DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

SEMANTIC TEXT MINING AND AN APPLICATION IN TURKISH DOCUMENTS

by
Volkan UZUN

August, 2014
İZMİR

SEMANTIC TEXT MINING AND AN APPLICATION IN TURKISH DOCUMENTS

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Master of Science
in Statistics, Statistics Program

by

Volkan UZUN

August, 2014

İZMİR

M.Sc. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"SEMANTIC TEXT MINING AND AN APPLICATION IN TURKISH DOCUMENTS"** completed by **VOLKAN UZUN** under supervision of **ASSIST. PROF. DR. ENGİN YILDIZTEPE** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Assist. Prof. Dr. Engin YILDIZTEPE

Supervisor

Yrd. Doç. Dr. Şen Çakır

(Jury Member)

Doç.Dr. A.Fırat Özdemir

(Jury Member)

Prof. Dr. Ayşe OKUR
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGMENTS

I am heartily thankful to my supervisor, Assist. Prof. Dr. Engin YILDIZTEPE. With his warm encouragement, thoughtful guidance and his great efforts to explain things clearly and simply, he was always available for my questions and he was positive and gave generously of his time and vast knowledge.

I wish to thank my dear companion Bahar YALÇIN for helping me get through the difficult times, and for all the emotional support, patience, camaraderie, entertainment, and caring she provided. Also, I am grateful to for Dr. Alper VAHAPLAR who helped me during this thesis.

Finally, I thank my family for supporting me throughout all my education life. They raised me, supported me, taught me, and loved me. This research would not have been achievable without them.

Volkan UZUN

SEMANTIC TEXT MINING AND AN APPLICATION IN TURKISH DOCUMENTS

ABSTRACT

Text mining has an increasing significance with the developing Web technologies**.** Text mining is a variation of data mining. It refers to the process of deriving information form unstructured textual data. Text mining is a multidisciplinary field related with statistics, information retrieval, natural language processing, linguistics and machine learning. Text mining approaches can be used for many different purposes. Semantic analysis is one of the text mining approaches. Numerous studies can be found about the semantic analysis for English. However, using semantic analysis methods for Turkish is not frequent. The reason for this is the structural difficulties of Turkish. In order for these methods to work efficiently the stemming and stop-word removal operations should be performed accurately. There are stemming algorithms for Turkish, yielding successful results but none of them has become a golden standard yet.

Latent Semantic Analysis (LSA), Probabilistic Latent Semantic Analysis (PLSA) and Latent Dirichlet Allocation (LDA) are the most common semantic analysis techniques. In this thesis, information on semantic analysis methods was provided and the performances of these methods to compute semantic similarity between Turkish texts were evaluated. Also Turkish morphology and text representation methods were explained. LSA, PLSA and LDA algorithms were applied on artificial and real data sets and results were evaluated. Artificial data sets were generated with different scenarios. Real data sets were collected from different news agencies with different topics.

All these algorithms perform while number of topics is known. However, number of topics may not be known in practice. In this study, an iterative algorithm using PLSA and LSA methods together, which yields accurate results when the number of topics is not known, was discussed. For the future, new approaches should be

devised for enabling use of semantic analysis in cases where the number of topics is not known.

Keywords: Text mining, semantic analysis, latent semantic analysis, probabilistic latent semantic analysis, latent Dirichlet allocation.

ANLAMSAL METİN MADENCİLİĞİ VE TÜRKÇE DÖKÜMANLAR ÜZERİNE BİR UYGULAMA

ÖZ

Metin madenciliğinin önemi gelişen web teknolojileri ile birlikte artmaktadır. Veri madenciliğinin bir alt türü olarak değerlendirilen metin madenciliğinde, metin halindeki yapılandırılmamış veri setlerinden anlamlı bilgiler çıkartılır. Metin madenciliği, istatistik, bilgi geri kazanımı, doğal dil işleme, dilbilim ve makine öğrenimi gibi birçok alan ile ilişkili bir araştırma alanıdır. Metin madenciliği birçok farklı alanda kullanılabilir. Anlamsal analiz bunlardan biridir. İngilizce için bir çok anlamsal analiz çalışması bulunmaktadır ancak Türkçe için anlamsal analiz çalışmaları oldukça azdır. Bunun sebebi Türkçenin yapısal zorluklarıdır. Anlamsal analiz yöntemlerini etkin bir şekilde kullanabilmek için gövdeleme (ek-kök ayrıştırması) ve anlamsız (cümle içinde tek başına anlam ifade etmeyen, edat bağlaç vb. gibi) kelimelerin ayrıştırılması işlemlerinin doğru yapılması gerekmektedir. Türkçe için başarılı sonuçlar veren gövdeleme algoritmaları bulunmaktadır ancak henüz hiçbiri standart hale gelmemiştir.

Gizli Anlam Analizi (LSA), Olasılıksal Gizli Anlam Analizi (PLSA) ve Gizli Dirichlet Bölüştürmesi (LDA) anlamsal analiz yöntemlerinin başında gelmektedir. Bu tez çalışmasında, anlamsal analiz yöntemleri hakkında bilgi verilmiş ve bu yöntemlerin Türkçe metinler arasındaki anlamsal benzerliği hesaplamadaki performansları değerlendirilmiştir. Ayrıca Türkçenin morfolojisi ve metin temsil yöntemleri açıklanmıştır. LSA, PLSA ve LDA algoritmaları yapay ve gerçek veri setlerine uygulanmış ve sonuçlar değerlendirilmiştir. Yapay veri setleri farklı senaryolar ile üretilmiştir. Gerçek veri setleri ise farklı haber ajanslarından farklı konularda haberler derlenerek oluşturulmuştur.

Anlamsal analiz yöntemleri konu sayısının bilindiği durumlarda kullanılabilirler. Ancak pratikte her zaman konu sayısı bilinmeyebilir. Bu çalışmada ayrıca konu sayısının bilinmediği durumlarda kullanılmak üzere, LSA ve PLSA'nın birlikte

kullanıldığı yinelemeli bir algoritma önerilmiştir. Yapılan uygulamalarda bu algoritmanın başarılı sonuçlar verdiği görülmüştür. Gelecek çalışmalarda özellikle konu sayısının bilinmediği durumlarda da çalışabilecek yeni anlamsal analiz yaklaşımları geliştirilmelidir.

Anahtar kelimeler: Metin madenciliği, anlamsal analiz, gizli anlam analizi, olasılıksal gizli anlam analizi, gizli Dirichlet bölüştürmesi.

CONTENTS

Page

M.Sc. THESIS EXAMINATION RESULT FORM ii
ACKNOWLEDGEMENTS iii
ABSTRACT iv
ÖZ vi
LIST OF FIGURES x
LIST OF TABLES xi

CHAPTER ONE – INTRODUCTION 1

CHAPTER TWO-TEXT REPRESENTATION AND DATA PREPERATION . 4

2.1 Text Representation 4
2.1.1 N-gram Method 4
2.1.2 Term Frequency Method 5
2.1.3 Bag of Words Method 5
2.2 Data Preparation 6
2.2.1 Turkish Morphology 6
2.2.2 Stemming 9
2.2.3 Stop-Words Removal 10

CHAPTER THREE - SEMANTIC TEXT MINING ALGORITHMS 11

3.1 Latent Semantic Analysis (LSA) 11
3.1.1 Singular Value Decomposition (SVD) 13
3.1.1.1 Deriving the Component U 13
3.1.1.2 Deriving the Component V 15
3.1.1.3 Deriving the Component S 16
3.1.2 A LSA Example for English 16
3.2 Probabilistic Latent Semantic Analysis (PLSA) 19
3.2.1 A PLSA Application for English 21

3.3 Latent Dirichlet Allocation (LDA) 23
3.3.1 Notation and Terminology 23
3.3.2 LDA Steps 24
3.4 LSA after PLSA 26

CHAPTER FOUR – APPLICATIONS 28

4.1 Artificial Data Sets 28
4.1.1 LSA Applications with Artificial Data Sets 29
4.1.2 PLSA Applications with Artificial Data Sets 30
4.1.3 LDA Applications with Artificial Data Sets 31
4.2 Real Data Sets 31
4.2.1 LSA Applications for Real Data Sets 35
4.2.2 PLSA Applications for Real Data Sets 37
4.2.3 LDA Applications for Real Data Sets 39
4.2.4 LSA after PLSA Applications for Real Data Sets 41

CHAPTER FIVE – CONCLUSION 44

REFERENCES 46

APPENDICES 50

A1 MATLAB codes for LSA 50
A2 MATLAB codes for PLSA 50
A3 R codes for LDA 54
A4 MATLAB codes for LSA after PLSA 55

LIST OF FIGURES

Page

Figure 3.1 Graphical model representation of LDA 24
Figure 3.2 Dirichlet density plots, k = 3 25

LIST OF TABLES

Page

Table 2.1 Vowels in Turkish language 7
Table 2.2 Some Turkish suffixes 9
Table 3.1 Texts used for LSA example 16
Table 3.2 Text-Word table 17
Table 3.3 The Correlation matrix of reduced A matrix 19
Table 3.4 Word probabilities P(w|z) randomly generated before the analysis 21
Table 3.5 Text probabilities P(d|z) randomly generated before the analysis 21
Table 3.6 Latent class probabilities P(z) randomly generated before the analysis 22
Table 3.7 Word probabilities P(w|z) after the first iteration 22
Table 3.8 Text probabilities P(d|z) after the first iteration 22
Table 3.9 Latent Semantic probabilities P(z) after the first iteration 22
Table 3.10 Text probabilities P(d|z) for English 22
Table 4.1 Data set-1: Generated word-document table while all documents belong to different topics 29
Table 4.2 Data set-2: Generated word-document table while all documents belong to same topic 29
Table 4.3 Correlation table obtained as a result of the LSA on data set-1 (different topics) 29
Table 4.4 Correlation table obtained as a result of the LSA on data set-2 (same topics) 30
Table 4.5 PLSA text probabilities P(d|z) for data set-1 (different topics) 30
Table 4.6 PLSA text probabilities P(d|z) for data set-2 (same topics) 31
Table 4.7 The results of LDA application on data set-1 (different topics) 31
Table 4.8 Real dataset-1: News texts with four topics 32
Table 4.9 Real dataset-2: News texts with three topics 34
Table 4.10 A part of the word-document table created for real data set-1 with four topics 35
Table 4.11 Correlation table obtained as a result of the LSA (real data set-1 with four topics) 36

Table 4.12 A part of the word-document table created for real data set-2 with three topics 37
Table 4.13 Correlation table obtained as a result of the LSA (real data set-2 with three topics) 37
Table 4.14 PLSA text probabilities P(d|z) for real data set-1 with four topic..... 38
Table 4.15 PLSA text probabilities P(d|z) for real data set-2 with three topics..... 38
Table 4.16 A part of the word-document table created for real data set-1 with four topics in LDA 39
Table 4.17 The results of LDA application on real data set-1 with four topics..... 40
Table 4.18 A part of the word-document table created for real data set-2 with three topics in LDA 40
Table 4.19 The results of LDA application on real data set-2 with three topics..... 41
Table 4.20 Table of frequencies of the desired P(d|z) matrix on real data set-1 with four topics..... 42
Table 4.21 The correlation table resulted from LSA after PLSA for real data set-1 with four topics..... 42
Table 4.22 Table of frequencies of the desired P(d|z) matrix on real data set-2 with three topics 43
Table 4.23 The correlation table resulted from LSA after PLSA for real data set-2 with three topics 43

CHAPTER ONE
INTRODUCTION

In recent years data mining techniques has become one of the most important tools for extracting data and establishing patterns. Due to technological advances, amount of data increased and extracting to useful information became difficult. Under these circumstances the importance of data mining gradually increases. Data mining is the process of discovering patterns in data sets. It is a process that to turn raw data into useful information. Data mining is an integration of techniques from multiple disciplines such as statistics, mathematics, information retrieval, machine learning, neural networks, natural language processing, etc. The data mining data sets could be numbers, images or texts. Text mining is a variation on data mining. Text mining is the analysis of data contained in natural language texts. It refers to the process of deriving information form unstructured textual data. The difference between text mining and data mining is that in text mining the patterns are extracted from natural language text rather than from structured databases of facts. Text mining is also different from web search. In web search, the user is looking for something that already known or written by someone else. In text mining the goal is to discover unknown information.

Text mining is a multidisciplinary field related with statistics, information retrieval, linguistics, data mining and machine learning. First text mining approaches came forward in the middle of 1980's (Peterson & Seligman, 1985). Last decade, text mining approaches were developed rapidly with technological advances. The emergence of text analytics in its current form in the late 1990's, identified by Marti A. Hearst in the paper Untangling Text Data Mining (Hearst, 1999).

Text mining approaches can be used for many different purposes such as natural language processing, e-discovery, sentiment analysis, semantic analysis, text classification, social media analysis, marketing applications, etc. In this thesis semantic text mining approaches was examined. Semantic text mining approaches are used for text summarization, author recognition, literature investigation and

determining similarity of texts. There are many examples and studies of semantic analysis in English. However, morphological properties of the language make difficult to semantic analysis. Morphological structures of some languages are convenient to analysis such as English, German etc. Some other languages (Turkish, Chinese etc.) are not convenient to semantic analysis due to morphological difficulties.

Latent Semantic Analysis (LSA) is one of the most common semantic analysis algorithms. The first article about LSA was published in 1990 by Scott Deerwester and colleagues (Deerwester et al.,1990). LSA algorithm was used in also various languages other than English; Bulgarian (Nakov, 2011a), German (Nakov, 2011b), Russian (Nakov, 2011c). For Turkish, an article entitled "Text Summarization Using Latent Semantic Analysis" was published in 2011 (Ozsoy et al.,2011). Another common algorithm of semantic analysis is Probabilistic Latent Semantic Analysis (PLSA). It was suggested in 1999 (Hoffman, 1999a) and first article was published in 2001 (Hoffman, 2001). In 2002, Brants et al. studied about topic-based document segmentation with PLSA (Brants et al. 2002). PLSA algorithm was applied on web search in 2005 by Lin et al. In 2007, PLSA was used in bibliometric analysis for technology forecasting by Wang et al. Latent Dirichlet Allocation (LDA) is another algorithm about semantic analysis. The first article about LDA was published in 2003 by Blei et al. In 2010, three aforementioned algorithms and correlated topic model were compared with each other by Lee et al.(Lee at al. 2010).

