Input matrix creation x Sentence selection alg.)

The results for the Duc2002-Task1 show that Cross approach achieves the best result. Topic approach achieved nearly the same results as (Murray, Renals and Carletta 2005), and both have similar results to (Gong and Liu 2001) approach.

The second dataset is Duc2004 dataset. This dataset defines five different tasks and in this paper task1 is chosen in order to evaluate the summarization results. For this task, 50 sets of document clusters with nearly 10 documents each is given as input. The input documents are collected from the AP newswire and New York Times newswire. The aim is to create very short summaries, 75 bytes summaries, for each document. As in Duc2002 dataset, instead of limiting the output result to predefined bytes, the output summary length is set to 10% of original document.

Table 10 – ROUGE-L scores for Duc2004-Task1, a) F-measure scores

ROUGE-L f-measure scores for the Duc2004-Task1		LSA Based Text Summarization Algorithms				
		G&L	S&J	MRC	Cross	Topic
Input Matrix Creation Methods	frequency	0.087	0.070	0.082	<u>0.102</u>	0.080
	binary	<u>0.103</u>	0.094	0.102	0.097	0.101
	tf-idf	0.072	0.064	0.074	<u>0.093</u>	0.071
	logentropy	0.088	0.068	0.082	<u>0.101</u>	0.081
	root type	0.088	0.068	0.082	<u>0.101</u>	0.081
	mod. tf-idf	0.075	0.063	0.080	<u>0.098</u>	0.074

Table 11 – ROUGE-L scores for Duc2004-Task1, b) Precision scores

ROUGE-L precision scores for the Duc2004-Task1		LSA Based Text Summarization Algorithms				
		G&L	S&J	MRC	Cross	Topic
Input Matrix Creation Methods	frequency	<u>0.078</u>	0.064	0.074	0.072	0.072
	binary	0.064	0.058	0.063	<u>0.065</u>	0.063
	tf-idf	0.063	0.057	0.063	<u>0.064</u>	0.061
	logentropy	<u>0.081</u>	0.065	0.075	0.071	0.075
	root type	<u>0.081</u>	0.065	0.075	0.071	0.075
	mod. tf-idf	0.063	0.057	0.064	<u>0.068</u>	0.060

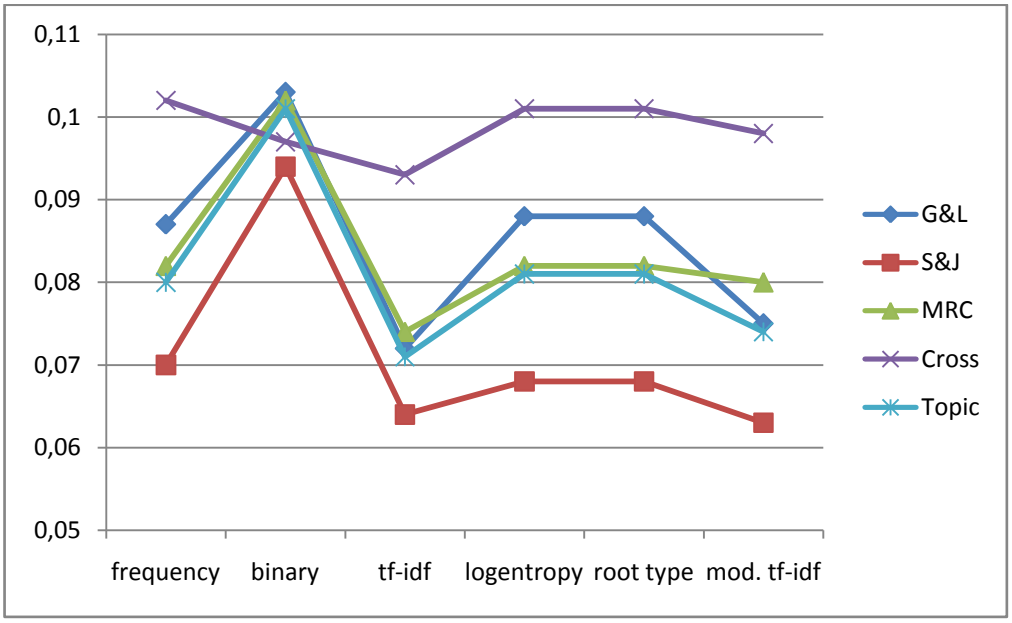

Figure 18 - ROUGE-L f-measure scores for the data set Duc2004-Task1 (Input matrix creation x Sentence selection alg.)

