IDENTIFYING THE EFFECTIVENESS OF A WEB SEARCH ENGINE WITH TURKISH DOMAIN DEPENDENT IMPACTS AND GLOBAL SCALE INFORMATION RETRIEVAL IMPROVEMENTS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF INFORMATICS
OF
THE MIDDLE EAST TECHNICAL UNIVERSITY

BY

GÜVEN FİDAN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
IN
THE DEPARTMENT OF INFORMATION SYSTEMS

FEBRUARY 2012

IDENTIFYING THE EFFECTIVENESS OF A WEB SEARCH ENGINE WITH TURKISH DOMAIN DEPENDENT IMPACTS AND GLOBAL SCALE INFORMATION RETRIEVAL IMPROVEMENTS

Submitted by **GÜVEN FİDAN** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Information Systems, Middle East Technical University** by,

Prof. Dr. Nazife Baykal ______________________________
Director, **Informatics Institute**

Prof. Dr. Yasemin Yardımcı Çetin ______________________________
Head of Department, **Information Systems**

Assoc. Prof. Dr. Onur Demirörs ______________________________
Supervisor, **Information Systems, METU**

Assist. Prof. Dr. Meltem Turhan Yöndem ______________________________
Co-Supervisor, **Computer Engineering, Okan University**

Examining Committee Members:

Assist. Prof. Dr. Aysu Betin Can ______________________________
Information Systems, METU

Assoc. Prof. Dr. Onur Demirörs ______________________________
Information Systems, METU

Assist. Prof. Dr. Meltem Turhan Yöndem ______________________________
Computer Engineering, Okan University

Dr. Ali Arifoğlu ______________________________
Information Systems, METU

Prof. Dr. İsmail Hakkı Toroslu ______________________________
Computer Engineering, METU

Date: 09.02.2012

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

Name, Last name : Güven Fidan

Signature : _________________

ABSTRACT

IDENTIFYING THE EFFECTIVENESS OF A WEB SEARCH ENGINE WITH TURKISH DOMAIN DEPENDENT IMPACTS AND GLOBAL SCALE INFORMATION RETRIEVAL IMPROVEMENTS

Fidan, Güven
Ph.D., Department Of Information Systems
Supervisor: Assoc. Prof. Dr. Onur DEMİRÖRS
Co-Supervisor: Assist. Prof. Dr. Meltem Turhan YÖNDEM

February 2012, 90 pages

This study investigates the effectiveness of a Web search engine with newly added or improved features in Web search engine architecture. These features can be categorized into three groups: The impact of link quality and usage information on page importance calculation; the use of Turkish stemmer for indexing and query substitution; and, the use of thumbnails for Web search engine result visualization.

As Web search engines have become the primary means for finding and accessing information on the Internet, the effectiveness of Web search engines should be evaluated on the idea of how effectively and efficiently they assist users achieve a query, which defines performance criteria rather than the pure precision and recall measures developed among basic information retrieval roles. In this thesis, we propose three distinguishing features to increase the efficiency of a Web search engine: The impact of link quality and usage information on page importance calculation outperforms classical hyperlink graph based methods notably, such as PageRank. The use of the Turkish stemmer for indexing and query substitution has remarkable improvements on Web relevance when used in a mixed framework with normal and stemmed forms. Finally, we have observed that users are able to find the most relevant results by using webpage thumbnails in the queries with decreased precision score values, despite their preferred search engine gazing behavior is much attributed.

Keywords: Search engine effectiveness, Turkish stemmer, webpage thumbnail, hyperlink graph.

ÖZ

BİR WEB ARAMA MOTORUNUN TÜRKÇE ALAN BAĞIMLI ETKİLERİ VE GENEL BİLGİ ALMA İYİLEŞTİRMELERİ İLE ETKİNLİĞİNİN BELİRLENMESİ

Fidan, Güven
Doktora, Enformatik Enstitüsü
Tez Danışmanı: Doç. Dr. Onur DEMİRÖRS
Yardımcı Tez Danışmanı: Yrd. Doç. Dr. Meltem Turhan YÖNDEM

Şubat 2012, 90 sayfa

Bu çalışmada, Web arama motoru mimarisine yeni eklenen veya iyileştirilmiş özelliklerin, Web arama motoru etkinliğine etkisi incelenmektedir. Bu özelikler üç grupta kategorize edilebilir: Web sayfalarının önemini belirlemede, sayfa bağlantı kalitesinin etkisi ve kullanım bilgisinin kullanımı; indeksleme ve sorgu düzenleme için Türkçe kelime kökü ayırıcının kullanımı; ve Web arama motoru sonuç görüntülemede, sayfalara ait küçük resimlerin kullanımı.

İnternet'te bilgiye erişim için birincil araç hale gelmiş olan Web arama motorlarının etkinliği, temel bilgi getiriminde kullanılan saf doğruluk duyarlılık ölçümleri yerine, performans kriteri olarak kullanıcılara, sorgularını nasıl etkili ve etkin bir şekilde gerçekleştireceklerine yardımcı olmaları ile değerlendirilmelidirler. Bu tezde, bir

Web arama motorunun etkinliğini arttırmak için üç ayırt edici özellik önerilmektedir. Sonuçlar, Web sayfalarının önemini belirlemede, sayfa bağlantı kalitesinin etkisi ve kullanım bilgisinin kullanımının, PageRank gibi klasik bağlantı çizgesi tabanlı yöntemlerden daha başarılı olduğunu göstermektedir. İndeksleme ve sorgu düzenleme için Türkçe kelime kökü ayırıcının kullanımı ise, normal ve kelime köküne ayrılmış formların bir çerçeve içinde kullanım ile, Web ilgililiğini dikkate değer şekilde geliştirmektedir. Ve son olarak, Web arama motoru kullanıcıları sonuçlara önyargılı odaklanmalarına rağmen, sonuç görüntülemede sayfalara ait küçük resimlerin kullanımı ile, duyarlılığı düşük sorgularda en alakalı sonuçlara ulaşabildikleri gözlenmektedir.

Anahtar Kelimeler: Arama motoru etkinliği, Türkçe kök ayırıcı, websayfası küçük resmi, bağlantı çizgesi.

To Kuzey, Öykü and Selda

ACKNOWLEDGEMENTS

I would like to express my deepest thanks to my supervisors Assoc. Prof. Dr. Onur Demirörs, Asst. Prof. Dr. Meltem Turhan Yöndem and Dr. Onur Tolga Şehitoğlu for their valuable assistance, guidance and patience through the duration of this thesis.

I would like to thank my thesis committee members for patiently spending their time and effort to read and comment on this thesis.

Finally, I would like to thank my family and express my deepest love and sincere gratitude to Selda for her endless support and understanding.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Motivation

The World Wide Web, or the Web shortly, is a collection of billions of documents such as hypertext documents, written in a way that connect to each other using hyperlinks, accessed via Internet[1]. This content is huge, and today it has become the largest and most diverse source of information in the world. Due to its dynamic nature, its size is increasing each day with multimedia sharing applications (e.g., video, image and audio), social networking sites, blogs, wikis, and different scaled enterprise Web sites, which makes it challenging to manipulate and reach relevant/required information. Today search engines are the first address for information seekers to begin their Web activity by sending millions of query requests and expecting the most appropriate results to start with. To accept this challenge, Web search engines are designed to search documents on the Web for user specified keywords, called search terms, and return a list of Web pages that may be relevant to the given keywords. Therefore, Web search engines should be efficient to respond user queries within milliseconds and effective enough for users to be satisfied with the information provided through the query process. Search engines periodically crawl the Web so as to get the most recent documents on the Web. These crawled document collections are stored on huge distributed document repositories for offline

[1] http://en.wikipedia.org/wiki/World_Wide_Web

processing tasks, such as calculating the link graph based page importance metrics or creating the inverted indexes structures which is the state-of-the-art Information Retrieval (IR) index structure (Zobel and Mo, 2006). When a user generates a query, the query keywords are sent to these inverted index structures and the relevant pages, meaning the pages with the highest scores, are displayed to the user based on their similarity scores with the given query keywords.