In the literature, there are many studies about revealing the semantic similarities between texts in many languages. However, their use for Turkish is not frequent. On the other hand, the most common semantic analysis algorithms are used while number of topic is known. One of the goals of this research is to find suitable ways to determine similarities between texts in Turkish and divide texts into topics even if number of topics unknown. The remainder of this study is organized as follows. In Chapter two, text representation methods are investigated. Also, morphological structure and difficulties of Turkish language is explained and methods of avoiding morphological difficulties are stated. In Chapter three, semantic analysis methods are

examined. In Chapter four, performances of these methods are investigated for Turkish documents. In Chapter five, an iterative algorithm is proposed to find the semantic similarities of the texts when the number of topics is not known prior to the analysis. Chapter six includes the conclusions and commentaries for further researches.

CHAPTER TWO
TEXT REPRESENTATION AND DATA PREPERATION

Natural languages are hard to analyze due to morphological difficulties. Before applying semantic analysis, lexical components such as polysemy, synonymy, antonymy, hyponymy, idiom, should be dealt. Avoid of these lexical components, some methods can be used. In order to apply LSA, PLSA and LDA on texts, the words should be parsed into their roots and suffixes. After the stemming operation, the function words, which do not express meanings, but relation between meanings, should be identified.

2.1 Text Representation

In order to do text mining, the first question is how to represent texts. All words and characters must be represented as a numbers. There are some levels of text representations such as lexical, syntactic and semantic. Text representations methods are n-gram, words, phrases, part of speech tags and taxonomies, bag of words and language model etc. In this section n-gram, bag of words and term frequency methods are explained.

2.1.1 N-gram Method

N-gram is a sequence of terms with the length of N. N-gram is a contiguous sequence of n items from a given sequence of text. One word from the document is represented as the set of overlapping N-grams.

- bi-gram: -t, te, ex, t-
- tri-gram: -te, tex, ext, xt-, t--
- four-gram: -tex, text, ext-, xt--, t---

An n-gram of size 1 is referred to as a unigram, size 2 is a bigram, size 3 is a trigram, size 4 is four-gram and size five is five-gram. Whole document can be represented by n-gram.

Words occur in different frequencies in languages. There are many ways to calculate these frequencies. After calculating frequencies of words, texts can be represented by their n-gram frequency profiles, using the list of n-grams ordered by the number of occurrences in texts. Another way to represent text is term weights for each n-gram. Each text is represented as a vector with values of term weights for each n-gram. Each vector for each document may cause a high dimensionality problem. There are some ways to reduce dimension. Weights and frequency of the terms are calculated after dimension reduction.

2.1.2 Term Frequency Method

Term frequency method is one of the easiest ways to represent texts. Term frequency method deals with the frequency of the words in the texts. A matrix is formed which shows the numbers of the words in the texts. After stemming and stop-words removal process, texts in the data set are composed of roots. Term frequency method simply counts every root that how many times they appeared in texts. In order to yield better results for the analysis, the words which is occurred at least two or more times were included in the analysis. This helps to reduce the total number of terms and dimension. In this study, term frequency method was used in the applications.

2.1.3 Bag of Words Method

One of the most common ways for text representation is bag of words method. In bag of words method, text is represented as a bag of its words. Their grammar and word order is ignored. All words in the texts create dictionary and each text is represented by a vector. Every vector refers to count corresponding values in the

dictionary. If a word appears in text n times, its value will be n. If a word not appears in text, its value will be 0. That method is similar to term frequency method.

After the text vectors contain term frequencies, each word is represented as a separate variable having numeric weight. The most popular weighting schema is normalized word frequency TFIDF.

$$\text{Tfidf}(w)=tf.\ \log\left(\frac{N}{df(w)}\right) \tag{2.1}$$

tf(w):	term frequency
df(w):	document frequency
N:	number of all documents
TfIdf(w):	relative importance of the word in the document

2.2 Data Preparation

Before to deal with roots, suffixes and function words, morphology of language should be investigate. Stemming and stop-words removal process vary by the languages. In this thesis Turkish morphology was examined.

2.2.1 Turkish Morphology

Turkish is one of the morphologically complex languages. Affixes are added at the end of word and this might be cause to lose some letter in stem. Contrary to English, it is highly agglutinative with a rich morphology of suffixes. When the word is examined structurally, it is seen that it comprises of a root and a number of suffixes. The bare form of the word is the root, and the units following this bare form are the suffixes. These roots and suffixes are classified into different categories.

Example:

“bilgi – knowledge

bilgisiz – without knowledge

bilgisizlik – lack of knowledge

bilgisizlikleri – their lack of knowledge

bilgisizliklerinden – from their lack of knowledge

bilgisizliklerindenmis – I gather that it was from their lack of knowledge” (Lewis, 1989. pp. 3).

Another important feature of the Turkish language is vowel harmony. Vowel harmony is basically described as a “progressive sound assimilation” phenomenon. In simple words, the features of a vowel depend on the features of the preceding vowel. We’ll be dealing exclusively with the vowel harmony of suffixes in Turkish (Makedonski, 2005).

Table 2.1 Vowels in Turkish language

	Unrounded		Rounded	
	Low	High	Low	High
Front	a	ı	o	u
Back	e	i	ö	ü

All the vowels type in a word according to first vowel of that word.

“+Back

sekiz – eight

seksen – eighty

sinir – nerve

sinirler – nerves

sinirlerimiz – our nerves

-Back

dokuz – nine

doksan – ninety

sınır – frontier

sınırlar – frontiers

sınırlarımız – our frontiers” (Lewis, 1989. pp. 11).

In cases of the disharmony in the root or if an invariable suffix is attached, the harmonic suffixes harmonize with the vowel of the last preceding syllable. So attaching the plural suffix (-ler/-lar), which harmonizes for backness, to “anne” (mother) will result in “anneler” (mothers) and not in “annular”, harmonizing with the vowel of the first syllable (Makedonski, 2005).

There is also another special rule for vowels. Unrounded vowels are typically followed by unrounded vowels and rounded vowels are typically followed by low unrounded or high rounded vowels.

Combining the two principles we end up with the following:

- a is followed by a or ı
- e is followed by e or i
- ı is followed by a or ı
- i is followed by e or i
- is followed by a or u
- ö is followed by e or ü
- u is followed by a or u
- ü is followed by e or ü

Turkish suffixes harmonize with, for the sake of simplicity, the vowel of the last syllable of the word they are attached to. They could divide two different groups: The vowels of the first group alternate between the low unrounded vowels “a” and “e” (also called e-type2 suffixes) and the vowels of the second group alternate between the high vowels “ı”, “i”, “u” and “ü” (the so-called i-type1 suffixes). Except one – the present tense verbal suffix –ıyor/–iyor/–uyor/–üyor, no other suffixes contain “o” and “ö”. The plural suffix -ler/-lar falls in the first class, whereas suffix like the definite objective case suffix is an i-type suffix (Makedonski, 2005).

Table 2.2 Some Turkish suffixes

Root	Plural Suffix	Objective Case Suffix
ev (house)	evler (houses)	evi (the house)
kol (arm)	kollar (arms)	kolu (the arm)
kitap (book)	kitaplar (books)	kitabı (the book)
köprü (bridge)	köprüler (bridges)	köprüyü (the bridge)

2.2.2 Stemming

Stemming suffixes for words is useful in semantic analysis. Terms with a common stem will usually have similar meanings. This helps to determine semantic similarity. Another advantage of the stemming process is that it reduces the total number of terms and dimension.

Stemming algorithms vary by languages. In English, Porter Stemmer Algorithm works very well (Porter, 1980). The Porter Stemmer Algorithm has some conditions and rules. One of the conditions on the stem is the measure of a stem which is vowel-consonant sequences in stem. This measure is called as m.

$$[C](VC)^{m}[V] \tag{2.2}$$

C: Consonant
V: Vowel
m: The measure of a stem which is vowel-consonant sequences in stem.

- m=0 cry, bye
- m=1 care, scare
- m=2 troubles, oaten

Rule condition is based on the number of measure m. There are about sixty rules based on number of m.

- (m > 0) ational⟶ate relational⟶relate
- (m > 0) ization⟶ize realization⟶realize
- (m > 1) ance⟶ allowance⟶allow
- (m > 1) able⟶ adjustable⟶adjust

For stemming the Porter algorithm developed for English has a widespread use and has become a gold standard. There are stemming algorithms for Turkish, yielding successful results. In this thesis, for the stemming process in Turkish, Zemberek, an open source Turkish Natural Language Processing library was used.

2.2.3 Stop-Words Removal

Stop-words removal is one of the crucial processes of semantic analysis. Stop-words are words that are filtered out by the system. Any group of words can be chosen as the stop-words for a given purpose. But generally, short function words are chosen as stop-words. Short function words are natural language words which have very little meaning, such as “and”, “the”, “a”, “an” in English and “ama”, “ve”, “veya”, “ancak”, in Turkish. Stop-words removal process improves to semantic analysis performance. Removing meaningless words will help to reduce the total number of terms and dimension.

CHAPTER THREE
SEMANTIC TEXT MINING ALGORITHMS

Information retrieval and text mining have become prominent areas of research with the developing web technologies. The augmentation of the stored data increases the need for methods that would enable rapid and accurate access to desired data. Text mining can be defined as methods used for accessing the required information from among data in text format. Text mining makes use of research areas such as statistics, mathematics, natural language processing and information retrieval. It is used in automatic summarization, author identification and inter-textual similarity, etc. Methods such as, Latent Semantic Analysis (LSA), Probabilistic Latent Semantic Analysis (PLSA) and Latent Dirichlet Allocation (LDA) are the most prominent ones that can be used in categorizing texts with regard to topics, and determining their similarities.

3.1 Latent Semantic Analysis (LSA)

Latent Semantic Analysis (LSA) is a method for representation of the words by various statistical and mathematical computations. As the name suggests, it is used for finding covert meaning relations which could not be revealed by traditional text mining and natural language processing algorithms. LSA or Latent Semantic Indexing was developed in Bellcore laboratories in 1988. In 1989, Scott Deerwester and colleagues secured it by patent. In 1990, the first article, "Indexing by Latent Semantic Analysis" was published by Deerwester et al. (Deerwester et al., 1990). LSA takes each word from the original sources (text, paragraph, article, etc.) and represents it in a multidimensional "semantic space". In order to perform the analysis, a method called singular value decomposition (SVD), similar to factor analysis, is facilitated. For representing the texts as a matrix, the Term Frequency (TF), which is frequently used in information retrieval, can be utilized. TF deals with the frequencies of the words in the text, regardless of their meanings. Therefore, it is not a semantically efficient method. LSA enables conducting semantic analyses using SVD method via TF. LSA is not effective to solve all the semantic issues

alone; because it falls short against many issues such as the structure of language, morphological intricacies and the importance levels of the words. In order to solve these issues, methods such as root-affix parsing, stop words removal, synonyms – homonyms lists, quotations and weight diagrams are used before LSA.

LSA, nowadays, is used to calculate the similarity of the texts and in text summarization. In the literature, there are examples of LSA used in various languages. Especially, studies on German, English, Bulgarian and Russian are frequent (Nakov, 1999a, 1999b, 1999c). For Turkish, there are studies in which LSA is used for automatic summarization.

Standard LSA model is confronted with some problems in application. The greatest issue LSA has difficulty in solving is the "polysemy" concept. Polysemy is similar to the homonymy in Turkish. However, these two concepts are different. Homonymy denotes to words with the same pronunciation but different meanings; while polysemy refers to the connotation. Polysemy used for words which are used in different but related meanings. For instance, the word "foot" in the sentences "The house is at the foot of the mountains." and "One of his shoes felt too tight for his foot." means "the lower part of the mountain" in the first sentence and "the terminal part of leg" in the second (Using English Term: Polysemy, 2013). They are different with regard to meaning but also they are related to each other. The term "polysemy" is used for defining such words and relations. LSA is not sufficient in explaining polysemy and using it in analysis.

LSA is used for extraction semantic relations over raw text. It defines the texts as original character sequences and divides the texts into meaningful parts or samples comprising of the sentences in the texts. The first step is expressing the text as a matrix formed by the words in the text. A matrix is formed by expressing the predefined words in the rows, and the texts in the columns. Each cell of this matrix represents the frequency of the word in a particular text. Later, Singular Value Decomposition (SVD) is applied on this matrix. Using this method, the matrix is

reduced to a matrix with fewer dimensions and the relations between the words are revealed.

3.1.1 Singular Value Decomposition (SVD)

Singular Value Decomposition (SVD), is a form of factor analysis, or more properly the mathematical generalization of which factor analysis is a special case (Launderer et al., 1998). A rectangular matrix divided into three other matrices. The component U is eigenvector matrix derived from original row entities. The other component V is eigenvector matrix derived from original column entities. The last component S is eigenvalues matrix.

$$A_{mn} = U_{mm}\, S_{mn}\, V^{T}_{nn} \tag{3.1}$$

Assume that A is a matrix that to perform SVD theorem on it.

$$A = \begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix} \tag{3.2}$$

3.1.1.1 Deriving the Component U

Firstly A matrix multiply by A^T

$$A.A^T = \begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix} \begin{bmatrix} 3 & -1 \\ 1 & 3 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 11 & 1 \\ 1 & 11 \end{bmatrix} \tag{3.3}$$

Then calculate eigenvalues and eigenvectors from $A.A^T$.