The results for Duc2004-Task1 are given in Table 10. As observed in Duc2002-Task1, Cross approach achieved best results among other approaches and Topic approach achieved similar results as (Murray, Renals and Carletta 2005).

The third dataset is Summac Dataset which contains 183 documents. All documents are scientific articles about computer science collected from ACL sponsored conferences. The comparison of extracted summaries is done against the abstracts of the given input articles. The ROUGE-L F-scores for this dataset are given in Table 12.

Table 12 – ROUGE-L F-measure scores for Summac Dataset

ROUGE-L f-measure scores for the Summac Data-set		LSA Based Text Summarization Algorithms				
		G&L	S&J	MRC	Cross	Topic
Input Matrix Creation Methods	frequency	0.161	0.131	0.156	0.177	0.156
	binary	0.118	0.136	0.119	0.177	0.119
	tf-idf	0.129	0.126	0.121	0.182	0.121
	logentropy	0.180	0.138	0.180	0.182	0.180
	root type	0.180	0.138	0.180	0.182	0.180
	mod. tf-idf	0.127	0.126	0.122	0.187	0.122

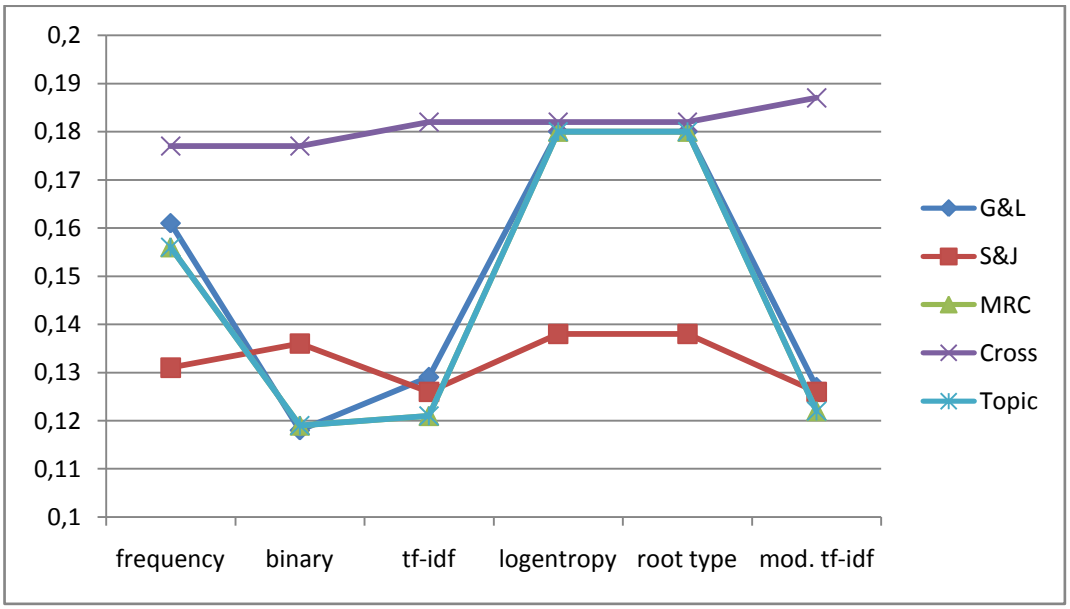

Figure 19 - ROUGE-L f-measure scores for the data set Summac
(Input matrix creation x Sentence selection alg.)