Search engines basically use well-known IR techniques that are developed for relatively small and coherent collections of data. However, search engines have to deal with the Web, which causes two challenging problems: The characteristics of the data; and the user and his interaction with the IR system (Baeze-Yates and Ribeiro, 1999).

Web data is distributed over millions of computers without known or predefined network topology, and the access method comes with multiple types of formats, languages and encodings. The volume of Web data has an exponential growth rate. In 1997, the size of Web indexes by Hotbot, Altavista, Excite and Infoseek, the largest search engine at that time, estimated at 200 million pages (Bharat and Broder, 1999); however, today global search engines like Google[2] and Bing[3] are supposed to index nearly 50 billion pages. This huge Web data also consists of poorly generated, redundant, and spam pages that decrease the quality of data. Web data is also highly volatile, which means it is easily added or removed. For example, Cho and Garcia-Molina (2003) reported that in a study of over half a million pages over 4 months, about 23% of pages changed daily. Web data is also semi structured, even unstructured, and cannot be easily modeled, even though it is assumed that the Web is a distributed hypertext. Broder et al. (2000) suggested that 28% of the pages constitute the strongly connected core of the so-called Web graph; 22% of the pages can be reached from the core but not vice versa; and 22% of the pages can reach core but not vice versa. The remaining pages cannot be reached from the core.

[2] http://www.google.com
[3] http://www.bing.com

The second class of problem is caused by the interaction of the user with the IR system. The formulation of the queries and the interpretation of the results are the main problem areas.

To solve these problems, a number of methods and approaches have been proposed within the last two decades. For example, distributed storage systems and parallel programming models are adapted to all tasks of a Web search engine, such as crawling, parsing, and indexing so that the scalability and efficiency of a Web search engine is notably increased. Also, the effectiveness of a Web search engine defines new performance criteria other than the pure precision and recall measures developed among traditional IR techniques. Thus, the main motivation and aim of this thesis study is to increase the effectiveness measures of a Web search engine with 3 main research areas**:** Webpage importance calculation; using stemmer for indexing and query substitution; and new result visualization components.

Many studies aiming to compute page importance can be found in literature; these are based on the structure of the link graph known as link analysis, such as PageRank (Brin et al., 1998, Page et al., 1999), HITS (Kleinberg, 1998) and TrustRank (Gyöngyi et al., 2004). Page importance calculation is one of the major problems that Web search engines struggle with due to vast, complex, and growing structure of the Web data. Working on such an environment, page importance calculation methods require continuous refinements.

Stemming is a commonly used technique for many IR studies and the effects of stemming on IR effectiveness are analyzed for different languages, such as Spanish (Figuerola et al., 2006), German (Braschler and Ripplinger, 2004), English (Hull, 1996; Krovetz, 1993), and Turkish (Can et al., 2006). These studies showed that stemming has positive effects on IR effectiveness. However, using stemming is a challenging area for a Web search engine, since stemming may reduce unrelated words to the same stem and it may fail to reduce related words to a common stem resulting in a decrease of Web relevance.

Web search engine user behavior analysis concentrates on why and how people search on the Web. Broder (2002) showed that there are three types of search goals: informational, navigational and transactional search. User-computer interaction studies, on the other hand, aim at understanding how people search on the Web, namely, how they react to and interact with Web search engines. Also, user clickthrough log analysis becomes the starting point for analyzing the user behavior (Joachims, 2002). Thus, how users interact with Web search engines in both the presence and the absence of visual components becomes a challenging problem.

1.2 Contributions

The contributions of this thesis consist of increasing the effectiveness of a Web search engine with three newly added or improved features. These are:

- The impact of link quality and usage information on page importance calculation,
- The use of Turkish stemmer for indexing and query substitution, and
- The use of thumbnails for Web search engine result visualization.

In the scope of link quality and usage information on page importance calculation, we assess the performance of PageRank algorithm on Turkish Web domain with TrustRank algorithm (Gyöngyi et al., 2004) fed by usage data and a refined link weighting scheme of the Web model. The results show that feeding PageRank with a trusted set of pages and refining the Web graph structure by giving different weights to different types of links improve PageRank performance notably. Our approach differs from those of the previous studies by concentrating on the hierarchical relationship of pages with their host and subdomains. Therefore, it is possible to say that this work is mutually exclusive with the previous work on PageRank link weighting scheme.

In the scope of the Turkish stemmer usage for indexing and query substitution, we propose a framework that increases IR precision score and Web relevance by using a stemmer/lemmatizer for Turkish Web search engine domain. We first show that for Turkish, which is a morphologically rich language, applying stemming increases retrieval effectiveness but decreases Web relevance. And then we show that the best precision, Web relevance pair is obtained by combining the results, retrieved by sending unstemmed query to unstemmed content and sending stemmed query to stemmed content. Another contribution of this study is that we adopt the MapReduce programming model for the stemming process during the indexing phase of the webpages. We show that Mapreduce overcomes the performance overheads of the stemming process.

In the scope of thumbnails usage for Web search engine result visualization, we investigate the effect of thumbnailed results on user behavior in Web search by interpreting users' eye tracking and clickthrough data. We concentrate on finding the answer to the question: "Can thumbnailed search results help users find the most relevant document on Search Engine Results Pages (SERPs)?" In order to answer this question, we added a mini snapshot of the corresponding webpage, called a thumbnail, in each search result, and we traced the eye movements and clicking behavior of the users to see if the thumbnails have an effect on users' searching and decision making behavior. When the ranking quality on a SERP was intentionally decreased, users were able to find the most relevant results by using thumbnails, but they were not able to do so in the absence of thumbnails.

The outline of this thesis is as follows:

- Chapter 2 provides the background information and literature survey about the general architecture of Web search engines. It also explains the base search engine framework to accomplish the work done throughout this thesis study. This is followed by the brief introduction to the three specific effectiveness challenges within the scope of this thesis: Link analysis for calculating Web relevance, stemmer usage in Web search engine, and Web search engine result visualization.

- Chapter 3 explains the impact of link quality and usage information on page importance calculation in detail. We first present that page importance calculation is one of the major problems that search engines struggle with and give background information and related work for determining the true model of this calculation. This is followed by an explanation of the details of link graph based algorithms, such as PageRank and TrustRank. Chapter 3 continues with our suggested modifications on these algorithms fed by usage data and refined link weighting scheme of the Web model. The experiments are conducted on Turkish Web domain, and their results are presented in detail in the rest of this chapter.
- Chapter 4 explains the use of Turkish stemmer for indexing and query substitution in detail. We first present the background information and related work for stemmer usage in IR precision score calculation on different languages and Turkish which is morphologically rich. We then explain our suggested framework for increasing IR precision score and Web relevance by using a stemmer/lemmatizer for Turkish Web search engine domain. We show in detail the experiments conducted for stemming effectiveness for both IR scores and Web relevance. The chapter ends with the discussion of these experiments and their results.
- Chapter 5 explains the effect of thumbnailed results on user behavior in Web search by interpreting users' eye tracking and clickthrough data. We first present the background information and related work for result visualization and user behavior analysis for Web search engines. We then give details about the eye tracking and user clickthrough implicit feedback mechanisms. The chapter continues with the effect of thumbnails, the mini snapshots of corresponding webpages, via conducting user experiments. The chapter ends with the discussion of these experiments and their results.
- Chapter 6 concludes with results and summarizes the contributions of this study which are presented in Chapters 3, 4 and 5. Potential follow-up work is also laid out in this chapter.