$A\vec{v} = \lambda\vec{v}$

$$\begin{bmatrix} 11 & 1 \\ 1 & 11 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \lambda \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$11x_1+x_2=\lambda x_1 \longrightarrow (11-\lambda)x_1+x_2=0$

$x_1+11x_2=\lambda x_2 \longrightarrow x_1+(11-\lambda)x_2=0$

$$\begin{vmatrix} (11-\lambda) & 1 \\ 1 & (11-\lambda) \end{vmatrix} = 0 \longrightarrow \lambda=10,\ \lambda=12 \tag{3.4}$$

For λ=10;

$(11-10)x_1+x_2=0$

$x_1= 1, \quad x_2= -1$ (3.5)

For λ=10 eigenvector is $\begin{bmatrix}1 & -1\end{bmatrix}$.

For λ=12;

$(11-12)x_1+x_2=0$

$x_1=1, \; x_2=1$ (3.6)

For λ=12 eigenvector is $\begin{bmatrix}1 & 1\end{bmatrix}$.

These eigenvectors become column vectors in a matrix ordered by the size of the corresponding eigenvalue. In the matrix below, the eigenvector for λ = 12 is column one, and the eigenvector for λ = 10 is column two.

$$\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix} \quad (3.7)$$

Finally, the eigenvectors have to normalize with Gram-Schmidt orthonormalization method.

$$\vec{u}=\frac{\vec{v}}{|\vec{v}|}$$

$$\frac{[1 \quad 1]}{\sqrt{1^2+1^2}}=\begin{bmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\end{bmatrix}$$

$$\frac{[1 \quad -1]}{\sqrt{1^2+\left(-1^2\right)}}=\begin{bmatrix}\frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}}\end{bmatrix} \quad (3.8)$$

Normalized eigenvectors generate U component.

$$U=\begin{bmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}}\end{bmatrix} \quad (3.9)$$

3.1.1.2 Deriving the Component V

Deriving V is similar to U. V is based on $A^T.A$.

$$A^TA = \begin{bmatrix} 3 & -1 \\ 1 & 3 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 3 & 1 & 1 \\ -1 & 3 & 1 \end{bmatrix} = \begin{bmatrix} 10 & 0 & 2 \\ 0 & 10 & 4 \\ 2 & 4 & 2 \end{bmatrix} \quad (3.10)$$

Then calculate eigenvalues and eigenvectors from $A^T.A$.

$$\begin{bmatrix} 10 & 0 & 2 \\ 0 & 10 & 4 \\ 2 & 4 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \lambda \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

$10x_1+2x_3= \lambda x_1 \quad \longrightarrow \quad (10-\lambda)x_1+2x_3=0$

$10x_2+4x_3=\lambda x_2 \quad \longrightarrow \quad (10-\lambda)x_2+4x_3=0$

$2x_1+4x_2+2x_3=\lambda x_3 \quad \longrightarrow \quad 2x_1+4x_2+(2-\lambda)x_3=0$

$$\begin{vmatrix} (10-\lambda) & 0 & 2 \\ 0 & (10-\lambda) & 4 \\ 2 & 4 & (2-\lambda) \end{vmatrix} = 0 \longrightarrow \lambda = 0,\ \lambda = 10,\ \lambda = 12 \quad (3.11)$$

For $\lambda=12$, eigenvector is $[1 \quad 2 \quad 1]$, for $\lambda=10$, eigenvector is $[2 \quad -1 \quad 0]$ and for $\lambda=0$, eigenvector is $[1 \quad 2 \quad -5]$.

Eigenvectors become column vectors in a matrix ordered by the size of the corresponding eigenvalue and normalize with Gram-Schmidt orthonormalization method.

$$V = \begin{bmatrix} \frac{1}{\sqrt{6}} & \frac{2}{\sqrt{5}} & \frac{1}{\sqrt{30}} \\ \frac{2}{\sqrt{6}} & \frac{-1}{\sqrt{5}} & \frac{2}{\sqrt{30}} \\ \frac{1}{\sqrt{6}} & 0 & \frac{-5}{\sqrt{30}} \end{bmatrix}, \quad V^T = \begin{bmatrix} \frac{1}{\sqrt{6}} & \frac{2}{\sqrt{6}} & \frac{1}{\sqrt{6}} \\ \frac{2}{\sqrt{5}} & \frac{-1}{\sqrt{5}} & 0 \\ \frac{1}{\sqrt{30}} & \frac{2}{\sqrt{30}} & \frac{-5}{\sqrt{30}} \end{bmatrix} \quad (3.12)$$

3.1.1.3 Deriving the Component S

For derive S, the square roots of the non-zero eigenvalues are put the largest one in S_{11}, the next largest in S_{22} and so on until the smallest one ends up in S_{mm}.

$$S_{mn}=\begin{bmatrix}\sqrt{12} & 0 & 0\\ 0 & \sqrt{10} & 0\end{bmatrix} \tag{3.13}$$

Multiplication of these three matrices equal to A matrix.

$$A_{mn}=\begin{bmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}}\end{bmatrix}\begin{bmatrix}\sqrt{12} & 0 & 0\\ 0 & \sqrt{10} & 0\end{bmatrix}\begin{bmatrix}\frac{1}{\sqrt{6}} & \frac{2}{\sqrt{6}} & \frac{1}{\sqrt{6}}\\ \frac{2}{\sqrt{5}} & \frac{-1}{\sqrt{5}} & 0\\ \frac{1}{\sqrt{30}} & \frac{2}{\sqrt{30}} & \frac{-5}{\sqrt{30}}\end{bmatrix}=\begin{bmatrix}3 & 1 & 1\\ -1 & 3 & 1\end{bmatrix} \tag{3.14}$$

3.1.2 A LSA Example for English

This chapter includes an example of an LSA from Launder et al. for English texts (Launder et al., 1998). Five texts about computer – human relations, and four about mathematical graphic theory were selected for analysis. The original matrix was comprised of nine columns. 12 words were determined for conducting the analysis and therefore the matrix also comprises of 12 rows. The words which were used at least two times in the texts are selected for analysis. Texts represented with "c" in Table 3.1 are the texts about computer – human relations and the texts represented with "m" are the texts about mathematical graphic theory.

Table 3.1 Texts used for LSA example

c1: *Human* machine *interface* for ABC *computer* applications
c2: A *survey* of *user* opinion of *computer system response time*
c3: The *EPS user interface* management *system*
c4: *System* and *human system* engineering testing of *EPS*
c5: Relation of *user* perceived *response time* to error measurement
m1: The generation of random, binary, ordered *trees*
m2: The intersection *graph* of paths in *trees*
m3: *Graph minors* IV: Widths of *trees* and well-quasi-ordering
m4: *Graph minors*: A *survey*

After obtaining the text-word matrix, the SVD phase is applied. Accordingly, the original matrix is separated into three matrices. Then the dimension reduction operation is performed and the new matrix is obtained.

Table 3.2 Text-Word table

	c1	c2	c3	c4	c5	m1	m2	m3	m4
human	1	0	0	1	0	0	0	0	0
interface	1	0	1	0	0	0	0	0	0
user	0	1	1	0	1	0	0	0	0
system	0	1	1	2	0	0	0	0	0
response	0	1	0	0	0	1	0	0	0
time	0	1	0	0	0	1	0	0	0
eps	0	0	1	1	0	0	0	0	0
survey	0	1	0	0	0	0	0	0	1
trees	0	0	0	0	0	1	1	1	0
graph	0	0	0	0	0	0	1	1	0
minors	0	0	0	0	0	0	0	1	1

$$U=\begin{pmatrix}
-0.22135 & -0.1131 & 0.2889 & -0.41475 & -0.10628 & -0.34098 & -0.52266 & 0.06045 & 0.406678 & -0.07586 & 0.322668 & -0.05508 \\
-0.19765 & -0.0720 & 0.13504 & -0.55224 & 0.281769 & 0.495878 & 0.070423 & 0.00994 & 0.10893 & -0.03389 & -0.4564 & -0.28438 \\
-0.24047 & 0.0431 & -0.1644 & -0.59496 & -0.10676 & -0.25496 & 0.30224 & -0.06233 & -0.49244 & 0.109757 & 0.133733 & 0.339456 \\
-0.4036 & 0.05707 & -0.3378 & 0.099114 & 0.331734 & 0.384832 & -0.00287 & 0.000391 & -0.01233 & -0.15173 & 0.645336 & -0.11015 \\
-0.64448 & -0.1673 & 0.3611 & 0.333462 & -0.15895 & -0.20652 & 0.165829 & -0.03427 & -0.2707 & -0.10976 & -0.13373 & -0.33946 \\
-0.26504 & 0.1071 & -0.426 & 0.073812 & 0.080319 & -0.16968 & -0.28292 & 0.016147 & 0.053875 & -0.57499 & -0.4324 & 0.308745 \\
-0.26504 & 0.1071 & -0.426 & 0.073812 & 0.080319 & -0.16968 & -0.28292 & 0.016147 & 0.053875 & 0.726719 & -0.21294 & -0.1986 \\
-0.30083 & -0.1412 & 0.3303 & 0.188092 & 0.114785 & 0.272155 & -0.03299 & 0.018998 & 0.165339 & 0.295378 & -0.0552 & 0.733987 \\
-0.20592 & 0.2736 & -0.1776 & -0.03235 & -0.53715 & 0.080944 & 0.466898 & 0.036299 & 0.579426 & 3.20E-16 & 2.50E-16 & -6.94E-18 \\
-0.01275 & 0.490 & 0.2312 & 0.024802 & 0.59417 & -0.39213 & 0.288317 & -0.25457 & 0.225424 & 1.28E-16 & -1.39E-17 & -1.04E-16 \\
-0.03614 & 0.622 & 0.2236 & 0.0007 & -0.06825 & 0.114909 & -0.15958 & 0.681125 & -0.23196 & -1.42E-16 & -6.85E-17 & 5.55E-17 \\
-0.03176 & 0.450 & 0.1411 & -0.00873 & -0.3005 & 0.277343 & -0.3395 & -0.67842 & -0.18253 & -1.60E-16 & -3.39E-16 & 2.78E-16
\end{pmatrix}$$

$$V=\begin{pmatrix}
-0.19739 & -0.05591 & 0.11027 & -0.94979 & 0.045679 & -0.07659 & -0.17732 & 0.014393 & 0.063692 \\
-0.60599 & 0.165593 & -0.49733 & -0.02865 & -0.20633 & -0.25648 & 0.432984 & -0.04931 & -0.24278 \\
-0.46292 & -0.12731 & 0.207606 & 0.041609 & 0.378336 & 0.7244 & 0.23689 & -0.00883 & -0.02408 \\
-0.54211 & -0.23176 & 0.569921 & 0.267714 & -0.2056 & -0.36886 & -0.2648 & 0.019467 & 0.084207 \\
-0.27947 & 0.106775 & -0.50545 & 0.150035 & 0.327194 & 0.034813 & -0.6723 & 0.05835 & 0.262376 \\
-0.00382 & 0.192848 & 0.098184 & 0.015081 & 0.394841 & -0.30016 & 0.34084 & -0.45448 & 0.619847 \\
-0.01463 & 0.437875 & 0.192956 & 0.015507 & 0.349485 & -0.2122 & 0.152195 & 0.761527 & -0.01798 \\
-0.02414 & 0.615122 & 0.252904 & 0.010199 & 0.149798 & 9.74E-05 & -0.24915 & -0.44964 & -0.51989 \\
-0.08196 & 0.529937 & 0.079273 & -0.02455 & -0.60199 & 0.362219 & -0.03803 & 0.069638 & 0.453507
\end{pmatrix}$$

$$S=\begin{pmatrix}
3.340884 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 2.541701 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 2.353944 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1.644532 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1.504832 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1.306382 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0.845903 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.560134 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.363677 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}$$

Although there is not any particular rule for dimension reduction, more successful results are obtained when reduction is done with reference to the number of the topics. Therefore, the matrices are reduced into two dimensions. The new matrix is obtained by multiplying these three reduced matrices. Later the correlation matrix is calculated to investigate the relation between the texts.

$$\text{Reduced U} = \begin{pmatrix} -0.22135 & -0.11318 \\ -0.19765 & -0.07209 \\ -0.24047 & 0.043152 \\ -0.4036 & 0.05707 \\ -0.64448 & -0.1673 \\ -0.26504 & 0.10716 \\ -0.26504 & 0.10716 \\ -0.30083 & -0.14127 \\ -0.20592 & 0.273647 \\ -0.01275 & 0.490162 \\ -0.03614 & 0.622785 \\ -0.03176 & 0.450509 \end{pmatrix} \quad \text{Reduced V} = \begin{pmatrix} -0.19739 & -0.05591 \\ -0.60599 & 0.165593 \\ -0.46292 & -0.12731 \\ -0.54211 & -0.23176 \\ -0.27947 & 0.106775 \\ -0.00382 & 0.192848 \\ -0.01463 & 0.437875 \\ -0.02414 & 0.615122 \\ -0.08196 & 0.529937 \end{pmatrix}$$

$$\text{Reduced S} = \begin{pmatrix} 3{,}340884 & 0 \\ 0 & 2{,}541701 \end{pmatrix}$$

$$\text{The new A} = \begin{pmatrix} 0.1620 & 0.4004 & 0.3789 & 0.4675 & 0.1759 & -0.0526 & -0.1151 & -0.1591 & -0.091 \\ 0.1405 & 0.3698 & 0.3289 & 0.4004 & 0.1649 & -0.0322 & -0.0705 & -0.0967 & -0.0429 \\ 0.1524 & 0.5050 & 0.3579 & 0.4101 & 0.2362 & 0.0247 & 0.05971 & 0.0868 & 0.1236 \\ 0.2580 & 0.8411 & 0.605 & 0.6973 & 0.3923 & 0.0331 & 0.08324 & 0.1212 & 0.1873 \\ 0.4487 & 1.2343 & 1.0508 & 1.2657 & 0.5563 & -0.073 & -0.1546 & -0.2096 & -0.0488 \\ 0.1595 & 0.5816 & 0.3752 & 0.4168 & 0.2765 & 0.0554 & 0.13221 & 0.1889 & 0.21690 \\ 0.1595 & 0.5816 & 0.3752 & 0.4168 & 0.2765 & 0.0559 & 0.13221 & 0.1889 & 0.21690 \\ 0.2184 & 0.5495 & 0.5109 & 0.6280 & 0.2425 & -6.54E-02 & -0.1425 & -0.1966 & -0.1079 \\ 0.0969 & 0.5320 & 0.2294 & 0.2114 & 0.2665 & 0.1367 & 0.3146 & 0.4441 & 0.4249 \\ -0.061 & 0.2321 & -0.1389 & -0.2656 & 0.1449 & 0.2404 & 0.5461 & 0.7673 & 0.6637 \\ -0.064 & 0.335 & -0.1456 & -0.3014 & 0.2027 & 0.3057 & 0.6948 & 0.9766 & 0.8487 \\ -0.0430 & 0.2539 & -0.0966 & -0.2078 & 0.15191 & 0.2212 & 0.5029 & 0.7069 & 0.6155 \end{pmatrix}$$

In order to observe the relations between the texts the correlation matrix is calculated. As it is seen Table 3.3, the topics which expressed similar concepts such as human and user and which are semantically close have a higher correlation. Similarly, the correlation of matrices which do not have any semantic relation or which have little relation is found relatively low.