The results in Table 12 show that Cross approach gets the best result when using scientific papers. Topic method gets the same results as the approach of (Murray, Renals and Carletta 2005) and similar results to the approach of (Gong and Liu 2001).

5.2.3 Comparison against Other Summarization Approaches

In this thesis, evaluation of the LSA based summarization systems is done using Turkish and English datasets. The evaluation is based on ROUGE scores. As stated in the PhD thesis of (Steinberger 2007), different ROUGE scores give different correlation with human judgment and strongest correlation is observed with ROUGE-1 and ROUGE-L scores.

The datasets in English are common datasets for the evaluation of the summarization systems. The first dataset used is Duc2002 dataset. In order to be able to compare LSA based approaches with other approaches that used Duc2002 dataset, different resources are used and their evaluation results for summarization are collected.

Using the MSc. thesis of (Ercan 2006), the ROUGE-L results for lexical chains based summarization systems are collected. From the paper of (Wan, Yang and Xiao 2007), results of approaches belonging to (Wan, Yang and Xiao 2007) and algorithms named SentenceRank (Mihalcea and Tarau 2004) and MutualRank (Zha 2002) are collected. The result for TextRank which is based on graphs is collected from the paper of (Mihalcea 2004). From the paper of (Patil and Brazdil 2007), the evaluation results for SumGraph, MEAD (Radev, Blair-Goldensohn and Zhang 2001), and LexRank (Erkan and Radev 2004) are obtained. The results for LSA based summarization algorithms collected from the evaluation results stated in Section 5.2.2 in Chapter 5. In this Section, different results for different input matrix creation approaches have been obtained. For the comparison, best results are obtained from these different approaches. All results are shown on the Table 13.

Table 13 – Comparison of precision scores on Duc2002 dataset,

a) ROUGE-L precision scores

Text Summarization Algorithms	ROUGE-L (precision)
Barzilay	0.309
Ercan	0.285
Gong-Liu	0.356
Steinberger-Jezek	0.307
Murray et al.	0.347
Cross	0.333
Topic	0.347

Table 14 – Comparison of precision scores on Duc2002 dataset,

b) ROUGE-1 precision scores

Text Summarization Algorithms	ROUGE-1 (precision)
Wan-WordNet	0.473
Wan-Corpus	0.472
SentenceRank	0.462
MutualRank	0.438
TextRank-HITS	0.502
TextRank-PageRank	0.500
SumGraph	0.484
MEAD	0.472
LexRank	0.469
Gong-Liu	0.432
Steinberger-Jezek	0.428
Murray et al.	0.428
Cross	0.453
Topic	0.428

Another dataset that we have used for evaluating the results for English documents is Duc2004 dataset. There are different tasks defined over this dataset, and we have chosen Task1, which is creation of very short summary of a single document. As in the Duc2002 dataset, we have collected results of different algorithms using different resources.

The results for lexical chains based approaches are collected from the MSc. thesis of (Ercan 2006). The results for a machine learning based algorithm (Dublin) are collected from the study of (Doran, et al. 2004). The last two algorithms are TF ad Hybrid algorithms, whose results are collected from the study of (Wang, Dunnion and Carthy 2005). Another machine learning based algorithm is LAKE, improved by (D'Avanzo, Magnini and Vallin 2004). As in Duc2002 dataset, results of LSA approaches are collected from the Section 5.2.2 in Chapter 5. On the Table 15, comparison of different algorithms that performed Duc2004-Task1 can be seen.