CHAPTER 2

BACKGROUND AND RELATED WORK

In this chapter, we provide background information for Web search engine architecture and briefly present related work in the literature about the link analysis for calculating Web relevance stemmer usage in Web search engine and Web search engine result visualization studies within the scope of this thesis. Detailed information from literature will be given in Chapters 3, 4 and 5 which are dedicated separately to these thesis topics.

2.1 Web Search Engine Architecture

Throughout this study, the required tasks are implemented on the top of a Web search engine, i.e. Bilgi.com[4]. Bilgi.com, which is funded by the Scientific and Technological Research Council of Turkey (TUBITAK[5]) as a research project conducted by AGMLab[6] Ltd., is a scalable, commercial Web search engine localized for the Turkish Web domain. Besides being a commercial search engine, Bilgi.com provides a research environment for this study with its open source development environment and the computer clusters available for research purposes located at METU Department of Computer Engineering. In the following sections, the presented search engine architecture is the one used for the basis of this thesis and

[4] http://www.agmlab.com/AGMLab_eng/bilgi_com.html
[5] http://www.tubitak.gov.tr
[6] http://www.agmlab.com

shall be seen in Figure 2.1. This architecture has been implemented on a cluster with 150 dual core computers with 600 terabytes of storage unit and 60 Mbits network bandwidth.

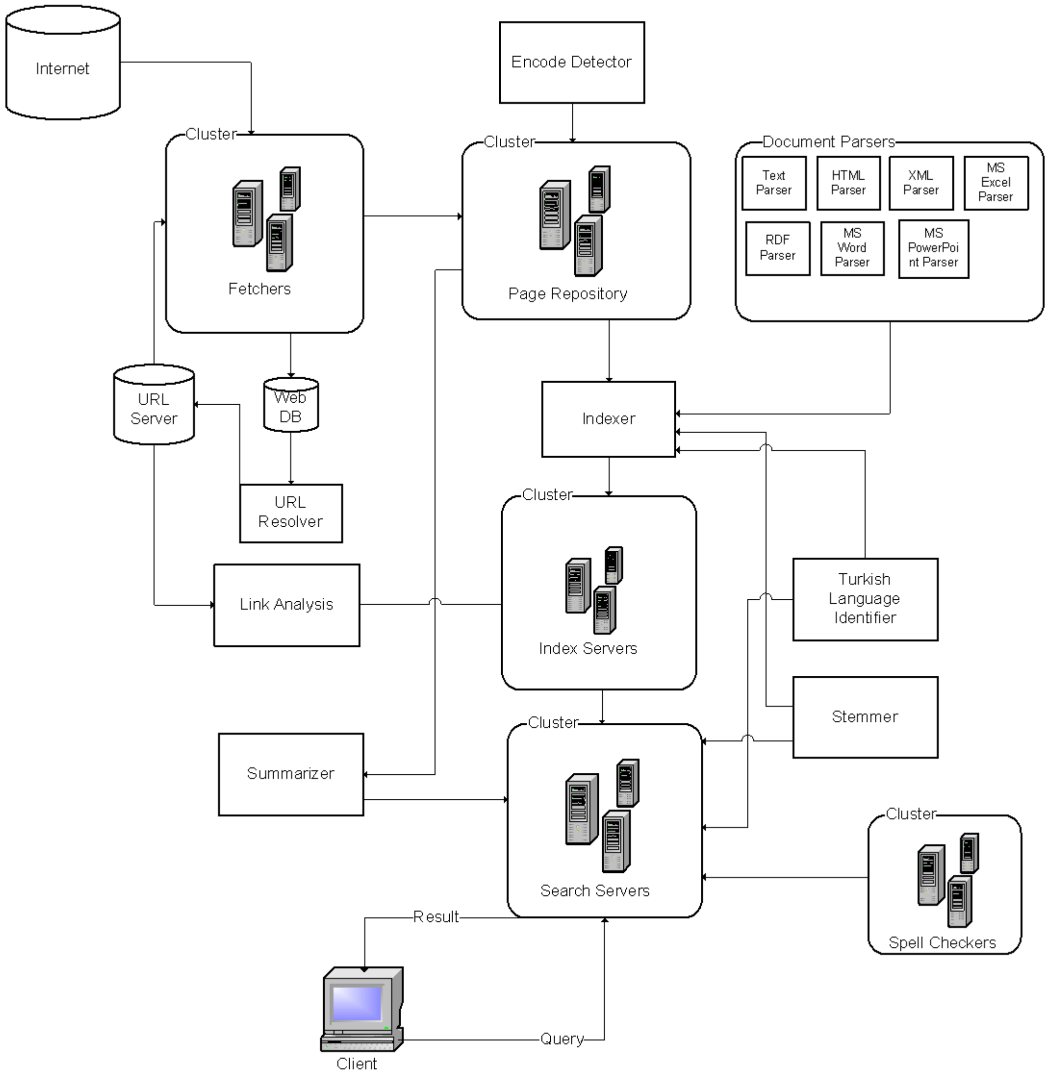


Figure 2.1: Web search engine architecture.

One of the basic components of the search engine is the *crawler* (or *fetcher*) module, which browses the Web similarly to how a human user follows links to reach target webpages. The crawlers are given a beginning list of URLs (i.e., seed list) that they will retrieve from the Web. The crawlers parse and then extract URLs appearing in the retrieved pages and give this information to the *WebDB* (Web database) module.

WebDB module determines what links to visit next and feeds the links to visit back to the crawlers. This information may come from the previous crawl cycles or the usage information that comes from the query modules. The crawlers also pass the retrieved pages into the page repository. Crawlers continue visiting the Web until local resources, such as storage, are exhausted or the crawlers decide to stop. These basic algorithms can be modified in some way for different coverage or topic bias. For example, crawlers might be biased to visit only specific sites, such as news sites or blogs; or crawlers might visit up to a certain level from the starting seed list.

The *indexer* module uses the contents of the parsed pages that are produced by different document *parsers* and are stored in the page repository. The result is a generally very large 'lookup table' that can provide all the URLs that point to pages where a given word occurs. Inverted files or inverted indexes (Salton, 1988; Witten et al., 1999) have traditionally been the index structure of choice on the Web. An inverted index over a collection of Web pages consists of a set of inverted lists, one for each word (or index term). The inverted list for a term is a sorted list of locations where the term appears in the collection and is called term-document matrix. In the simplest case, a location consists of a page identifier and the position of the term in the page. However, search algorithms often make use of additional information about the occurrence of terms in a Web page. For example, terms occurring in the webpage's title, metatags, parsed URL and anchor text might be weighted differently in the ranking algorithms.

The goal of the *query* module is to provide quality search results efficiently. The query engine module is responsible for receiving the search requests, i.e., search terms, from users and sends the preprocessed queries to index servers. Each index server processes the query using the inverted index structure and applies a ranking algorithm to sort the documents based on their relevance to the query. The *ranking* module has the task of sorting the results such that the results near the top *k* are the most likely to be what the user is looking for. The returned k-list of pages is combined with the summaries (or snippets), i.e., a well described text generated by the *summarizer* module based on the given search terms and displays to the users.

Search engines need a scalable storage system for managing large collections of Web content. This storage system, which is called as *page repository*, provides an efficient access means that the parser, indexer, ranking and summarizer modules can use to retrieve the contents. The search engine process is not a onetime pass mechanism, since the Web is rapidly changing (Cho and Garcia-Molina, 2000b; Wills and Mikhailov, 1999). Therefore, this dynamic, huge storage mechanism must be distributed on more than one server, via *distributed file system*, and with the ability to parallelize the search engine modules via *distributed processing mechanism*:

- *A Distributed File System*: Distributed file systems provide a reliable storage across a network of machines. Google proposed Google File System (Ghemawat et. al., 2003) (GFS) which stores files as a sequence of blocks and each one is replicated on multiple hosts with no single point of failure. Replication and fail-over are handled automatically, providing applications with an easy-to-manage, efficient file system that scales to multi-petabyte installations. The GFS architecture shall be seen in Figure 2.4(Ghemawat et. al., 2003).