Table 3.3 The Correlation Matrix of Reduced A Matrix

	c1	**c2**	**c3**	**c4**	**c5**	**m1**	**m2**	**m3**	**m4**
c1	1	0.91048	0.99999	0.99823	0.84220	-0.85752	-0.85346	-0.85169	-0.81119
c2	0.91048	1	0.91205	0.88430	0.98978	-0.56801	-0.56154	-0.55873	-0.49674
c3	0.99999	0.91205	1	0.99800	0.84424	-0.85555	-0.85147	-0.84969	-0.80896
c4	0.99823	0.88430	0.99800	1	0.80867	-0.88657	-0.88292	-0.88132	-0.84450
c5	0.84220	0.98978	0.84424	0.80867	1	-0.44482	-0.43779	-0.43474	-0.36789
m1	-0.85752	-0.56801	-0.85555	-0.88657	-0.44482	1	0.99997	0.99994	0.99645
m2	-0.85346	-0.56154	-0.85147	-0.88292	-0.43779	0.99997	1	0.99999	0.99708
m3	-0.85169	-0.55873	-0.84969	-0.88132	-0.43474	0.99994	0.99999	1	0.99734
m4	-0.81119	-0.49674	-0.80896	-0.84450	-0.36789	0.99645	0.99708	0.99734	1

3.2 Probabilistic Latent Semantic Analysis (PLSA)

Probabilistic Latent Semantic (PLSA) method was devised by Hofmann in 1999. PLSA is a statistical method used in revealing the latent semantic values (Hoffman, 1999a). It tries to determine at what probability texts and words belong to the topics. It can be expressed as a probabilistic interpretation of the LSA method. In LSA method, the matrix representing the texts is extricated into sub-matrices and these extricated matrices, after dimension reduction, are used in the calculation of the fewer-dimension matrix. With the calculations (correlation coefficient, etc.) performed on the matrix, the similarity between texts can be interpreted.

In PLSA method, random prior probabilities pertaining to texts, topics and words are defined. Via these prior probabilities, an intermediate value is calculated and the probabilities are updated by maximizing this value. This stage is repeated until the difference of the pre-maximization and post-maximization probabilities reaches the desired value or the calculation reaches the determined number of iterations. The similarity between the texts can be determined with reference to the probabilities obtained. PLSA and LSA methods can be used in cases where the number of topics is known exactly.

In order to use the PLSA method, the texts are required to be represented in matrix form and the number of the texts to be known. The definitions of the probabilities used in PLSA are as below:

P(di|zj) : the probability of ith text to belong to jth topic i:1,…,d j: 1,…,k

P(zj) : the probability of jth topic (latent class) j: 1,…,k

P(wi|zj) : the probability of nth word to belong to jth topic i: 1,…,w j: 1,…,k

Before commencing the analysis, the text, the latent class and the word probabilities are determined. The values for P(z) and P(w|z) are assigned randomly. For P(d|z), equal probabilities are assigned with regard to the number of the texts.

PLSA uses Expectation Maximization (EM) algorithm. EM algorithm is an iterative method which includes the maximum likelihood estimation for estimating parameters (Dempster et al., 1977). EM starts from the likelihood function formed with reference to the observed data, and estimates the parameters by maximizing these likelihood functions via iteration. EM algorithm is comprised of two phases; expectation (E) phase and maximization (M) phase. In the Expectation phase, the probabilities are calculated for latent variables with regard to the parameters. In Maximization phase, the parameters are updated according to the probabilities determined in the E phase. This procedure is repeated until the difference between the P(z) probabilities, determined in the E and M phases, reached at the desired value (ε) or the desired iteration number is reached.

Expectation phase;

$$P(z|d,w)=\frac{P(z)\times P(d|z)\times P(w|z)}{\sum_i P(z_i)\times P(d|z_i)\times P(w|z_i)} \quad (3.15)$$

Separate calculations are made for each latent class in the expectation phase. At the end of the procedure, matrices of each latent classes (P(z|d,w)) are formed. P(z|d,w) probabilities are used in the Maximization phase. Three separate probabilities are calculated in Maximization phase. These are;

$$P(w|z)=\frac{\sum_j n(d_j,w)\times P(z|d_j,w)}{\sum_j \sum_i n(d_j,w_i)\times P(z|d_j,w_i)} \quad (3.16)$$

$$P(d|z)=\frac{\sum_i n(d,w_i)\times P(z|d,w_i)}{\sum_j \sum_i n(d_j,w_i)\times P(z|d_j,w_i)} \quad (3.17)$$

$$P(z)=\frac{\sum_j \sum_i n(d_j,w_i)\times P(z|d_j,w_i)}{\sum_k \sum_j \sum_i n(d_j,w_i)\times P(z_k|d_j,w_i)} \quad (3.18)$$

In Maximization phase, P(z), P(w|z) and P(d|z) probabilities are calculated again, using the P(z|d,w) probabilities calculated in the expectation phase. The newly calculated P(z), P(w|z) and P(d|z) probabilities are compared to the P(z), P(w|z) and P(d|z) probabilities before expectation phase. If the expected ε value is not reached, the EM phases are repeated until ε value is reached or the determined iteration number is reached.

3.2.1 A PLSA Application for English

In this section, a PLSA example was conducted on English texts. For this purpose, the data set in section 2.1.2 was used. This data comprised of two topics and nine texts. Before starting the analysis, the word, text and latent class probabilities were assigned. Accordingly, latent class probabilities P(z) and word probabilities P(w|z) were randomly determined, text probabilities P(d|z) were determined as equal as 1/9, with regard to the number of texts. ε value was determined as 0.0000000001. Before starting the analysis text, latent class and word probabilities were generated randomly.

Table 3.4 Word probabilities P(w|z) randomly generated before the analysis

	human	interface	user	system	response	time	eps	survey	trees	graph	minors
Topic 1	0.8147	0.1270	0.6324	0.2785	0.9575	0.1576	0.9572	0.8003	0.4218	0.7922	0.6557
Topic 2	0.9058	0.9134	0.0975	0.5469	0.9649	0.9706	0.4854	0.1419	0.9157	0.9595	0.0357

Table 3.5 Text probabilities P(d|z) randomly generated before the analysis

	C1	C2	C3	C4	C5	M1	M2	M3	M4
Topic 1	0.1111	0.1111	0.1111	0.1111	0.1111	0.1111	0.1111	0.1111	0.1111
Topic 2	0.1111	0.1111	0.1111	0.1111	0.1111	0.1111	0.1111	0.1111	0.1111

Table 3.6 Latent class probabilities P(z) randomly generated before the analysis

Topic 1	Topic 2
0.8491	0.9340

The ε value could not be reached after the first iteration. and the procedure was repeated.

Table 3.7 Word probabilities P(w|z) after the first iteration

	human	interface	user	system	response	time	eps	survey	trees	graph	minors
Topic 1	0.0706	0.0176	0.2012	0.0993	0.0744	0.0202	0.1007	0.1313	0.0695	0.0673	0.1480
Topic 2	0.0830	0.1340	0.0328	0.2063	0.0793	0.1315	0.0540	0.0246	0.1596	0.0862	0.0085

Table 3.8 Text probabilities P(d|z) after the first iteration

	C1	C2	C3	C4	C5	M1	M2	M3	M4
Topic 1	0.044091	0.204826	0.151046	0.135293	0.067064	0.070447	0.056786	0.130796	0.139651
Topic 2	0.108507	0.180266	0.156539	0.171695	0.010946	0.158614	0.096295	0.100559	0.016579

Table 3.9 Latent Semantic probabilities P(z) after the first iteration

Topic 1	Topic 2
0.490313	0.509687

The EM phases were repeated until the ε value was reached. After 15 iterations the desired ε value was reached.

Table 3.10 Text probabilities P(d|z) for English

	C1	C2	C3	C4	C5	M1	M2	M3	M4
Topic 1	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.2222	0.3333	0.1111
Topic 2	0.1176	0.2941	0.2353	0.2353	0.0588	0.0000	0.0000	0.0000	0.0088

The results of the analysis showed that the texts were separated into two with reference to topic. The first five texts belonged to the first topic and the other four belonged to the second.

3.3 Latent Dirichlet Allocation (LDA)

Latent Dirichlet Allocation (LDA) was suggested in 2003 by Blei et.al. (Blei et al., 2003). The main idea behind LDA is that the documents are represented as a

random mixture of hidden topics. LDA assumes that the characteristics of topics and documents are drawn from Dirichlet distributions. LDA tries to determine the belonging probabilities of the words to the topics using Dirichlet distributions. It can be used to categorize texts in terms of topics according to the words they contain. LDA can be used for classifying documents, summarizing and calculating similarities. Three criteria are taken into consideration while implementing LDA:

1) If the topics, in which a document belongs, similar words are encountered in other texts belonging to that topic.
2) If there is not any information about the topics the texts belong to, the probability for the text to belong topics are equal.
3) If there is not any information about which words are encountered in any topic, the words are encountered in the topics with the same probability.

3.3.1 Notation and Terminology

Let k be a "corpus" comprising of M documents mentioning different topics. Let each document comprise of words. Accordingly;

V: Number of unique words in corpus
k: Number of topics
M: Number of documents
N: Number of words per document
D: A document is a sequence of N words. $D = (w_1,w_2,w_3,\ldots,w_N)$
C: A corpus is a sequence of M documents. $C = (D_1,D_2,D_3,\ldots,D_M)$

V words are selected from each text. The group comprising of these words are called "vocabulary". The number of words in the texts is represented as N. These words are expressed as $D = (w_1,w_2,w_3,\ldots,w_N)$. w_i represents the i^{th} word in the text. There are M documents in the data set and are expressed as $C = (D_1,D_2,D_3,\ldots,D_M)$

3.3.2 LDA Steps

1. Choose N ~ Poisson (ξ)
2. Choose θ ~ Dirichlet (α)
3. For each of the N words, generate w_i:
   a. Choose a topic z_n ~ Multinomial (θ)
   b. Choose a word w_i $p(w_i \mid z_n, \beta)$ (Blei et al., 2003)

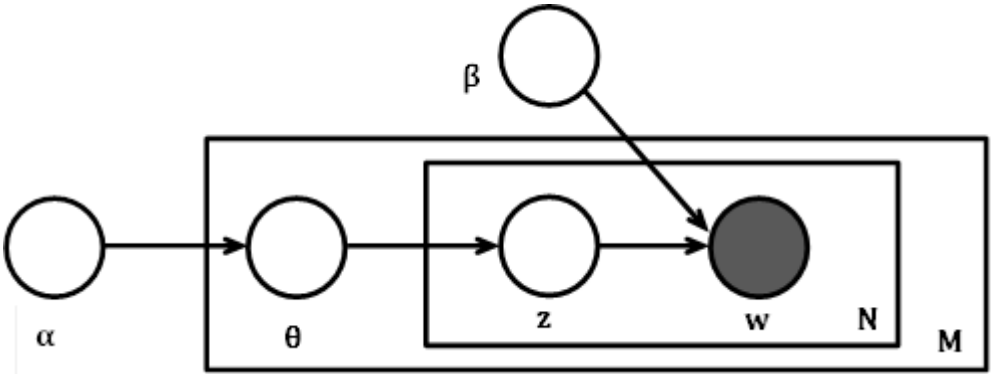


Figure 3.1 Graphical model representation of LDA (Blei et al., 2003)

Parameter α is the parameter of the Dirichlet prior on the per-document topic distributions. α is k dimensional vector. α_i represents a global weight for a certain topic i. Parameter θ is a Dirichlet k-dimensional random variable. θ_d determines the topic mixture per document d. Each parameter θ_d is sampled according to α_i.

The probability density function of the k dimensional Dirichlet random variable is:

$$p(\theta|\alpha)=\frac{\Gamma(\sum_{i=1}^{k}\alpha_i)}{\prod_{i=1}^{k}\Gamma(\alpha_i)}\theta_1^{\alpha_1-1}\ldots\theta_k^{\alpha_k-1} \quad (3.19)$$

Variable z represents topics. z_{di} is the topic for the i^{th} word in document d. Multinomial random variable chosen with θ_d as its parameter. Parameter β is defined as k × V matrix. Determines what the probability of a vocabulary word is given in a topic.