Table 15 – Comparison of precision scores on Duc2004 dataset,
a) ROUGE-L precision scores

Text Summarization Algorithms	ROUGE-L (precision)
Barzilay	0.155
Ercan	0.170
Dublin	0.176
TF	0.171
Hybrid	0.176
LAKE	0.156
Gong-Liu	0.081
Steinberger-Jezek	0.065
Murray et al.	0.075
Cross	0.072
Topic	0.075

Table 16 – Comparison of precision scores on Duc2004 dataset,
b) ROUGE-1 precision scores

	ROUGE-1 (precision)
Barzilay	0.178
Ercan	0.195
Dublin	0.219
TF	0.244
Hybrid	0.219
LAKE	0.188
Gong-Liu	0.090
Steinberger-Jezek	0.071
Murray et al.	0.083
Cross	0.085
Topic	0.083

5.2.4 Analysis of Evaluation Results

The evaluation of LSA based summarization systems are performed on multiple datasets that are in Turkish and in English. The evaluation is performed using ROUGE system.

The evaluation of LSA based summarization systems on Turkish documents is done using four different datasets. The evaluation results of these datasets can be seen on Section 5.2.1. From the results, it has been observed that Cross method works better than other LSA based approaches. The results for the Topic method are usually same as the results of the approach of (Murray, Renals and Carletta 2005). Also, it is observed that, the Cross method does not perform well with shorter documents, such as news texts.

The evaluation results performed on English datasets are introduced in Section 5.2.2. For the evaluation of approaches on English documents, three different datasets are

used- Duc2002, Duc2004, and Summac datasets. Among different LSA based summarization approaches, the Cross method performed better than the others. It is observed that the Topic method's performance is nearly the same as the approach of (Gong and Liu 2001). Also, as observed in Turkish datasets, it is observed that the performance of the LSA based approaches is lower for shorter documents.

In both of the Turkish and English datasets, the evaluation of the approaches is done using different input matrix creation methods. Different summarization algorithms performed differently for each input matrix creation approach. But it is observed that the Cross approach is not affected from the different methods of input matrix creation, and it performed nearly equally well in all approaches.

The input matrix creation approaches that are used in this paper are all well known approaches in literature. The modified tf-idf approach is a newer approach which is proposed in this thesis. It has been observed that this new approach does not produce better results when the input document is short. This is the case, because every word/sentence in shorter document carries information, and removing some of them may remove important information.

Comparison of LSA based summarization systems against other summarization approaches is explained in Section 5.2.3. From the tables in that section, it has been observed that, LSA based algorithms do not perform as good as machine learning based algorithms or other algorithms that uses external information. But attention should be given to the point that LSA based algorithms uses information in the input document only.LSA based algorithms are unsupervised, and they do not need any outer information to extract semantic information that exists in the document.

Another concern related to LSA based algorithms is that they do not perform well while creating shorter summaries. LSA based summarization approaches are extractive approaches, and as stated in the paper of (Das and Martins 2007), there is a claim of (Witbrock and Mittal 1999) which states that extractive summarization methods are not very efficient when creating very short summaries. This observation is

also done by other researchers stating that finding a single or a few sentences that gives the main idea of a text is very difficult (Doran, et al. 2004).

CHAPTER 6

CONCLUSION

Finding out the information related to the needs of a user among large number of documents is a problem that has come with the growth of text based resources. In order to solve this problem, text summarization methods are proposed and evaluated. The research on summarization started with the extraction of simple features and improved to use different methods, such as lexical chains, statistical approaches, graph based approaches, and algebraic solutions. One of the algebraic-statistical approaches is Latent Semantic Analysis method.

In this thesis, general information about text summarization is given before explaining the Latent Semantic Analysis (LSA). After giving information on LSA, text summarization methods based on LSA are explained, such as approaches of (Gong and Liu 2001), (Steinberger and Jezek 2004), (Murray, Renals and Carletta 2005), and approaches proposed in the thesis, namely Cross and Topic methods.

All approaches are evaluated on Turkish and English datasets. For the evaluation of the results, ROUGE scores are used. Comparison of the results against other text summarization approaches is also done.

The results show that the Cross method performs better than all other LSA based approaches. Another important result of this approach is that it is not affected by different input matrix creation methods. Also, it is observed that the Cross and Topic methods, which are proposed in this thesis, perform equally well on both Turkish and

English datasets. This work shows that Cross and Topic methods can be used in any language for summarization purposes.