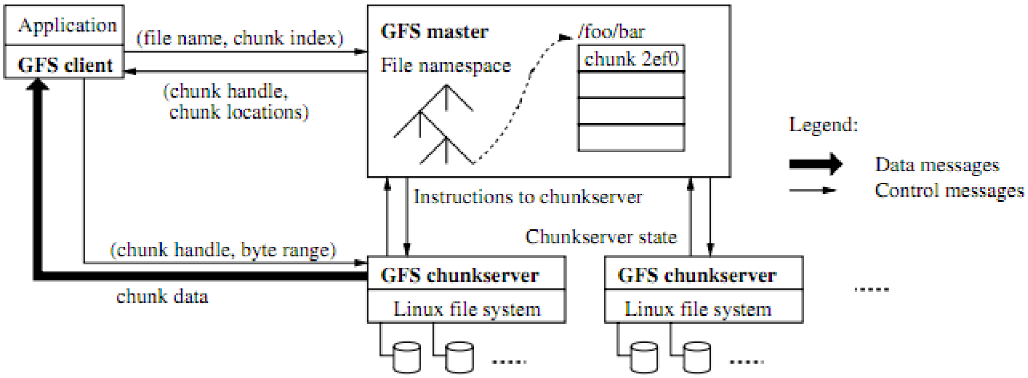


Figure 2.2. GFS architecture.

- *Distributed Processing Mechanism*: *MapReduce* is the most popular distributed computing layer of the search engines after Google published its architecture (Dean and Ghemawat, 2004). It is a programming model for processing large datasets. MapReduce, as its name implies, is a two-step operation, *map* followed by *reduce*. With this model, search engine processes are first implemented a map function that processes a key/value pair to generate an intermediate key/value pairs; after the map function completes, a reduce function merges the values having the same intermediate keys. The MapReduce programming model shall be seen in Figure 2.3. Today most of the search engine tasks, i.e., the components of the modules, can be adapted to the MapReduce programming model easily even for programmers without experience with parallel and distributed systems, since the details of parallelization, fault-tolerance, and load balancing are hidden by the MapReduce programming model.

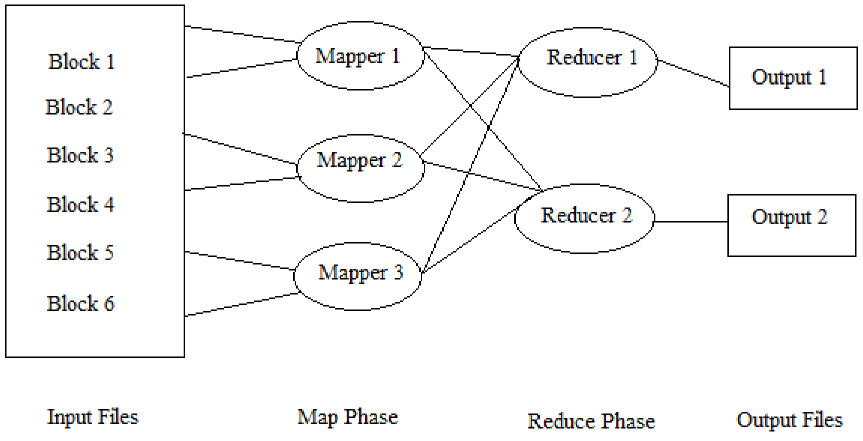


Figure 2.3: MapReduce programming model.

2.2 Link Analysis for Web Importance Calculation

When a user generates a query, the query module retrieves the ranked results after the search terms are sent to index servers where the inverted index structure computes the ranking of the webpages. However, the classical IR techniques are not effective enough to have a good ranking score, based on the characteristics (volume, quality, heterogeneous, etc.) of the Web data. Therefore, another means, the link structure of the Web, contains important implied information that may help in ranking or scoring the webpages. The considered idea is that if webpage A refers to page B, this means that A recommends B. The algorithms exploiting this link structure, give more information to ranking than just the contents of the pages.

The most well-known page importance algorithms based on the link graph structure are PageRank (Brin et al., 1998; Page et al., 1999), HITS (Kleinberg, 1998), and TrustRank (Gyöngyi et al., 2004). For example, the PageRank score depends on the number and scores of all pages that link to it. A page that is linked to by many pages with high score receives a high rank itself. If there are no links to a web page there is no support for that page. The variants of the PageRank algorithm have been suggested by Baeza-Yates et al. (2004), Xing and Ghorbani (2004) and Ding et al. (2002). To eliminate the spam pages, Chen et al. (2008), Nie et al. (2007) and Wu et al., (2006) considered new approaches to the TrustRank algorithm. The details of these algorithms and the literature will be discussed in Chapter 3.2.

2.3 Stemmer Usage in Web Search Engine

Although English is primarily the lingua franca of the Web, the penetration of both user and content of the other languages cannot be negligible. Based on the data provided by Internet World Stats[7], today 26.8% of the Internet users are English speakers (as native or primary language); however, it was estimated to be 36.5% in 2002. Thus, it is obvious that the global search engines shall handle non-English language queries and provide effective answers to the users, just like Google that has 149 local language interfaces[8].

The challenging research area is how well the search engines enable and encourage queries in non-English languages and to what extent they shall take into account the specific linguistic characteristics of these languages. A number of studies have been conducted to answer this question: Bar-Ilan and Gutman (2003; 2005) for Russian, French, Hungarian and Hebrew; Mujoo et al. (2000) for Indian; Moukhad and Large

[7] http://www.internetworldstats.com/stats7.htm
[8] http://www.google.com/language_tools?hl=en

(2001) for cross language IR; Kettunen et al. (2005) for Finnish; Braschler and Ripplinger (2004) for German; Figuerola et al. (2006) for Spanish; and Ahlgren and Kekalainen (2007) for Swedish.

The effect of stemming, which is removing inflectional elements and certain other endings from a word (Porter, 1980), as a linguistic tool on IR effectiveness is concerned by many researchers, such as Frakes and Baeza-Yates (1992); Figuerola et al. (2006) for Spanish; Braschler and Ripplinger (2004) for German (Braschler and Ripplinger, 2004); Hull (1996); and Krovetz (1993) for English.

The studies on the effectiveness of Turkish stemming methods on IR scores and search engines were conducted by Köksal (1981); Solak and Can (1994); Ekmekçioğlu and Willett (2000); Sever and Bitirim (2003); Can et al. (2006). The details of these studies and the literature will be discussed in Chapter 4.2.

2.4 Web Search Engine Result Visualization

Web search engines have become the primary means for finding and accessing information on the Internet. Nearly 70% of Web surfers are using a search engine as an entry point and generating millions of queries per day, and search engines present billions of results per week in response to these queries (Sullivan, 2006). To understand why and how people do search on the Web becomes a research area for many studies. Starting from the point of why people search on the Web, Broder (2002) introduced three types of search goals: informational, navigational and transactional search. In this taxonomy, informational search yields specific information; navigational search finds a specific webpage or site; and transactional search seeks for a site to make transactions, such as online gaming and shopping. Belkin (1993) stated that searching steps could be classified as: goal of the interaction; method of interaction; mode of retrieval; and, type of resource interacted with during the search.

Discovering the intent of Web searchers has the potential to drastically improve system performance of Web search engine (Gisbergen et.al, 2007; Jansen et al., 2008). User intent research falls into three sub-areas: empirical studies and surveys of search engine use; manual analysis of search engine transaction logs; and automatic classification of Web searches. A number of studies have been conducted to determine the user intent (e.g., Downey et al. (2008); Joachims (2002); Joachims et al. (2005), Buscher et al. (2010); Dumais et al. (2010); Huang et al. (2011); Lagun et al. (2011)). The details of these studies and the literature will be discussed in Chapter 5.2.