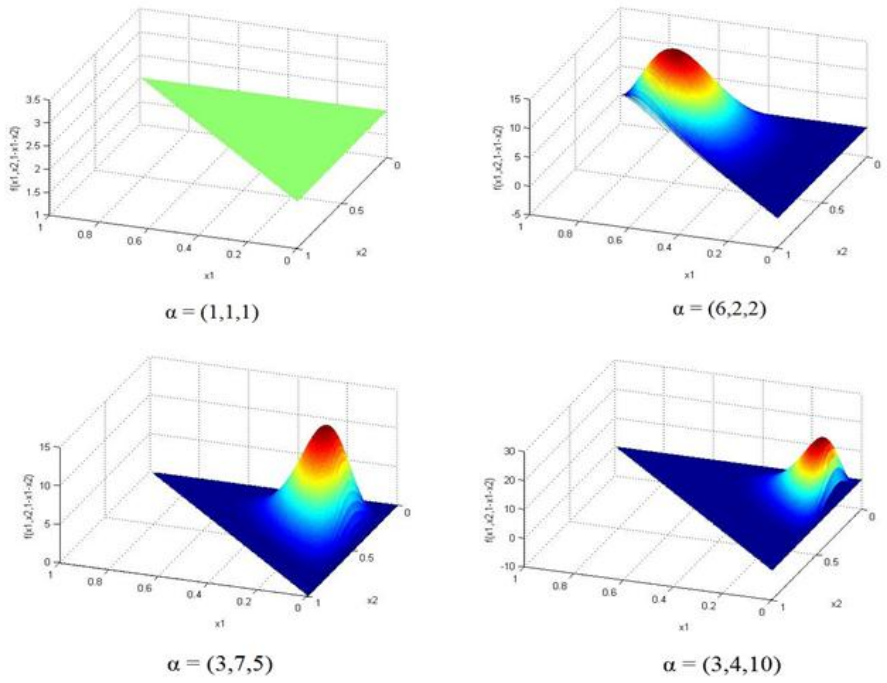


Figure 3.2 Dirichlet density plots, k = 3

$$\beta_{i,j}=p\left(w^j=1\middle|z^i=1\right) \tag{3.20}$$

Given the parameters α and β, the joint distribution of a topic mixture θ, a set of N topics z, and a set of N words D is given by:

$$p(\theta,z,D|\alpha,\beta)=p(\theta|\alpha)\prod_{i=1}^{N}p(z_i|\theta)p(w_i|z_i,\beta) \tag{3.21}$$

For obtain marginal distribution of a document, integrate over θ and sum over z.

$$p(D|\alpha,\beta)=\int p(\theta|\alpha)\left(\prod_{i=1}^{N}\sum_{z_i}p(z_i|\theta)p(w_i|z_i,\beta)\right)d\theta \tag{3.22}$$

To calculate probability of a corpus (Blei et al., 2003):

$$\begin{aligned}p(C|\alpha,\beta) &= \prod_{d=1}^{M}p(D_d|\alpha,\beta)\\ &=\prod_{d=1}^{M}\int p(\theta_d|\alpha)\left(\prod_{i=1}^{N_d}\sum_{z_{dn}}p(z_{di}|\theta_d)p(w_{di}|z_{di},\beta)\right)d\theta_d\end{aligned} \tag{3.23}$$

3.4 LSA after PLSA

All semantic analysis methods such as LSA, PLSA and LDA work while number of topics is known. But in fact number of topics may not be known. For analyze the texts while number of topics unknown semantically, an iterative algorithm which uses PLSA and LSA jointly is proposed.

The probabilities which calculated after PLSA help to determine number of topic. Three different probabilities are calculated after PLSA;

1) The probabilities of texts to belong latent classes P(d|z),
2) Latent class (topic) probabilities P(z),
3) The probabilities of words to belong topics P(w|z).

In order for text probabilities P(d|z) and latent class probabilities P(z) to be consistent the number of latent classes (number of topics) should be known previously and the analysis should be conducted accordingly. When the number of the topics is known prior to the analysis, the method is expected to assign only one probability to each text. This is the desired P(d|z) matrix. The latent class probabilities which are determined randomly at the beginning are expected to be found as 1/k at the end of the analysis. However, when the number of topics is not known previously PLSA and LSA methods cannot be used.

If PLSA method perform with true number of topics, every documents has only one probabilities in P(d|z) document probabilities matrix. Based on these feature, true number of topics could be determined. If analysis is repeated sufficiently for every possible number of topics, true number of topics could be found.

In that case there are some difficulties. If system start to analysis lower number of correct latent class number P(d|z) document probabilities matrices look like desired matrices. On the other hand, due to random number phase in Probabilistic Latent Semantic Analysis result does not converge to the desired matrices.

An iterative algorithm is proposed in which LSA and PLSA methods are jointly used for determining the similarities of text when the number of topics is not known previously. The steps of this algorithm are as below;

- PLSA method is tried for r times for the number of topic from 1 to the number of texts (k) and the P(d|z) results are obtained.
- As a result of the analysis, the results giving the desired P(d|z) matrix for each topic are counted.
- The successive differences of the result numbers yielding desired the P(d|z) matrix for each topic number are taken and the number of topics yielding the maximum difference are determined.
- The smaller one of the successive number of topics yielding the maximum difference is determined as the number of topics.
- The determined number of topics is used in LSA analysis and the similarity measurements are performed.

The application of LSA after PLSA algorithm is explained in Chapter four.

CHAPTER FOUR
APPLICATIONS

This chapter investigated the LSA, PLSA and LDA methods and their performances on Turkish texts. Numerous studies can be found about revealing the semantic similarities between English texts using these methods. However, their use for Turkish is not frequent. The reason for this is the structural difficulties of Turkish. The performance of LSA, PLSA and LDA is depending on true number of topics. In this chapter, performances of LSA, PLSA and LDA are investigated while number of topics is known. Also, performance of the iterative algorithm which uses LSA and PLSA jointly is investigated while number of topic is unknown.

Different data sets were used for the applications. Some data sets were composed from news articles. These articles were taken from news agencies. News articles were taken from their agencies according to determined topics. On the other hand, due to morphological difficulties of Turkish languages, collecting Turkish texts and representing as a term frequency matrix is a very time consuming process. There is not any automated method for stemming, stop words removal and text representing. All these processes are completed manual. Artificial data sets were generated for extended applications. All these data sets were used on LSA, PLSA, LDA and LSA after PLSA algorithms. MATLAB and R statistical programming language were used for the implementation of these algorithms. The codes used in the study were given in Appendices.

4.1 Artificial Data Sets

Due to morphological structure of Turkish, collecting the data from natural language texts is really difficult. Especially stemming process takes a long time. For extend to applications, some word-document matrices were generated with different conditions. In first condition all texts belong to different topics and in second condition all texts belong to same topic.

Table 4.1 Data set-1:Generated word-document table while all documents belong to different topics

	Text 1	Text 2	Text 3	Text 4	Text 5
Word 1	1	0	0	0	0
Word 2	2	0	0	0	0
Word 3	1	0	0	0	0
Word 4	1	0	0	0	0
Word 5	3	0	0	0	0
Word 6	0	1	0	0	0
Word 7	0	2	0	0	0
Word 8	0	3	0	0	0
Word 9	0	1	1	0	0
Word 10	0	0	1	0	0
Word 11	0	0	3	0	0
Word 12	0	0	2	0	0
Word 13	0	0	0	1	0
Word 14	0	0	0	2	0
Word 15	0	0	0	3	0
Word 16	0	0	0	2	0
Word 17	0	0	0	0	3
Word 18	0	0	0	0	1
Word 19	0	0	0	0	3
Word 20	0	0	0	0	2

Table 4.2 Data set-2: Generated word-document table while all documents belong to same topic

	Text 1	Text 2	Text 3	Text 4	Text 5
Word 1	1	2	2	3	1
Word 2	2	3	3	1	2
Word 3	1	2	1	3	1
Word 4	2	2	1	2	1
Word 5	3	2	1	1	2
Word 6	2	1	3	2	1
Word 7	2	1	4	1	3
Word 8	2	1	1	2	2
Word 9	1	1	2	1	1
Word 10	1	3	1	1	3

4.1.1 LSA Applications with Artificial Data Sets

Firstly LSA was applied on data set-1.

Table 4.3 Correlation table obtained as a result of the LSA on data set-1 (different topics)

	Text 1	Text 2	Text 3	Text 4	Text 5
Text 1	1	-0.220	-0.220	-0.232	-0.231
Text 2	-0.220	1	-0.115	-0.205	-0.204
Text 3	-0.220	-0.115	1	-0.205	-0.204
Text 4	-0.232	-0.205	-0.205	1	-0.214
Text 5	-0.231	-0.204	-0.204	-0.214	1

According the Table 4.3 LSA method did not calculate any similarities between texts. These results yielded accurate result while all texts from different topics.

Secondly, LSA applied on data set-2.

Table 4.4 Correlation table obtained as a result of the LSA on data set-2 (same topics)

	Text 1	Text 2	Text 3	Text 4	Text 5
Text 1	1	1	1	1	1
Text 2	1	1	1	1	1
Text 3	1	1	1	1	1
Text 4	1	1	1	1	1
Text 5	1	1	1	1	1

According to Table 4.4, LSA results yielded successful results. All texts belong same topics.

4.1.2 PLSA Applications with Artificial Data Sets

The results of PLSA on data set-1 with different topics are shown at Table 4.5. The analysis stopped at 14^{th} iteration.

Table 4.5 PLSA text probabilities P(d|z) for data set-1 (different topics)

	Text 1	Text 2	Text 3	Text 4	Text 5
Topic 1	0	0,999	0	0	0
Topic 2	0	0	0	0,999	0
Topic 3	1	0	0	0	0
Topic 4	0	0	0,999	0	0
Topic 5	0	0	0	0	1

According to Table 4.5, the texts were classified into five topics. Each text has only one possibility to belong to only one topic. Each topic has only one text.

PLSA was used on generated data set-2 while all texts belong to same topic.

Table 4.6 PLSA text probabilities P(d|z) for data set-2 (same topics)

	Text 1	Text 2	Text 3	Text 4	Text 5
Topic 1	0,193	0,204	0,215	0,193	0,193

In Table 4.6 all texts have only one probability. This result indicates that all texts related with same topic.

4.1.3 LDA Applications with Artificial Data Sets

LDA method was applied using the functions in the "topicmodels" package in R programming language (Grün & Hornik, 2011). The result of LDA on data set-1 (different topics) is shown in Table 4.7.

Table 4.7 The results of LDA application on data set-1 (different topics)

	Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
Text 1	0.0008	0.0008	0.0008	0.9966	0.0008
Text 2	0.0009	0.0009	0.0009	0.0009	0.9961
Text 3	0.9961	0.0009	0.0009	0.0009	0.0009
Text 4	0.0008	0.9966	0.0008	0.0008	0.0008
Text 5	0.0007	0.0007	0.9970	0.0007	0.0007

According to results in Table 4.7, LDA yielded accurate results on data set-1. Due to restrictions of "topicmodels" package number of topics must be greater than 1. So LDA could not be used on generated data set-2.

4.2 Real Data Sets

In this section two real data sets were used. In order to find semantically similar texts, news articles from different news agency are collected. The news texts about the same topic were collected from various news agencies. In first real data set, news texts about four different topics were obtained from three different news agencies (DHA: Dogan News Agency; IHA: Ihlas News Agency; AA: Anadolu Agency) and a total of 12 texts were gathered.