In future, ideas that are used in other methods, such as graph based approaches, will be used together with the proposed approaches to improve the performance of the summarization system.

REFERENCES

Aone, Chinatsu, Marry Ellen Okurowski, James Gorlinsky, and Bjornar Larsen. "A trainable Summarizer with Knowledge Acquired from Robust NLP Techniques." In *Advances in Automatic Text Summarization*, by Inderjeet Mani and Mark T. Maybury, 71-80. MIT Press, 1999.

Baeza-Yates, Ricardo, and Berthier Ribeiro-Neto. *Modern Information Retrieval.* Addison Wesley, 1999.

Barzilay, Regina, and Michael Elhadad. "Using Lexical Chains for Text Summarization." *In Proceedings of the Intelligent Scalable Text Summarization Workshop (ISTS'97).* Madrid: ACL, 1997. 10-17.

Baxendale, P B. "Machine-made index for technical literature: an experiment." *IBM Journal of Research and Development* 2 (1958): 354-361.

Brandow, Ronald, Karl Mitze, and Lisa F Rau. "Automatic condensation of electronic publications by sentence selection." *Information Processing and Management: an International Journal - Special issue: summarizing text* 31 (1995): 675-685.

Brin, Sergey, and Lawrence Page. "The anatomy of a large-scale hypertextual Web search engine." *Computer Networks and ISDN Systems* 30 (1998): 1-7.

Conroy, John M, and Dianne P O'leary. "Text summarization via hidden Markov models." *SIGIR '01 Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval.* ACM New York, 2001. 406-407.

D'Avanzo, Ernesto, Bernardo Magnini, and Alessandro Vallin. "Keyphrase Extraction for Summarization Purposes: The LAKE System at DUC-2004." *Proceedings of the 2004 document understanding conference.* 2004.

Das, Dispanjan, and Andre F.T Martins. *A Survey on Automatic Text Summarization.* Literature survey for Language and Statistics II, Carnegie Mellon University, 2007.

Deerwester, Scott, Susan T Dumais, Thomas K Landauer, George W Furnas, and Richard Harshman. "Indexing by Latent Semantic Analysis." *Journal of the American Society for Information Science and Technology (JASIST)* 41 (1990).

DeJong, Gerald F. "Fast Skimming of News Stories: The FRUMP System." PhD Thesis, Computer Science Department, Yale University, 1978.

Doran, William, Nicola Stokes, Eamonn Newman, John Dunnion, Joe Carthy, and Fergus Toolan. "News Story Gisting at University College Dublin." *Document Understanding Conference.* 2004.

Duc2002 Dataset. 2002. http://duc.nist.gov/duc2002/ (accessed 20 December 2010).

Duc2004 Dataset. 2004. http://duc.nist.gov/duc2004/ (accessed 20 December 2010).

Edmundson, H P. "New methods in automatic extracting." *Journal of the ACM* 16 (1969): 264-285.

Ercan, Gönenç. "Automated Text Summarization and Keyphrase Extraction." MSc Thesis, Computer Engineering Department, Bilkent University, 2006.

Ercan, Gönenç, and İlyas Çiçekli. "Lexical Cohesion based Topic Modeling for Summarization." *CICLing'08 Proceedings of the 9th international conference on Computational linguistics and intelligent text processing.* 2008. 582-592.

Erkan, Güneş, and Dragomir R. Radev. "LexRank: Graph-based Lexical Centrality as Salience in Text Summarization." *Journal of Artificial Intelligence Research* 22 (2004): 457-479.

Gong, Yihong, and Xin Liu. "Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis." *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR'01).* ACM, 2001. 19-25.

Hahn, Udo, and Inderjeet Mani. "The challenges of automatic summarization." *Computer* 33 (2000): 29-36.

Hassel, Martin. "Resource Lean and Portable Automatic Text Summarization." PhD thesis, 2007.