CHAPTER 3

THE IMPACT OF LINK QUALITY AND USAGE INFORMATION ON PAGE IMPORTANCE CALCULATION

Page importance calculation is one of the major problems that search engines struggle with due to vast, complex and growing structure of the Web. Working on such an environment, page importance calculation methods require refinements. To create an ideal Web graph structure, spam pages need to be filtered out first. Once the Web graph is cleaned from bad pages, the second step is to adjust link weights in order to prevent undeserved PageRank (Brin and Page, 1998) contributions of internal links to their own hosts or subsets. The motivation behind our study is to assess the performance of PageRank algorithm on Turkish Web domain by using the TrustRank algorithm (Gyöngyi et al., 2004) fed by usage data and a refined link weighting scheme of the Web model. Our results show that feeding PageRank with a trusted set of pages and refining the Web graph structure by giving different weights to different types of links improve PageRank performance notably.

3.1 Introduction

Coping with spam pages is always a challenge for search engines. Extracting a true model of the Web from huge sets of fetched webpages is another obstacle due to different characteristics of hyperlinks between pages. The current study proposes a new method to increase PageRank performance by integrating two major factors into the calculation. The first factor is filtering out spam pages by the help of a trusted set of pages created based on real Web users' browsing information. The second factor is to create a Web model in which internal links are considered less valuable compared to external links so that PageRank scores are distributed in a more objective way.

In our spam page elimination process we use the TrustRank algorithm (Gyöngyi et al., 2004). The heuristic behind TrustRank is that good pages usually give link to good pages and rarely to spam pages. The algorithm starts with a seed list of highly trusted pages and iterates over the set to determine more trustable pages by tracing outlinks from the seed list. Gyöngyi et al. (2004) use a seed selection method combining PageRank algorithm with human evaluation to determine the initial set of trusted pages. However, we follow a rather different approach on seed selection. Instead of using PageRank algorithm to extract top ranked webpages, we use a top ranked list created by pure human evaluation on Turkish Web domain. The trustable seed list we use is provided by Alexa[9] which is a Web information company collecting the browsing logs of "Alexa Toolbar users." Alexa provides a list of top ranked pages which are most frequently visited by the majority of users in the last 30 days. We use a domain specific Alexa data; that is only Turkish users' usage data, is used. For that reason, we have also implemented a variant of PageRank algorithm to eliminate non-Turkish pages from our data set and measure the performance under that condition.

Another factor that we evaluate as a performance metric for the PageRank algorithm is refining the link weighting scheme of Web model. The baseline of PageRank algorithm, proposed by Brin and Page (1998), takes all types of links equally weighted to 1. Therefore, the PageRank scores are not affected by the types of links on which they are carried. In our approach, we define three types of links; links between the pages that are within the same host or the same subdomain; links

[9] http://www.alexa.com

between the pages of different hosts; and links between the pages of different subdomains of the same host. With different weighted links, we observe that PageRank performance is improved compared to baseline algorithm.

3.2 Related Work

Many algorithms aiming to compute page importance may be found in literature. These methods are mainly based on the structure of the link graph known as link analysis. The most well-known works related to page importance are PageRank (Brin et al., 1998; Page et al., 1999), HITS (Kleinberg, 1998), and TrustRank (Gyöngyi et al., 2004).

A new PageRank variant, Weighted PageRank (WPR), has been proposed by Baeza-Yates et al. (2004). This variant functions by assigning different weights to links based on three attributes: the relative position of links in a page; the anchor tag used; and the length of the anchor text. Their argument is that webpage developers give more importance to some links by using these three attributes.

Xing and Ghorbani (2004) show a rather different approach on PageRank link weighting. Their approach, the Weighted-Link PageRank (WLPR), is to assign larger rank values to more important pages instead of distributing the PageRank values of pages evenly among outlinks. As a result, each page gets a PageRank value proportional to its importance.

An approach combining PageRank and HITS algorithm is proposed by Ding et al (2002). They generalize the key concepts in HITS and PageRank algorithms and create three new algorithms that are intermediates between HITS and PageRank: InormRank, OnormRank, and SnormRank. These algorithms stand for Inlink Normalized Rank, Outlink Normalized Rank, and Symmetric Normalized Rank. The baseline of PageRank distributes the PageRank scores depending on the outdegree of pages. In this model, indegree and combination of indegree and outdegree of pages are taken into consideration as well.

Chen et al. (2008) proposed a link variable TrustRank method to fight spam pages. They suggested taking the variance of the link structure into consideration. Nie et al. (2007) suggested a novel cautious surfer to incorporate trust into the process of calculating authority for webpages. They biased their surfer to favor more trustworthy pages when randomly jumping to a page. In some other studies (e.g., Wu et al., 2006) topical information has been used to partition the seed set and calculate trust scores for each topic separately.

In recent studies, users' behavior is included in page importance calculation. The hyperlink graph is combined with the usage graph that is obtained from user logs (obtained from Web) to adjust the transition weights between two pages (Oztekin et al., 2003). Similarly, Eirinaki and Vazirgiannis (2005), use the Web clickthrough logs and Web user behaviors (Liu et al., 2008). Their main idea is that real behavior data is crucial to calculate page importance. Following those approaches, Liu et al. (2010) presented a framework with a stochastic process, the user browsing process, for modeling the user behavior data. Their experiments have shown that the user behavior data modeling outperforms the traditional pure hyperlink graph based approaches like PageRank in determining page importance.

The baseline of PageRank and TrustRank algorithms is explained in Sections 2.1 and 2.2, respectively.

3.2.1 PageRank

Page ranking may simply be described by giving scores to webpages by including special factors. These scores are used by search engines to determine the relevancy between the query and the webpage. In PageRank, the content of the site is completely unimportant; the method basically depends on the graphical structure of the Web (Page et al., 1999). This structure is created by using the Markov Model. A

Web graph structure reflects the pages and the hyperlinks between them. There are two simplification procedures to create the Web graphs: to assume the multiple hyperlinks from page *q* to page *p* as one; and to remove the hyperlinks which link page p with itself.

In the HITS method, two values for each page are calculated: the degree of authority (page importance measure); and the degree of outdegree (measure of the usefulness for a Web user).

In PageRank, the importance, or the authority, of a page depends on both the number of incoming links to the page and the importance of the pages which lead to those links. Google visualizes the concept with a non-egalitarian voting mechanism; if a page is voted by a high authority page, then the authority of the voted page becomes high.

The calculated PageRank value represents the probability of arriving to a particular page by clicking the random links in a set of pages. According to Brin and Page (1998), PageRank is calculated by the Equation 3.1.

$$x_p = d \sum_{q \in pa[p]} \frac{x_q}{h_q} + (1 - d) \qquad \text{(Equation 3.1)}$$

In Equation 3.1, p is the voted page, q is the voter. x_p is the PageRank value of page p and p_a is the set of pages that give link to *p*. $d \in (1,0)$ is the dumping factor and it indicates the probability of the user to continue clicking at any time. h_q is the outdegree of q which means the hyperlinks that are casted from q.

In the Markov model, a vector is created to store PageRank values for each page. We call it vector x and it is represented by Equation 3.2.

$$\boldsymbol{x} = d\mathbf{W}\boldsymbol{x} + (1 - d)\mathbb{1}_N \qquad \text{(Equation 3.2)}$$

PageRank is an iterative method. A fixed number of iterations are used to calculate the score of the pages by means of the Jacobi method.