Table 4.8 Real dataset-1: News texts with four topics

	DHA	IHA	AA
News Text 1	Güvenlik kulübeleri yakıp çöp konteynerlerinden barikat yapan grup, polise taş, şişe ve sopa atarken, güvenlik güçleri biber gazı, gaz bombası ve tazyikli suyla karşılık verdi. Müdahale sırasında çok sayıda öğrenci ve basın mensubu gazdan etkilendi.	Polis ve güvenlik güçleri grupla yaklaşık bir saat çatıştı. Bölücü örgüt yandaşları, daha sonra üniversite girişindeki nöbetçi kulübesini ve kurdukları barikatı ateşe verdi. Çıkan yangın itfaiye ekipleri tarafından söndürüldü. Grup daha sonra kampüsün içine çekildi.	Polis, taş, şişe ve sopa atan gruba, biber gazı, gaz bombası ve tazyikli suyla müdahale etti. Müdahale sırasında çok sayıda öğrenci atılan gazdan etkilendi. Müdahalenin ardından kampüse çekilen gruptakiler, yerleşkedeki özel güvenlik görevlilerine ait kulübeleri ateşe verdi, güvenlik kameralarının bulunduğu direkleri devirdi.
News Text 2	Ankara'da Başbakan Recep Tayyip Erdoğan başkanlığında Bakanlar Kurulu toplantısı devam ettiği sırada Başbakanlık binasına gelen ve adının Nuri Başkapan olduğu bildirilen bir kişi, binaya girerek kuru sıkı tabanca ile üç el ateş etti. Şüpheli gözaltına alındı	Başbakanlık Merkez Bina'da, Bakanlar Kurulu'nun toplanmasından kısa süre sonra hareketli dakikalar yaşandı. Toplantının başlamasından 15 dakika sonra Merkez Bina'da bakanların giriş yaptığı bölümde silah sesleri duyuldu. A kapısından içeri giren bir şahsın, silahını ateşlediği öğrenildi. Korumaların hemen apar topar eylemciyi etkisiz hale getirdiği görülürken, silahını ateşleyen şahsın Nuri Başkapan olduğu öğrenildi.	Başbakanlık'ta Bakanlar Kurulu toplantısının başlamasından 10 dakika sonra 3 el silah sesi duyuldu. Alınan bilgiye göre saldırgan, B Blok girişindeki güvenlik noktasından geçerek yetkili bir kişiyle görüşmek istedi. Ancak üzerinde kuru sıkı tabir edilen tabanca tespit edilen saldırgan, üç el ateş etti. Tabanca elinden alınarak etkisiz hale getirilen kişi gözaltına alındı.
News Text 3	Genelkurmay Başkanı, beraberinde Hava Kuvvetleri Komutanı olduğu hâlde, Malatya ve Diyarbakır'da konuşlu Hava Kuvvetleri Komutanlığı birliklerinde inceleme ve denetlemelerde bulunmuşlardır.	Genelkurmay Başkanı Orgeneral Necdet Özel'in beraberindeki Hava Kuvvetleri Komutanı Orgeneral Mehmet Erten ile birlikte Malatya'daki 7. Ana Jet Üssü'nde inceleme ve denetleme yaptığı bildirildi. Orgeneral Özel'in daha sonra Diyarbakır'a geçerek buradaki Diyarbakır 2. Hava Kuvvet Komutanlığında ve Diyarbakır 8. Ana Jet Üssü'nde inceleme ve denetleme yaptığı aktarıldı.	Genelkurmay Başkanı Orgeneral Necdet Özel, Malatya ve Diyarbakır'da konuşlu Hava Kuvvetleri Komutanlığı birliklerinde inceleme ve denetlemelerde bulundu. Genelkurmay Başkanlığı'nın internet sitesinde yer alan açıklamaya göre, Orgeneral Necdet Özel, Hava Kuvvetleri Komutanı Orgeneral Mehmet Erten ile Malatya ve Diyarbakır'da konuşlu Hava Kuvvetleri Komutanlığı birliklerinde inceleme ve denetleme yaptı.
News Text 4	Özal'ın cenazesi, İstanbul Büyükşehir Belediyesi Mezarlıklar Müdürlüğü'nün cenaze aracıyla polis eskortu eşliğinde saat 10.00'da Topkapı'ya getirildi. Özal'ın Türk bayrağına sarılı tabutu tören mangası tarafından omuzlara alınarak, anıt mezara taşındı. Cumhurbaşkanı Özal, 3 gün önce çıkarıldığı mezara, dualar eşliğinde 19 yıl aradan sonra ikinci kez defnedildi. Yaklaşık 45 dakika süren süren dini tören, basın mensupları anıt mezarın dışında tutulduğu için görüntülenemedi.	2 Ekim'de mezarından çıkarılan 8. Cumhurbaşkanı Özal için uzmanların ölümüne ilişkin zehirlenme iddialarıyla ilgili bulguların belirlenmesi için yaptığı çalışmalar dün tamamlandı. Merhum cumhurbaşkanının naaşı saat 09.40'da Adli Tıp Kurumu'ndan tören mangası ile birlikte alınarak, Anıt Mezar'a götürüldü. Naaşının büyük bölümünün bozulmadığı belirtilen Özal için düzenlenen dini törene, ailesi ve yakınlarının yanı sıra vatandaşlar katıldı. Bu sabah Adli Tıp Kurumu'ndan İstanbul Büyükşehir Belediyesi Mezarlıklar Müdürlüğü yetkililerine teslim edilen Özal'ın naaşı, güvenlik önlemleri altında Anıt Mezar'da toprağa verildi.	Adli Tıp Kurumu tarafından Ankara Cumhuriyet Başsavcılığı'nın, ölümüyle ilgili iddialar üzerine yürüttüğü soruşturma kapsamında 2 Ekim'de mezarından çıkarılan Turgut Özal'ın cenazesi, İstanbul Büyükşehir Belediyesi Mezarlıklar Müdürlüğü'nün cenaze aracıyla polis eskortu eşliğinde saat 10.00'da Topkapı'ya getirildi. Özal'ın Türk bayrağına sarılı tabutu tören mangası tarafından omuzlara alınarak, anıt mezara taşındı. Semra Özal'ın, eşinin tabutu merdivenlerden indirilirken gözyaşlarını tutamadığı görüldü. Tören mangası tarafından taşınan tabut, mezar kazılmadan önce sac levhalarla kapatılan bölüme getirildikten sonra İstanbul Müftüsü Rahmi Yaran'ın katıldığı dini törenle başladı. Cumhurbaşkanı Özal, 3 gün önce çıkarıldığı mezara, dualar eşliğinde 19 yıl aradan sonra ikinci kez defnedildi.

In second real data set, 12 news texts were retrieved from four different news agencies in three different topics (DHA: Dogan News Agency; IHA: Ihlas News Agency; AA: Anadolu Agency; CHA: Cihan News Agency).

The words in the news texts used in the applications were first subjected to stemming (root-affix parsing) operation. All words were cleansed of affixes and only the roots were included in the analysis. The purpose of this operation is to determine the words that have same roots and same meanings but changed form with affixes and to prevent them from being understood as separate words. The Zemberek Natural Language Processing algorithm was used for stemming. In addition to stemming, the stop-words were removed from the texts. Function words such as conjunctions and postpositions were picked out of the texts. The matrix which shows the frequency of each word in the texts was formed using the Term Frequency model after the stemming and picking process. This matrix comprises of 228 rows and 12 columns for real data set-1.

Table 4.9 Real dataset-2: News texts with three topics

DHA	IHA	AA	CHA
Russell Crowe, yönetmen koltuğunda oturduğu ve başrolünü oynadığı 'The Water Diviner'ın İstanbul çekimlerinin ardından Fethiye'ye geldi. Filmin son sahnelerinin çekileceği Kayaköy, Crowe için film platosu haline getirildi. Crowe, tarihi kilise ve evlerde çekimleri tamamlanacak film için yapılan son hazırlıkları yerinde inceledi.	Gladyatör filminin ünlü oyuncusu Avustralyalı Russell Crowe, The Water Diviner filminin çekimlerine İstanbul'un ardından Muğla'nın Fethiye ilçesinde devam ediyor. Çanakkale Savaşı'nda iki oğlunu kaybeden ve sağ kalan diğer oğlunu bulmak için büyük bir mücadeleye girişen Avustralyalı bir çiftçiyi canlandıran Crowe, Kayaköy'de atlı sahnelerin yanı sıra kilisedeki çekimlerini gerçekleştirirken bu sahnelerde Yılmaz Erdoğan da eşlik ediyor.	Avustralyalı aktör Russell Crowe'un yönetmenliğini ve başrolünü üstlendiği "The Water Diviner" filminin Türkiye'deki sahnelerinin bir bölümünün Fethiye'nin tarihi Kayaköy ören yerinde çekilmesinin ilçenin tanıtımına katkı sağlayacağı bildirildi. Çanakkale Savaşı'nda çocuklarını kaybeden ve onların ardından Türkiye'ye gelen bir babanın hikâyesini anlatan "The Water Diviner" filminin İstanbul'daki çekimleri, 26 Şubat'ta başladı.	Oscar ödüllü oyuncu Russell Crowe, yönetmen koltuğunda oturduğu ve başrolünü oynadığı 'The Water Diviner'ın İstanbul çekimlerinin ardından Fethiye'ye geldi. Filmin son sahnelerinin çekileceği 5 bin yıllık Kayaköy, Crowe için film platosu haline getirildi. Ünlü yönetmen, tarihi kilise ve evlerde çekimleri tamamlanacak film için yapılan son hazırlıkları yerinde inceledi.
Malezya Havayolları'na ait 239 kişi taşıyan yolcu uçağı Tho Chu adası yakınlarında denize çakıldı. Kuala Lumpur - Pekin uçuşunu yapmakta olan MH370 sefer sayılı Boeing 777-200 tipi yolcu uçağıyla bağlantı yerel saatle gece 2.40'ta (TSİ 20.40) kesilmişti.	Kuala Lumpur'dan Pekin'e gitmekte olan Malezya Havayolları'na ait yolcu uçağı, Vietnam yakınlarında denize düştü. Uçakta 227'si yolcu, 12'si mürettebat toplam 239 kişinin bulunduğu öğrenildi. Uçaktan sinyal alınan bölgeye arama kurtarma ekiplerinin sevk edildiği açıklandı	Malezya Havayollarına ait 227 yolculu MH 370 sefer sayılı yolcu uçağından haber alınamıyor. Malezya Havayollarından yapılan açıklamada, Kuala Lumpur'dan Pekin'e gitmek üzere havalanan 12 mürettebat ve 227 yolcusu bulunan uçağın kaybolduğu bildirildi.	Malezya havayollarına ait Boeing B777-200 tipi uçakla iletişim kesildi. Kuala Lumpur'dan Pekin'e hareket eden uçak ile bağlantı kesildi. Uçakta 239 yolcu 12' de kabin görevlisi bulunuyordu. Yolculardan 160'ının Çin vatandaşı olduğu belirtildi.
Galatasaray, Şampiyonlar Ligi 2. tur ilk maçında İngiltere'nin Chelsea takımı ile 1-1 berabere kaldı. Türk Telekom Arena'da oynanan maçta Galatasaray'ın golünü 64. dakikada Aurelien Chedjou atarken, konuk ekip Chelsea'nin golünü ise 9. dakikada Fernando Torres kaydetti.	Şampiyonlar Ligi 2. Tur ilk maçında Galatasaray, Türk Telekom Arena'da İngiliz ekibi Chelsea'yi ağırladı. 1-1 biten karşılaşma sonunda düzenlenen basın toplantısında iki takımın teknik direktörleri açıklamalarda bulundu.	UEFA Şampiyonlar Ligi 2. tur ilk maçında Galatasaray, İngiliz ekibi Chelsea ile 1-1 berabere kaldı. Çeyrek finale kalacak takımı rövanş karşılaşması belirleyecek. 64. dakikada Galatasaray beraberliği sağladı. Sneijder'in soldan kullandığı korner atışında altıpasta uygun durumda topla buluşan Chedjou, meşin yuvarlağı ağlarla buluşturdu: 1-1	Şampiyonlar Ligi 2. tur ilk maçında Galatasaray, konuk ettiği İngiltere Premier Ligi'nin güçlü ekibi Chelsea ile 1-1 berabere kaldı. Türk Telekom Arena'da oynanan ve İspanyol hakem Carlos Velasco Carballo'nun düdük çaldığı maçta konuk Chelsea'nin golünü 9. dakikada Torres attı. Galatasaray'ın beraberlik golünü ise 60. dakikada Chedjou kaydetti.

4.2.1 LSA Applications for Real Data Sets

In order to apply Latent Semantic Analysis on Turkish texts, the words should be parsed into their roots and suffixes. After the stemming operation, the function words, which do not express meanings, but relation between meanings, should be identified.

In first application of LSA, real data set-1 with four different topics is used. The news texts about the same topic from various news agencies were collected. News texts about four different topics were obtained from three different news agencies and a total of 12 texts were gathered.

After stemming and stop-word removal process, an input matrix was formed from the root words using the term frequency method. Words with a frequency of two or more were considered in forming the matrix. The purpose of this operation was to shrink the matrix and to obtain more effective results.

Table 4.10 A part of the word-document table created for real data set-1 with four topics

	DHA 1	IHA 1	AA 1	DHA 2	IHA 2	AA 2	DHA 3	IHA 3	AA 3	DHA4	IHA 4	AA 4
adli	0	0	0	0	0	0	0	0	0	0	2	1
alın	0	0	0	1	0	3	0	0	0	0	1	1
ana	0	0	0	0	0	0	0	2	0	0	0	0
anıt	0	0	0	0	0	0	0	0	0	2	2	1
ankara	0	0	0	1	0	0	0	0	0	0	0	1
ara	0	0	0	0	0	0	0	0	0	1	0	1
araç	0	0	0	0	0	0	0	0	0	1	0	1
ata	1	0	1	0	0	0	0	0	0	0	0	0
ateş	0	1	1	1	2	1	0	0	0	0	0	0

The calculations required for LSA were conducted in MATLAB. The dimension reduction to the matrices obtained after SVD was performed with regard to the number of the topics.

Table 4.11 Correlation table obtained as a result of the LSA (real data set-1 with four topics)

	DHA News 1	IHA News 1	AA News 1	DHA News 2	IHA News 2	AA News 2	DHA News 3	IHA News 3	AA News 3	DHA News 4	IHA News 4	AA News 4
DHA News 1	1	0.989	0.997	-0.075	-0.219	-0.056	-0.211	-0.175	-0.161	-0.231	-0.163	-0.246
IHA News 1	0.989	1	0.994	0.060	-0.082	0.083	-0.256	-0.225	-0.214	-0.241	-0.172	-0.249
AA News 1	0.997	0.994	1	-0.016	-0.162	-0.007	-0.182	-0.148	-0.135	-0.269	-0.200	-0.282
DHA News 2	-0.075	0.060	-0.016	1	0.988	0.994	-0.124	-0.170	-0.188	-0.366	-0.362	-0.326
IHA News 2	-0.219	-0.082	-0.162	0.988	1	0.9851	-0.127	-0.177	-0.197	-0.283	-0.289	-0.241
AA News 2	-0.056	0.083	-0.007	0.994	0.9851	1	-0.212	-0.257	-0.274	-0.295	-0.291	-0.254
DHA News 3	-0.211	-0.256	-0.182	-0.124	-0.127	-0.212	1	0.998	0.996	-0.259	-0.247	-0.269
IHA News 3	-0.175	-0.225	-0.148	-0.170	-0.177	-0.257	0.998	1	0.999	-0.243	-0.229	-0.255
AA News 3	-0.161	-0.214	-0.135	-0.188	-0.197	-0.274	0.996	0.999	1	-0.244	-0.230	-0.257
DHA News 4	-0.231	-0.241	-0.269	-0.366	-0.283	-0.295	-0.259	-0.243	-0.244	1	0.996	0.999
IHA News 4	-0.163	-0.172	-0.200	-0.362	-0.289	-0.291	-0.247	-0.229	-0.230	0.996	1	0.995
AA News 4	-0.246	-0.249	-0.282	-0.326	-0.241	-0.254	-0.269	-0.255	-0.257	0.999	0.995	1

As it is seen in the Table 4.11, the correlation between the texts about the same topic is quite high. The correlations of the texts mentioning different news are low, even negative. The algorithm used yielded successful results for all texts, and also it is seen that texts with closer meanings could be identified.

In second application of LSA, real data set-2 with three different topics was used. The news texts about the same topics were collected from four different news agencies. Words with a frequency of two or more were selected for input matrix. LSA calculations were made with MATLAB.