Hovy, Eduard, and Chin-Yew Lin. "Automated Text Summarization in SUMMARIST." In *Advances in automaic Text Summarization*, by Inderjeet Mani and Mark T Maybury, 81-94. MIT Press, 1999.

Jezek, Karel, and Josef Steinberger. "Automatic Text Summarization (The state of the art 2007 and new challenges)." *Znalosti 2008.* Bratislava, Slovakia, 2008. 1-12.

Jones, Karen. "Automatic Summarising: Factors and Directions." In *Advances in Automatic Text Summarization*, by Inderjeet Mani and Mark T Maybury, 1-12. MIT Press, 1999.

Kleinberg, Jon M. "Authoritative sources in a hyper-linked environment." *Journal of the ACM* 46 (1999): 604-632.

Knight, Kevin, and Daniel Marcu. "Statistics-based summarization-Step one: Sentence compression." *Seventeenth National Conference on Artificial Intelligence*

and Twelfth Conference on Innovative Applications of Artificial Intelligence (AAAI-2000). 2000. 703-710.

Kolda, Tamara G, and Dianne P O'Leary. "A semidiscrete matrix decomposition for latent semantic indexing information retrieval." *ACM Transactions on Information Systems (TOIS)* 16 (1998): 322-346.

Kupiec, Julian, Jan Pedersen, and Francine Chen. "A Trainable Document Summarizer." *SIGIR '95 Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval.* 1995. 68-73.

Landauer, Thomas K, Pete W Foltz, and Darrell Laham. "An introduction to Latent Semantic Analysis." *Discourse Processes* 25 (1998): 259-284.

Lee, Daniel D, and H Sebastian Seung. "Learning the parts of objects by non-negative matrix factorization." *Nature* 401 (1999): 788-791.

Lin, Chin-Yew. "ROUGE: a Package for Automatic Evaluation of Summaries." *Workshop on Text Summarization Branches Out (WAS 2004).* 2004. 25-26.

Lin, Chin-Yew, and Eduard Hovy. "Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics." *NAACL '03 Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology.* Association for Computational Linguistics, 2003. 71-78.

Lin, Chin-Yew, and Eduard Hovy. "Identifying topics by position." *ANLC '97 Proceedings of the fifth conference on Applied natural language processing.* Association for Computational Linguistics, 1997. 283-290.

Luhn, H P. "The Automatic Creation of Literature Abstracts." *IBM Journal of Research and Development* 2 (1958): 159-165.

Luk, Robert Wing Pong. "An IBM-PC environment for Chinese corpus analysis." *COLING '94 Proceedings of the 15th conference on Computational linguistics.* Association for Computational Linguistics, 1994. 584-587.

Marcu, Daniel. "From Discourse Structures to Text Summaries." *Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization.* 1997. 82-88.

Mihalcea, Rada. "Graph-based Ranking Algorithms for Sentence Extraction, Applied to Text Summarization." *ACLdemo '04 Proceedings of the ACL 2004 on Interactive poster and demonstration sessions.* Association for Computational Linguistics, 2004. 170-173.

Mihalcea, Rada, and Paul Tarau. "Text-rank: bringing order into texts." *Proceeding of the Conference on Empirical Methods in Natural Language Processing- EMNLP (2004).* 2004. 404-411.

Morris, Andrew H, George M Kasper, and Dennis A Adams. "The Effects and Limitations of Automatic Text Condensing on Reading Comprehension Performance." *Information Systems Research* 3 (1992): 17-35.

Murray, Gabriel, Steve Renals, and Jean Carletta. "Extractive summarization of meeting recordings." *Proceedings of the 9th European Conference on Speech Communication and Technology.* 2005.

Nenkova, Ani, and Rebecca Passonneau. "Evaluating Content Selection in Summarization: The Pyramid Method." *HLT/NAACL*. 2004.

Ono, Kenji, Kazuo Sumita, and Seiji Miike. "Abstract Generation Based on Rhetorical Structure Extraction." *COLING '94 Proceedings of the 15th conference on Computational linguistics.* 1994. 344-348.