The vector function which has the iteration factor is shown in Equation 3.3.

$$\boldsymbol{x}(t) = d\mathbf{W}\boldsymbol{x}(t - 1) + (1 - d)\mathbb{1}_N \qquad \text{(Equation 3.3)}$$

W = { $w_{i,j}$ } is a transition matrix. Each column (j) contains the total value of 1 or 0. If a page(x) has no hyperlinks, jx is summed to 0, otherwise value 1 is distributed to the all outgoing links as 1/$h_{x.}$

3.2.2 TrustRank

The TrustRank method determines a semi-automatic way to separate spam pages from reliable pages which are worth indexing (Gyöngyi et al., 2004). It is not possible to evaluate the whole Web and decide manually whether the pages are real or spam. To automate this issue, a trust function is created shown in Equation 3.4. This function calculates a score between 0 and 1 for each page on the Web and it is called the ideal trust property.

$$\mathrm{T(p) = Pr\ [O(p) = 1]} \qquad \text{(Equation 3.4)}$$

In Equation 3.4, $O(p)$ is called the oracle function and it represents the human evaluation on page p. $O(p) = 1$ means page p is a trusted page where $O(p) = 0$ means p is a spam. As its name suggests, function $T(p)$ represents the ideal case. Practically it is very difficult to compute actual likelihood of being trusted for a page p. Instead, two more additional properties are used to make a distinction between trusted and spam pages: ordered and threshold trust properties.

Ordered trust property is useful when pages need to be ordered according to their TrustRank. This means, a search engine should order the results according to the validity. Equation 3.5 and Equation 3.6 represent the ordered trust property.

$$T(p) < T(q) \Leftrightarrow \Pr[\mathrm{O}(p) = 1] < Pr\ [O(q) = 1] \qquad \text{(Equation 3.5)}$$

$$T(p) = T(q) \Leftrightarrow \Pr[\mathrm{O}(p) = 1] = \Pr\ [\mathrm{O}(q) = 1] \qquad \text{(Equation 3.6)}$$

Threshold trust property, given in Equation 3.7, assumes that the pages are good if their TrustRank is greater than a threshold value. However, the quantity of the greatness cannot be used for ordering the results.

$$T(p) > \ \delta \Leftrightarrow \mathrm{O}(p) = 1 \qquad \text{(Equation 3.7)}$$

These three trust property; mentioned above, work in a cooperative way in the TrustRank algorithm. First, the ideal property of each page is calculated. The result is a set of pages and their corresponding $T(p)$ values showing how trustable they are. The next step is to sort the pages according to probabilities in consistency with ordered trust property. Finally, a threshold value δ between 1 and 0 is selected. This threshold value δ separates the sorted set into two where the subset containing higher scored pages are accepted as the set of trusted pages where the remaining pages are labeled as spams.

TrustRank algorithm determines trusted pages by iterating over the outlinks of pages that are accepted to be trusted. The idea is that if a page p has a backlink from a highly trusted page, then it is highly probable that page p is trusted, too. Therefore, TrustRank algorithm needs an initial seed list of highly trusted pages to iterate over. In the selection process of seeds, first the PageRank algorithm is used to rank all the pages. Then, the top ranked n pages are selected and evaluated by human evaluators to create the final seeds. The details about the process can be found in Gyöngyi's (2004) work.

To calculate the actual trust score of a page, we use the trust splitting method. The method works in the way shown in Figure 3.1.

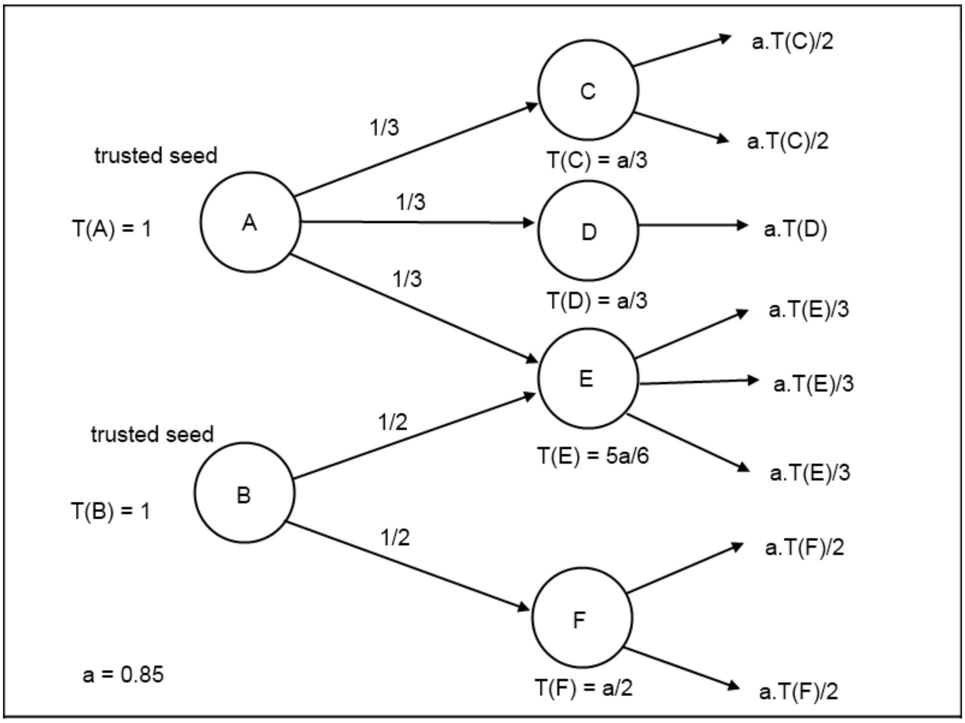


Figure 3.1: TrustRank calculation by splitting method.

In each iteration, we multiply the incoming trust scores with a decay factor α = 0.85. By the decay factor, as we trace more links, it becomes less and less probable that the arrived page is a trusted one.

To decide the usefulness of the function T in TrustRank, three function evaluation metrics are used. The first one is "pairwise orderedness" as shown in Equation 3.8.

$$pairord(T, O, \mathcal{P}) = \frac{|\mathcal{P}| - \sum_{(p,q)\in\mathcal{P}} I(T, O, p, q)}{|\mathcal{P}|} \qquad \text{(Equation 3.8)}$$

If the *pairord* value equals 0, this means there are no pair of pages which are in the wrong order for ordered property.

I is a binary function and decides if a spam page has a trust score more than or equal to a trustable page.

The other metrics are *precision* and *recall*. These are related with threshold property. Precision is a ratio which is the division of the pages that have a trust score greater than a threshold value with all pages in the page set X. It is calculated as shown in Equation 3.9.

$$\mathsf{prec}(\mathsf{T},\mathsf{O}) = \frac{|\{p \in \mathcal{X} | \mathsf{T}(p) > \delta \text{ and } \mathsf{O}(p) = 1\}|}{|\{q \in \mathcal{X} | \mathsf{T}(q) > \delta\}|} \qquad \text{(Equation 3.9)}$$

Recall, on the other hand, is a ratio which is the division of the pages that have a trust score greater than a threshold value with all “trustable” pages in page set X. The formula is shown in Equation 3.10.

$$\mathsf{rec}(\mathsf{T},\mathsf{O}) = \frac{|\{p \in \mathcal{X} | \mathsf{T}(p) > \delta \text{ and } \mathsf{O}(p) = 1\}|}{|\{q \in \mathcal{X} | \mathsf{O}(q) = 1\}|} \qquad \text{(Equation 3.10)}$$

3.3 Different Page-Rank and TrustRank Implementations based on Link Quality and Usage Data

We modified the PageRank and TrustRank algorithms in order to assess their performance under our link weighting scheme and usage data based trusted seed selection, respectively. These modifications are presented in the following sections.

Modifications on PageRank Algorithm

In order to improve the PageRank score calculation, we focused on a basic factor, which is modifying the link weighting scheme.

Varying the weight of links between webpages entirely relies on link topology. We specify three types of links:

1. Links between pages from different hosts.

2. Link between pages of different subdomains belonging to the same host.

3. Links between pages belonging to the same subdomain.

The first type of links is given the weight of 1. Therefore, they have no effect on the PageRank score that flow upon them.