Table 4.12 A part of the word-document table created for real data set-2 with three topics

	DHA1	IHA1	AA1	CHA1	DHA2	IHA2	AA2	CHA2	DHA3	IHA3	AA3	CHA3
Ait	0	0	0	0	1	1	0	1	0	0	0	0
Alın	0	0	0	0	0	1	1	0	0	0	0	0
Arena	0	0	0	0	0	0	0	0	1	1	1	1
Art	1	1	1	1	0	0	0	0	0	0	0	0
At	0	1	0	0	0	0	0	0	1	0	1	1
Bağlantı	0	0	0	0	1	0	0	1	0	0	0	0
Beraber	0	0	0	0	0	0	0	0	1	0	2	2
Bildir	0	0	1	0	0	0	1	0	0	0	0	0
Bir	0	2	2	0	0	0	0	0	0	0	0	0
Boeing	0	0	0	0	1	0	0	1	0	0	0	0
Bul	0	0	0	0	0	1	1	1	0	1	0	0

Table 4.13 Correlation table obtained as a result of the LSA (real data set-2 with three topics)

	DHA1	IHA1	AA1	CHA1	DHA2	IHA2	AA2	CHA2	DHA3	IHA3	AA3	CHA3
DHA1	1	0.927	0.968	1.000	-0.471	-0.506	-0.459	-0.454	-0.374	-0.548	-0.375	-0.403
IHA1	0.927	1	0.990	0.931	-0.113	-0.147	-0.098	-0.092	-0.509	-0.567	-0.509	-0.513
AA1	0.968	0.990	1	0.970	-0.235	-0.275	-0.221	-0.216	-0.505	-0.606	-0.506	-0.519
CHA1	1.000	0.931	0.970	1	-0.463	-0.498	-0.450	-0.446	-0.379	-0.550	-0.379	-0.407
DHA2	-0.471	-0.113	-0.235	-0.463	1	0.993	1.000	0.999	-0.343	-0.023	-0.342	-0.281
IHA2	-0.506	-0.147	-0.275	-0.498	0.993	1	0.994	0.995	-0.238	0.087	-0.237	-0.173
AA2	-0.459	-0.098	-0.221	-0.450	1.000	0.994	1	1.000	-0.343	-0.023	-0.342	-0.280
CHA2	-0.454	-0.092	-0.216	-0.446	0.999	0.995	1.000	1	-0.334	-0.013	-0.333	-0.271
DHA3	-0.374	-0.509	-0.505	-0.379	-0.343	-0.238	-0.343	-0.334	1	0.947	1.000	0.998
IHA3	-0.548	-0.567	-0.606	-0.550	-0.023	0.087	-0.023	-0.013	0.947	1	0.947	0.966
AA3	-0.375	-0.509	-0.506	-0.379	-0.342	-0.237	-0.342	-0.333	1.000	0.947	1	0.998
CHA3	-0.403	-0.513	-0.519	-0.407	-0.281	-0.173	-0.280	-0.271	0.998	0.966	0.998	1

In Table 4.13, the correlation between texts about same topic is quite high. Also correlation between texts about different topic is about zero or negative. The method yielded successful results on data set with three topics too.

4.2.2 PLSA Applications for Real Data Sets

In this section, performance of PLSA on Turkish texts was investigated. For first application the real data set-1 with four topics was used for PLSA. There were four different news topic and 12 news texts in the data set. Same word-document Table in Table 4.10 used as input matrix in this application.

The ε value was determined as 0.00000001 before the analysis. P(z) and P(w|z) probabilities were assigned randomly, and P(d|z) was assigned as equal, as 1/12 according to the number of texts. The desired ε value was reached after 35 iterations.

Table 4.14 PLSA text probabilities P(d|z) for real data set-1 with four topics

	DHA1	IHA 1	AA1	DHA2	IHA2	AA2	DHA3	IHA 3	AA3	DHA4	IHA4	AA4
Topic 1	0	0	0	0	0	0	0	0	0	0.289	0.251	0.459
Topic 2	0	0	0	0	0	0	0.195	0.402	0.402	0	0	0
Topic 3	0.337	0.229	0.432	0	0	0	0	0	0	0	0	0
Topic 4	0	0	0	0.271	0.339	0.388	0	0	0	0	0	0

As a result of PLSA the texts were classified into four topics. Each text had only one possibility to belong to only one topic. Each topic had three texts.

For second application, real data set-2 with three topics was used. There were three different topics from four different news agency and twelve news article. Same word-text table in Table 4.12 was used as input matrix.

Table 4.15 PLSA text probabilities P(d|z) for real data set-2 with three topics

	DHA 1	IHA 1	AA 1	CHA 1	DHA 2	IHA 2	AA 2	CHA 2	DHA 3	IHA 3	AA 3	CHA3
Topic1	0	0	0	0	0.269	0.256	0.243	0.230	0	0	0	0
Topic2	0.235	0.260	0.226	0.277	0	0	0	0	0	0	0	0
Topic3	0	0	0	0	0	0	0	0	0.276	0.180	0.234	0.308

According to the results in Table 4.15, the PLSA method classified the text into three different topics. Only the probability of the topic each text belonged was calculated and the probability for other topics was found 0. According to these results, it is seen that PLSA method yielded accurate results on data set with three topics.

4.2.3 LDA Applications for Real Data Sets

For first application of LDA, the real data set-1 with four topics was used. The properties of this data set are;

V: 138 unique words in corpus

k: 4 topics

M: 12 documents

N: Number of words per document {25, 17, 32, 28, 35, 40, 18, 37, 37, 46, 40, 73}

In order to more efficient results, words with a frequency at least two were selected for the term frequency matrix. Due to the notation of LDA, transpose of Table 4.10 was used as input matrix.

Table 4.16 A part of the word-document table created for real data set-1 with four topics in LDA

	adli	alın	ana	anıt	ankara	ara	araç	ata	ateş
DHA 1	0	0	0	0	0	0	0	1	0
IHA 1	0	0	0	0	0	0	0	0	1
AA 1	0	0	0	0	0	0	0	1	1
DHA 2	0	1	0	0	1	0	0	0	1
IHA 2	0	0	0	0	0	0	0	0	2
AA 2	0	3	0	0	0	0	0	0	1
DHA 3	0	0	0	0	0	0	0	0	0
IHA 3	0	0	2	0	0	0	0	0	0
AA 3	0	0	0	0	0	0	0	0	0
DHA4	0	0	0	2	0	1	1	0	0
IHA 4	2	1	0	2	0	0	0	0	0
AA 4	1	1	0	1	1	1	1	0	0

LDA method was applied using the functions in the “topicmodels” package in R programming language and the results in Table 4.17 were obtained. According to the results in Table 4.17 the probabilities of texts about same topic are very close to 1. On the other hand the probabilities of texts about different topic are close to 0. According to these results LDA yielded successful results on real data set-1 with four topics.

Table 4.17 The results of LDA application on real data set-1 with four topics

	Topic 1	Topic 2	Topic 3	Topic 4
DHA 1	0.001	0.001	0.001	0.998
IHA 1	0.001	0.001	0.001	0.997
AA 1	0.000	0.000	0.000	0.999
DHA 2	0.001	0.001	0.998	0.001
IHA 2	0.000	0.000	0.999	0.000
AA 2	0.000	0.000	0.999	0.000
DHA 2	0.998	0.001	0.001	0.001
IHA 2	0.999	0.000	0.000	0.000
AA 3	0.999	0.000	0.000	0.000
DHA 4	0.000	0.999	0.000	0.000
IHA 4	0.000	0.999	0.000	0.000
AA 4	0.000	0.999	0.000	0.000

For second application of LDA, the real data set-2 with three topics was used.

Table 4.18 A part of the word-document table created for real data set-2 with three topics in LDA

	Ait	Arena	Art	At	Bir	Bul	Çekim
DHA1	0	0	1	0	0	0	2
IHA1	0	0	1	1	2	0	2
AA1	0	0	1	0	2	0	2
CHA1	0	0	1	0	0	0	2
DHA2	1	0	0	0	0	0	0
IHA2	1	0	0	0	0	1	0
AA2	0	0	0	0	0	1	0
CHA2	1	0	0	0	0	1	0
DHA3	0	1	0	1	0	0	0
IHA3	0	1	0	0	0	1	0
AA3	0	1	0	1	0	0	0
CHA3	0	1	0	1	0	0	0

The properties used in the data set are;

V: 53 unique words in corpus

k: 3 topics

M: 12 documents

N: Number of words per document {21, 25, 20, 24, 10, 13, 11, 14, 24, 16, 20, 26}

53 words were determined with a frequency of three or greater than three. There are 12 news texts from four different news agencies with three different topics.

According to the results in Table 4.19, LDA method separated 12 news texts in three topics. The probability of the topic in which a text belongs to, was found close to 1, and the probability for other topics was close to found 0. The probabilities of four texts, belongs the first topic, were calculated as 0.9981, 0.9984, 0.9980 and 0.9984 respectively. The probabilities for these texts to belong other subjects were extremely low. Similarly, the probabilities for the texts belongs the other topics were found close to 1 for the topics they belong. According to these results, it is seen that LDA yielded accurate results on real data set-2 with three topics, and classified the texts with regard to their topics.

Table 4.19 The results of LDA application on real data set-2 with three topics

	Topic 1	Topic 2	Topic 3
DHA 1	0.9981	0.0009	0.0009
IHA 1	0.9984	0.0008	0.0008
AA 1	0.9980	0.0010	0.0010
CHA 1	0.9984	0.0008	0.0008
DHA 2	0.0020	0.0020	0.9961
IHA 2	0.0015	0.0015	0.9970
AA 2	0.0018	0.0018	0.9964
CHA 2	0.0014	0.0014	0.9972
DHA 3	0.0008	0.9984	0.0008
IHA 3	0.0661	0.9327	0.0012
AA 3	0.0010	0.9980	0.0010
CHA 3	0.0008	0.9985	0.0008

4.2.4 LSA after PLSA Applications for Real Data Sets

An iterative algorithm which uses PLSA and LSA jointly is proposed to find the semantic similarities of the texts when the number of topics is not known prior to the analysis. The algorithm of LSA after PLSA was applied on the real data set-1 with four topics. 12 news texts of four different topics were taken from three different news agencies; stemming and picking procedures were conducted on these texts and

texts were represented in matrix form. The number of topics was not stated previously; the analyses were conducted 500 times for all values from 1 to 12, the number of the texts, and the results giving the desired P(d|z) matrix were counted.

Table 4.20 Table of frequencies of the desired P(d|z) matrix on real data set-1 with four topics.

T1	T2	T 3	T4	T5	T6	T7	T8	T9	T10	T11	T12
500	500	457	348	143	107	89	91	75	43	23	9

When the frequencies of the desired P(d|z) matrix were investigated, the maximum decrease was found in the fourth and the fifth cells (348 – 143=205). According to this, the analysis yielded a correct result and it determined the number of topics correctly as 4. This result, obtained from PLSA, was used in LSA and the semantic similarities of the texts were measured.

As a result of the trials, it was seen that when the number of the topics was not determined, the semantic similarity can be measured with the aforementioned algorithm. However, some difficulties were run into, during this process. The first one of these was that PLSA did also resulted in the expected P(d|z) matrix for the number of topics different than real value. The second one was that the expected P(d|z) matrix could not be obtained in some cases due to the random determination of the P(z) and P(w|z) probabilities at the beginning of the analysis.

Table 4.21 The correlation table resulted from LSA after PLSA for real data set-1 with four topics

	DHA 1	IHA 1	AA1	DHA 2	IHA 2	AA 2	DHA 3	IHA 3	AA 3	DHA4	IHA4	AA4
DHA 1	1	0.989	0.997	-0.075	-0.219	-0.056	-0.211	-0.175	-0.161	-0.231	-0.163	-0.246
IHA1	0.989	1	0.994	0.060	-0.082	0.083	-0.256	-0.225	-0.214	-0.241	-0.172	-0.249
AA 1	0.997	0.994	1	-0.016	-0.162	-0.007	-0.182	-0.148	-0.135	-0.269	-0.200	-0.282
DHA 2	-0.075	0.060	-0.016	1	0.988	0.994	-0.124	-0.170	-0.188	-0.366	-0.362	-0.326
IHA2	-0.219	-0.082	-0.162	0.988	1	0.9851	-0.127	-0.177	-0.197	-0.283	-0.289	-0.241
AA 2	-0.056	0.083	-0.007	0.994	0.9851	1	-0.212	-0.257	-0.274	-0.295	-0.291	-0.254
DHA 3	-0.211	-0.256	-0.182	-0.124	-0.127	-0.212	1	0.998	0.996	-0.259	-0.247	-0.269
IHA3	-0.175	-0.225	-0.148	-0.170	-0.177	-0.257	0.998	1	0.999	-0.243	-0.229	-0.255
AA 3	-0.161	-0.214	-0.135	-0.188	-0.197	-0.274	0.996	0.999	1	-0.244	-0.230	-0.257
DHA4	-0.231	-0.241	-0.269	-0.366	-0.283	-0.295	-0.259	-0.243	-0.244	1	0.996	0.999
IHA4	-0.163	-0.172	-0.200	-0.362	-0.289	-0.291	-0.247	-0.229	-0.230	0.996	1	0.995
AA 4	-0.246	-0.249	-0.282	-0.326	-0.241	-0.254	-0.269	-0.255	-0.257	0.999	0.995	1

LSA after PLSA algorithm was used on real data set-2 with three topics also. Number of topics is not stated previously. Probabilities calculated all possible number of topics and the results giving the desired P(d|z) matrix were counted. According to Table 4.22, the maximum decrease was found in the third and the fourth cells (441 – 214=227). That results yields number of topics is 3 and used in LSA.

Table 4.22 Table of frequencies of the desired P(d|z) matrix on real data set-2 with three topics.