Osborne, Miles. "Using maximum entropy for sentence extraction." *AS '02 Proceedings of the ACL-02 Workshop on Automatic Summarization.* 2002. 1-8.

Patil, Kaustubh, and Pavel Brazdil. "Sumgraph: Text summarization using centrality in the pathfinder network." *International Journal on Computer Science and Information Systems* 2 (2007): 18-32.

Porter Stemmer. 2000. http://www.tartarus.org/~martin/PorterStemmer (accessed 20 December 2010).

Porter, Martin. "An algorithm for suffix stripping." *Program* (British Library) 14, no. 3 (1980): 130-137.

Qazvinian, Vahed, and Dragomir R Radev. "Scientific paper summarization using citation summary networks." *COLING '08 Proceedings of the 22nd International Conference on Computational Linguistics.* 2008. 689-696.

Radev, Dragomir R, Eduard Hovy, and Kathleen McKeown. "Introduction to the special issue on summarization." *Computational Linguistics* 28 (2002): 399-408.

Radev, Dragomir R, et al. "Evaluation Challenges in Large-scale Document Summarization." *ACL '03 Proceedings of the 41st Annual Meeting on Association for Computational Linguistics.* Association for Computational Linguistics, 2003. 375-382.

Radev, Dragomir R, Hongyan Jing, and Malgorzata Budzikowska. "Centroid-based summarization of multiple documents." *NAACL-ANLP-AutoSum '00 Proceedings of the 2000 NAACL-ANLPWorkshop on Automatic summarization.* Association for Computational Linguistics Morristown, 2000. 21-30.

Radev, Dragomir R, Sasha Blair-Goldensohn, and Zhu Zhang. "Experiments in single and multi-document summarization using MEAD." *Document Understanding Conference.* 2001.

Rau, Lisa F, and Paul S Jacobs. "Creating segmented databases from free text for text retrieval." *SIGIR '91 Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, 1991. 337-346.

Steinberger, Josef. "Text Summarization within the LSA Framework." PhD Thesis, 2007.

Steinberger, Josef, and Karel Jezek. "Latent Semantic Analysis in Text Summarization and Summary Evaluation." *Proceedings of ISIM'04.* 2004. 93-100.

Stop Words List-English. 2010. http://www.ranks.nl/resources/stopwords.html (accessed 20 December 2010).

Stop Words List-Turkish. 2010. http://www.ranks.nl/stopwords/turkish.html (accessed 20 December 2010).

Summac Dataset. 2000. http://www.itl.nist.gov/iaui/894.02/related_projects/tipster_summac/cmp_lg.html (accessed 20 December 2010).

Svore, Krysta, Lucy Vanderwende, and Chris Burges. "Enhancing single-document summarization by combining RankNet and third-party sources." *Proceedings of EMNLP-CoNLL.* 2007. 448-457.

Teufel, Simone, and Marc Moens. "Sentence extraction as a classification task." *ACL/EACL workshop on" Intelligent and scalable Text summarization.* 1997. 58-65.

Wan, Xiaojun, Jianwu Yang, and Jianguo Xiao. "Towards an Iterative Reinforcement Approach for Simultaneous Document Summarization and Keyword Extraction." *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics.* 2007. 552-559.

Wang, Ruichao, John Dunnion, and Joe Carthy. "Machine Learning Approach To Augmenting News Headline Generation." *Second International Joint Conference on Natural Language Processing: Companion Volume including Posters/Demos and tutorial abstracts.* 2005.

Witbrock, Michael J, and Vibhu O Mittal. "Ultra-summarization: a statistical approach to generating highly condensed non-extractive summaries." *SIGIR '99 Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, 1999. 315-316.

Zemberek. 1999. http://code.google.com/p/zemberek/ (accessed 20 December 2010).

Zha, Hongyuan. "Generic summarization and keyphrase extraction using mutual reinforcement principle and sentence clustering." *SIGIR '02 Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval.* ACM, 2002. 113- 120.