The links in the second group are considered as internal links distributing PageRank scores to their neighbor pages within the same host. Objectivity in mind, they are given the link weight of 0.01 in order to prevent too much internal links promoting each other.

The links in the third group are prohibited (given the weight zero). As a result, they have no role in distributing PageRank score.

Under this new link weighting scheme, the modified PageRank score calculation function is given in Equation 3.11.

$$x_p = d \sum_{q \in pa[p]} w_l \frac{x_q}{h_q} + (1 - d) \qquad \text{(Equation 3.11)}$$

In Equation 3.11, w_l stands for the weight of the link between pages p and q.

Modifications on TrustRank Algorithm

In addition to the distinctive link weighting schemes, we assess the PageRank performance under spam page elimination with TrustRank. The modification we made on TrustRank covers the seed list selection part only. The rest of the algorithm is the same as what was proposed by Gyöngyi (2004). The seed list we use consists of Alexa's top ranked 200 URLs for Turkish domain.

The Overall Setup

Including link weighting and usage data integration, we have seven different features for PageRank algorithm shown in Table 3.1.

Table 3.1: Features used in different PageRank algorithms.

Feature No	Feature
1	All types of links are allowed and are given the same weight in the Web graph.
2	The links within the same subdomain are eliminated in the Web graph.
3	Only fetched pages are allowed in the Web graph.
4	A penalty (decrease in link weight) is given to internal link between subset and host or vice versa.
5	A penalty (decrease in link weight) is given to node pairs of internal link if the URLs serve for the same IP.
6	Only Turkish URLs are allowed since we do not discover outlinks of webpages in other locales; they become a sink in the Web graph.
7	The crawl data PageRank works on is filtered out of spam pages using the modified TrustRank.

In Table 3.2, the variants of the PageRank algorithm we use are presented.

Table 3.2: Variants of PageRank algorithm evaluated in this study.

Algorithm No	Features Used
1	Feature 1.
2	Feature 2.
3	Feature 2, Feature 3.
4	Feature 2, Feature 3, Feature 4, Feature 7.
5	Feature 2, Feature 3, Feature 4, Feature 5, Feature 6, Feature 7.

We discuss the effects of varying link quality and integrating usage data in the following sections.

3.3.1 The Effects of Link Quality

When assessing link quality, we take two factors into consideration. The first factor is eliminating spam pages, and the second one is assigning different weights to links of different types to reveal their true character in a Web model.

Since PageRank calculation relies on link topology, the link quality has a significant effect on PageRank calculation. The PageRank score of any webpage is proportional to the number of webpages that refers it. Therefore, most of the spam page owners create artificial links to their webpages in order to improve their PageRank score. To gather actual PageRank scores, spam page elimination is required.

We have discussed how the TrustRank algorithm automatizes the elimination of spam pages. Our TrustRank algorithm performs the elimination tracing through the most popular trusted pages among real Web surfers. The experiments we conducted suggest that integrating the real surfing information into TrustRank score calculation has a positive impact on increasing link quality of the Web model. PageRank variants 4 and 5 in Table 3.2, work on the page sets that are filtered using Alexa user browsing data, and these variants prove to have a better performance over the other variants which do not use such filtered page sets.

Considering link quality, classifying links into different categories is another factor we evaluate when assessing PageRank performance. Considering link topology among webpages, it is vital to see that internal and external links have different characteristics, so they should have different weights on the contribution of PageRank scores. As we analyze the link topology of the Web, we realize that internal links generally serve as navigational paths inside a website. Unlike internal links, external links have the purpose of pointing to an information source outside the website. As an example, we can think of internal links as references to a table or a figure within an academic paper; however, external links act more like references to other academic papers. Therefore, external links play a more significant role in PageRank calculation. Figure 3.2 depicts an example showing different link types in different colors. The link weights are given in the same color on the right bottom of the figure.

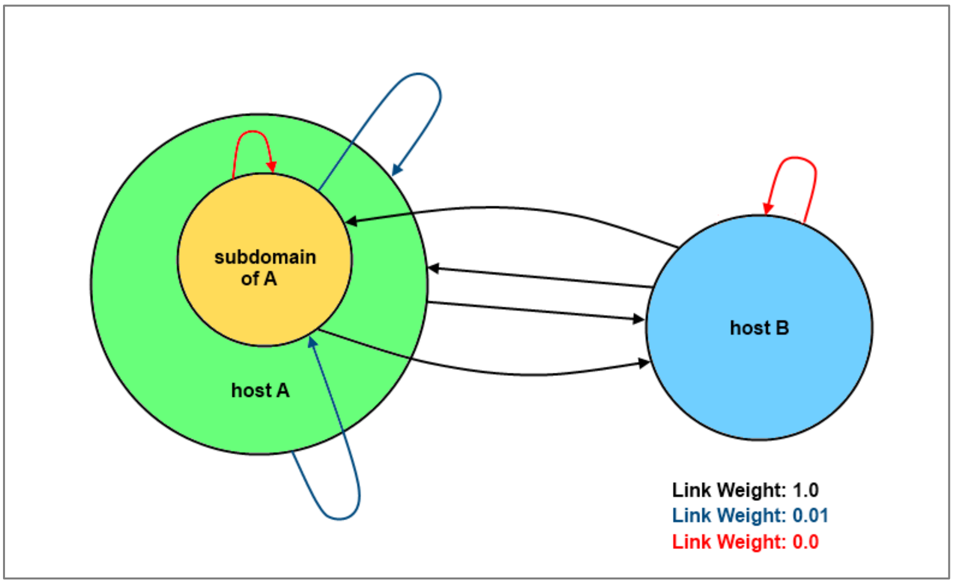


Figure 3.2: Link types taking different weights.

3.3.2 The Effect of Integrating Usage Data

It is already proven by Gyöngyi et al. (2004) that TrustRank can effectively identify a significant amount of good pages given a set of highly trusted set of pages. In this study, the goal is to measure the spam page elimination performance of TrustRank algorithm given a set of popular pages provided by Alexa. We observe that Alexa's top ranked lists can be an efficient source of trusted seed pages to be used with TrustRank algorithm. Detailed experimental results are shown in the following section.

3.4 Experimental Results

We evaluate five different versions of PageRank algorithms against Alexa's top lists in different categories. The best performance among these 5 algorithms is achieved by PageRank Case-4 and Case-5 algorithms. These two algorithms use a set of trusted pages, created by TrustRank, building their Web graph. The quality of TrustRank classification depends on the quality of the initial seed list of trusted pages, it is vital to provide a highly trustable seed list to TrustRank algorithm (Gyöngyi, 2004) Since the evaluation of webpages is a time and human-effort consuming job, we use Alexa's global list of top ranked webpages. Having TrustRank and PageRank running in cooperation, we integrate the usage data from Alexa into PageRank calculation and it significantly improves the PageRank results.

3.4.1 Comparison Metric

In order to compare the different implementations of the page-rank algorithm, we used the Kendall Tau Correlation metric. The Kendall Tau correlation is widely used for calculating the correlation between two sequences of items (Langville & Meyer, 2004). The important property of the Kendall Tau correlation is that the order of items in upper positions is more important than the order of items in lower positions. This property enables the Tau correlation to be one of the best methods for comparing search engine results. The value of the Tau correlation between two sequences of items is always a real number in the interval [-1, 1] in general which means that the positive values are accepted as a similarity and where the negative values shows dissimilarity. General properties of the Kendall Tau distance are given below:

- If there is a perfect match between two sequences which is the case where two rankings are the same, the correlation has value 1.
- If there is perfect disagreement between the two sequences which is the case where one ranking is the reverse of the other, the correlation has value -1.