T1	T2	T 3	T4	T5	T6	T7	T8	T9	T10	T11	T12
500	499	441	214	125	85	76	41	16	4	1	0

Table 4.23 The correlation table resulted from LSA after PLSA for real data set-2 with three topics

	DHA1	IHA1	AA1	CHA1	DHA2	IHA2	AA2	CHA2	DHA3	IHA3	AA3	CHA3
DHA1	1	0.927	0.968	1.000	-0.471	-0.506	-0.459	-0.454	-0.374	-0.548	-0.375	-0.403
IHA1	0.927	1	0.990	0.931	-0.113	-0.147	-0.098	-0.092	-0.509	-0.567	-0.509	-0.513
AA1	0.968	0.990	1	0.970	-0.235	-0.275	-0.221	-0.216	-0.505	-0.606	-0.506	-0.519
CHA1	1.000	0.931	0.970	1	-0.463	-0.498	-0.450	-0.446	-0.379	-0.550	-0.379	-0.407
DHA2	-0.471	-0.113	-0.235	-0.463	1	0.993	1.000	0.999	-0.343	-0.023	-0.342	-0.281
IHA2	-0.506	-0.147	-0.275	-0.498	0.993	1	0.994	0.995	-0.238	0.087	-0.237	-0.173
AA2	-0.459	-0.098	-0.221	-0.450	1.000	0.994	1	1.000	-0.343	-0.023	-0.342	-0.280
CHA2	-0.454	-0.092	-0.216	-0.446	0.999	0.995	1.000	1	-0.334	-0.013	-0.333	-0.271
DHA3	-0.374	-0.509	-0.505	-0.379	-0.343	-0.238	-0.343	-0.334	1	0.947	1.000	0.998
IHA3	-0.548	-0.567	-0.606	-0.550	-0.023	0.087	-0.023	-0.013	0.947	1	0.947	0.966
AA3	-0.375	-0.509	-0.506	-0.379	-0.342	-0.237	-0.342	-0.333	1.000	0.947	1	0.998
CHA3	-0.403	-0.513	-0.519	-0.407	-0.281	-0.173	-0.280	-0.271	0.998	0.966	0.998	1

According to the results in Table 4.22 and 4.23 the algorithm determined the number of topics and calculated similarity with LSA.

CHAPTER FIVE
CONCLUSION

In the traditional search techniques, keywords are needed, and the search results are presented in terms of inclusion of the keywords. On the other hand, searching for semantically similar texts is an approach which attracts more attention due to the developing Web technologies. Methods such as LSA, PLSA and LDA can be used to search semantic similarity of the documents. In order for semantic analysis methods to operate efficiently, the stemming and stop-word removal operations should be performed correctly. Stemming and stop-words removal processes are very crucial to make better semantic analysis. Before the any semantic analysis, lexical components such as polysemy, synonymy, antonymy, hyponymy, idiom, should be dealt. Stemming and stop-words removal processes are help to deal with those lexical components. In this study, stemming and stop-words processes removal are explained to deal lexical components. The stemming algorithm developed by Porter has a frequent use for English and has become a standard (Porter, 1980). Researchers studying on natural language processing and information retrieval have studies which produced successful results for Turkish stemming algorithms. But none of them has become a golden standard yet. Another crucial step is selection of text representation method. In this study, n-gram, term frequency and bag of words methods were explained. Term frequency method was used in applications.

In this thesis, information on LSA, PLSA and LDA methods were provided and the performances of these methods on Turkish news texts were included. With regard to results obtained, LSA can yield successful results in the determination of semantic similarities in Turkish texts. According to the result matrix, the news texts about the same topic showed high similarities while texts about different topics showed little similarity. The performance of the LSA method depends on the appropriateness of the parsing and dimension reduction procedures. In PLSA method, the probabilities of the topics to which the news texts belonged were calculated and the probabilities for the remaining topics were found zero. In LDA method, probability values very close to 1 were obtained for the topics to which the text belonged, and very low

values were found for the remaining topics. With regard to the results obtained, the texts expressing the same topic were separated from other texts. MATLAB and R statistical programming language were used for the implementation of these methods.

For the LSA, PLSA and LDA methods to yield accurate results, the number of the topics should be known exactly. In LSA algorithm, number of topic is used in dimension reduction process. A random number can be used in that process but in that case the results may fail. In PLSA algorithm, prior probabilities are generated by number of topics. In LDA algorithm, number of topics is used at Dirichlet distribution. However, in practice, the number of the topics to which the texts to be evaluated relate could not be known. An iterative algorithm which uses LSA and PLSA jointly is proposed to find the semantic similarities of the texts when the number of topics is not known prior to the analysis. The proposed algorithm is tried on news texts using MATLAB. According to the results obtained, first the number of topics can be estimated by PLSA, and then this estimated number of topics can be used in LSA method to calculate the semantic similarities of texts. "LSA after PLSA" algorithm was applied on real data sets and successful results were obtained. However, data sets with a great number of topics, algorithm takes a very long time. For the future, "LSA after PLSA" algorithm must be improved for cases where there are many topics.

In this study, term frequency model was used as text representation method. Different results can be obtained by using different text representation methods. In the last phase of the analysis Pearson correlation was used to determine the similarities. In this phase, different correlation coefficients can be tried. For the future studies, new approaches should be devised for enabling use of LDA method in cases where the number of topics is not known.

REFERENCES

Blei, M.D., Ng, A.Y., & Jordan, M.I.(2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research, 3,* 993-1022.

Brants, T., Chen, F., & Tsochantaridis, I. (2002). Topic based document segmentation with Probabilistic Latent Semantic Analysis. *Conference Information and Knowledge management.*

Deerwester, S., Dumais, S., Furnas, G., Laundauer, T., & Harshman R. (1990). Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Sciences, 41*(6), 391-407.

Dempster, A.P., Laird, N.M., & Rubin, D. (1977). Maximum likelihood estimation from incomplete data via the EM algorithm. *Journal Royal Statistical Society, 39,* 1–38.

Dittmer, J., & Parr, A.D., (2011). Mediating sovereignty: a comparative latent semantic analysis of US newspapers and conflicts in Kosovo and South Ossetia. *Media, War & Conflict, 4* (2), 124-141.

Girolami, M. & Kabàn, A. (2003). On an equivalence between PLSI and LDA. *26th annual international ACM SIGIR Conference on Research and Development in information retrieval.*

Grün, B., & Hornik, K. (2011). topicmodels: An R Package for Fitting Topic Models. *Journal of Statistical Software. 40* (13), 1-30.

Hearts, M.A., (1999). Untangling text data mining. *Association for Computational Linguistics*

Hofmann, T., (1999). Probabilistic latent semantic analysis. *15th Conference on Uncertainty in AI*.

Hofmann, T., (1999). Probabilistic latent semantic indexing, *SIGIR'99*.

Hofmann, T., (2001). Unsupervised learning by Probabilistic Latent Semantic Analysis. *Machine Learning, 42*, 177-196.

Landauer T.K., (1998). Learning and representing verbal meaning: The Latent Semantic Analysis Theory, *Current Directions in Psychological Science, 7* (5), 161-164.

Landauer, T.K., Foltz, P.W., & Laham, D. (1998). Introduction to Latent Semantic Analysis. *Discourse Processes, 25*, 259-284.

Lee, S., Song, J., & Kim, Y. (2010). An empirical comparison of four text mining methods. *Journal of Computer Information Systems, 51* (1), 1-10

Lewis, G.(1989) *Turkish: teach yourself books (1st ed.)*. London: Hodder and Stoughton.

Lin, C., Xue, G.R., Zeng, H.J., & Yu, Y. (2005). Using Probabilistic Latent Semantic Analysis for personalized web search. *APWeb 2005,* 3399, 707-717.

Makedonski, P.(2005) Finite State Morphology: The Turkish Nominal Paradigm, *Bachelor of Art Thesis, Universität Tübingen.*

Nakov, P., (2001a). Latent Semantic Analysis for Bulgarian literature investigation, *Spring Conference of Bulgarian Mathematicians Union.*

Nakov, P., (2001b). Latent Semantic Analysis for German literature investigation, *7th Fuzzy Days on Computational Intelligence*.

Nakov, P., (2001c). Latent Semantic Analysis for Russian literature investigation, *Bulgarian Naval Academy Conference*.

Nakov, P., Popova, A., & Mateev. P., (2001). Weight functions impact on LSA performance, *Recent Advances in NLP*

Ozsoy, G.M., Alpaslan, F.N., & Cicekli, I. (2011), Text Summarization Using Latent Semantic Analysis, *Journal of Information Science, 37*, 405-417.

Peterson, C., Bettes, B., & Seligman, M.E.P. (1985). Depressive symptoms and unprompted causal attributions: Content analysis. *Behaviour Research and Therapy*, *23*, 379-82.

Porter, M.F.,(1980). An algorithm for suffix stripping, *Program, 14*(3), 130−137.

Rossi, A.R., (2009). Latent Semantic Analysis of the Languages of the Life, *Computational Intelligence and Intelligent Systems*, *51*, 128-137.

Using English Term: Polysemy, (nd). Retrieved August 2013, from http://www.usingenglish.com/glossary/polysemy.html.

Uzun, V., & Yildiztepe, E., (2013). Gizli Anlam Analizi ile Türkçe Metinlerin Anlamsal Benzerliğinin Ölçülmesi. *Genç İstatistikçiler Sempozyumu*.

Uzun, V., & Yildiztepe, E., (2013). Measuring semantic similarity in Turkish textual data using LSA and PLSA. *8th International Statistics Congress*.

Yıldıztepe E. & Uzun V. (2014). Semantic analysis of Turkish Texts using Latent Dirichlet Allocation. *9th International Statistics Days Symposium.*

Wang, Z., Tsim, Y.C., Yeung, W.S., Chan, K.C., & Liu, J. (2007). Probabilistic latent semantic analyses (plsa) in bibliometric analysis for technology forecasting. *Journal of Technology Management & Innovation. 2* (1), 11-24

Zemberek NLP, (nd). Retrieved August, 2013 from http://zemberek-web.appspot.com.

APPENDICES

A1 MATLAB codes for LSA

```
function [p,M]=LSA(x,k)
[U,S,V]=svd(x);
u=U; v=V; s=S;
u(:,k+1:end)=[];
v(:,k+1:end)=[];
s(:,k+1:end)=[];
s(k+1:end,:)=[];
p=u*s*v';
M=corr(p);
return;
```

A2 MATLAB codes for PLSA

```
[Pd_z,n]=PLSA(x,k)
[p,l]=size(x);
n=0;
err=0.00001;
% p=>number of words
% l=>number of documents
% k=>number of topics

for g = 1:k
   for j = 1:l
      Pdi_z(g,j)=1/l;
   end
end
Pdi_z;
```

```
for i=1:k
  Pzk(1,i)=1/k;
end
%Pzk=rand(1,k);
Pzk;

Pwj_z=rand(k,p);
Pwj_z;

%Expectation Maximization phase
Pz_b=zeros(p,l);
Pw_b1=zeros(k,p);
Pb=zeros(1,k);
Pw_b2=zeros(k,l);
Pzk2=rand(1,k);
R=sum(sum(x));

while  sum(abs(Pzk2-Pzk)>err)>=1
   % while n<60

Pzk=Pzk2;

%expectation phase
Pwj_z;
Pdi_z;
Pzk;
% p=>number of words
% l=>number of documents
% k=>number of topics
Pb=zeros(1,k);
Pz_b=zeros(p,l);
```

```
Pw_b1=zeros(k,p);
Pw_b2=zeros(k,l);

%denominator
for i=1:p
   for j=1:l
      for g=1:k
         Pz_b(i,j)=Pz_b(i,j)+Pzk(g)*Pdi_z(g,j)*Pwj_z(g,i);
      end
   end
end
%P(z|d,w)
for i=1:p
   for j=1:l
      for g=1:k
         Pzk_di_wj(i,j,g)=Pzk(g)*Pdi_z(g,j)*Pwj_z(g,i)/(Pz_b(i,j)+eps);
      end
   end
end
Pzk_di_wj;

%maximization phase
% p=>number of words
% l=>number of documents
% k=>number of topics
%P(w|z)
%numerator
for j=1:p
   for i=1:l
      for g=1:k
         Pw_b1(g,j)=Pw_b1(g,j)+x(j,i)*Pzk_di_wj(j,i,g);
      end
```

```
    end
end
Pw_b1;

%denominator
for g=1:k
    for j=1:p
        for i=1:l
            Pb(g)=Pb(g)+x(j,i)*Pzk_di_wj(j,i,g);
        end
    end
end
for i=1:p
    for g=1:k
        for j=1:l
            Pw_z(g,i)=Pw_b1(g,i)/Pb(g);
        end
    end
end
Pw_z;
Pb;

% p=>number of words
% l=>number of documents
% k=>number of topics
%P(d|z)
%nominator
for g=1:k
    for i=1:l
        for j=1:p
            Pw_b2(g,i)=Pw_b2(g,i)+x(j,i)*Pzk_di_wj(j,i,g);
        end
```

```
    end
end
for i=1:l
    for g=1:k
        for j=1:p
            Pd_z(g,i)=Pw_b2(g,i)/Pb(g);
        end
    end
end
Pd_z;
%P(z)

for g=1:k
    Pzk2(g)=Pb(g)/R;
end
Pzk2;
Pwj_z=Pw_z;
Pdi_z=Pd_z;

n=n+1;
end
% save('PLSA.mat')
return;
```

A3 R codes for LDA

```
#“topicmodels” package was used
#k: number of topic
lda<-LDA(x,control=list(alpha = y), k = z)
lda_inf<-posterior(lda,x)
lda_inf$topics
```

A4 MATLAB codes for LSA after PLSA

```
[p,l]=size(x);
k=l;
t=0;
ox=0;
oxv=zeros(1,l);
for i=1:k
   t=0;
   while t<500
      [Pd_z]=PLSA(x,i);
      pk=Pd_z>0.09;
        if sum(sum(pk))<=l
            ox=ox+1;
         end
      t=t+1;
   end
oxv(1,i)=ox
ox=0;
end
f=zeros(1,l-1);
for i=1:l-1
   f(i)=oxv(i)-oxv(i+1);
end
[B,I]=max(f(:));
[p,M]=LSA(x,I)
```