- Apart from the two extreme cases mentioned above, the correlation value for all other arrangements the value lies between -1 and 1, and increasing values imply more correlation between sequences. If the sequences are completely independent, the correlation has value 0 on average.
- It is used for calculating similarity of two sequences having equal length.

The Kendall Tau correlation function is given in Equation 3.12 below.

$$\tau = \frac{2\text{P}}{\frac{1}{2}n(n-1)} - 1 = \frac{4\text{P}}{n(n-1)} - 1 \qquad \text{(Equation 3.12)}$$

Here, n is the number of items in each sequence, and *P* number of concordant pairs over all the items. It is calculated by counting the number of item pairs (a, b) for which *a* comes before *b* in both sequences. Higher values for P lead to higher values for the Tao correlations also. We provide the following example in order to show calculation for the Kendall Tau correlation.

Example:

Suppose we rank a group of five people by height and by weight where person C is the tallest and the fourth heaviest, etc. as shown in Table 3.3.

Table 3.3: An example of two ranked sets: by height and by weight.

Person	A	B	C	D	E
Rank by Height	4	3	1	5	2
Rank by Weight	5	4	3	2	1

We see that there is a correlation between the two rankings, but the correlation is far from perfect. We can use the Kendall Tau coefficient to objectively measure the degree of correlation between these two heights and weight sequence. In order to find the number of concordant pairs, we can process each element in the sequence height one by one:

- The first element of height sequence is 4, notice that only 3 elements {1, 2, 3} come after 4 in both sequences, so there are three concordant pairs starting with 4:(4, 1), (4, 2) and (4, 3).
- The second element of height sequence is 3, in both of the sequences the 1 and 2 come after 3. So there are two concordant pairs starting with 3: (3, 1), (3, 2).
- The third element of height sequence is 1; there is no concordant pairs starting with 1 since there is no element in weight sequence after 1.
- The fourth element of height is 5; the only concordant pairs start with 5 is (5, 2) since 2 comes after 5 in both sequences.
- The fifth element of height sequence is 2 since there is no element after 2 in this sequence; there are no concordant pairs starting with 2.

The set of all concordant pairs is, P_{con} = {(4, 1), (4, 2), (4, 3), (3, 1), (3,2), (5,2)} And the total number of concordant pairs is:
P = 3 + 2 + 0 + 1 + 0 = 6.

The corresponding Kendall Tau relation value is:

$$\tau = \frac{4 * 6}{5 * (5 - 1)} - 1 = \frac{6}{5} - 1 = 0.2$$

This result indicates an agreement between the rankings since it is above 0. However, it is not as strong as expected since the agreement is 60%.

3.4.2 Experimental Setup and Results

3.4.2.1 Experimental Setup

Data Set

We compared 5 different versions of the page-rank algorithm with respect to different metrics over Turkish Web having 50 million webpages and more than 200 million links. In each experiment, we used 5 popular categories {shopping, news, regional, sports and general} and Alexa URL list as gold standard in order to compare 5 different page rank implementations. In each category, we used the top 10 URLs from Alexa list and we calculated the Kendall Tau correlation results of each method with Alexa's lists. The important property of these 5 categories is that all of the methods give positive correlation with the gold standard list.

What is Alexa?

Alexa is a Web information company collecting and providing the Web traffic information. Alexa's information source is Alexa Toolbar users. Alexa toolbar works integrated with compatible Web browsers. The toolbar collects users' browsing information while they surf. The browsing data consists of two types of information: *Reach* and *PageView*. *Reach* is the number of unique Alexa users who visit a site on a given day. *PageView* is the total number of Alexa user URL requests for a given site. *PageView* determines how many unique pages are requested from the given site by unique users in a given day.

However, there are some concerns about the accuracy of Alexa. Alexa collects information over its toolbar and the toolbar is only compatible with the Internet Explorer, Firefox and Chrome Web browsers. Clearly, Alexa rankings lack the usage data of those who do not use one of these three browsers. Moreover, Alexa reports that some local websites with low Reach and PageView counts may not be ranked accurately. The scope of our study covers the Turkish Web domain, and we are only interested in the top 200 websites from Alexa's list. In this regard, the above concerns on the accuracy of Alexa rankings are out of our consideration.

Alexa provides a useful seed list of trustable pages for the TrustRank algorithm. In the ideal case of spam page elimination, all pages are given input to the oracle function. The oracle function represents human evaluation on a page p; that is, if

$O(p)$ equals 1, then p is a good page, and p is a spam if $O(p)$ equals zero. In this regard, we can see the Alexa top ranked list as a huge set of pages on which the oracle function is applied by millions of users.

3.4.2.2 Experimental Results

In the experiment part, we evaluated the performance of each PageRank variant shown in Table 3.2. Our performance metric is the distance value of the PageRank results against the Alexa rankings on each category. We compared the average Kendall Tau distance of the first ten URLs of each case (PageRank variant) with Alexa's URLs. We interpreted the experimental results in two different ways. Firstly, we calculated the Tau distances for each category. Then, the average Tau distances were calculated for each PageRank variant. Secondly, we calculated weighted distances to see a clearer characteristic of each PageRank variant. The next section analyzes the results in a more detailed way.

Average Kendall Tau Distances with Alexa

In the first experiment, we compared the average Kendall Tau distance of the first ten URLs of each case with Alexa's URLs. The result of the experiment is given in Figure 3.3.

Case-1 shows the performance of the baseline algorithm of PageRank. In this case, no link weighting strategies and no spam page elimination is applied on the Web graph.

In Case-2, the edges (links) between pages within the same subdomain are prohibited in the Web graph. For instance, a hyperlink from *accounts.google.com/signIn* to *accounts.google.com* is given the weight of zero so that there will be no PageRank score contribution through that link.

In Case-3, again inlinks are eliminated from the Web graph. Here, we also eliminated the pages that could not be fetched. These unfetched pages are generally spam pages with too many redirections or the pages that no longer exist. Excluding these factors leads to a significant improvement in the PageRank performance. Like in Case-1 and Case-2, no spam page elimination is applied on the Web graph.

In addition to the features of Case-3, in Case-4, we decrease the outlink weights between two subdomains of a given host. For example, if *www.google.com* gives link to *mail.google.com*, the link weight is decreased from 1.0 to 0.01. For the first time, spam pages are filtered out of the Web graph in this variant.

In Case-5, in addition to features of Case-4, we give penalty to links (URLs) which serve for the same IP. Like Case-4, spam page elimination is applied.

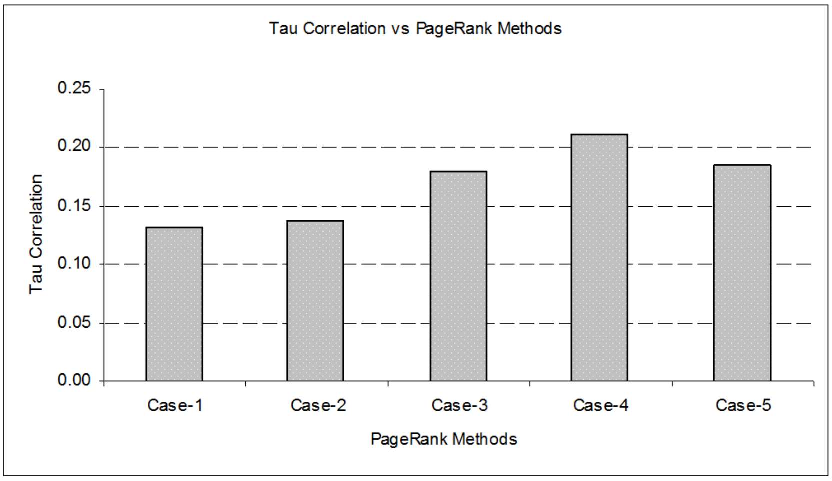


Figure 3.3: PageRank Cases versus corresponding average Kendall Tau distances against Alexa’s top lists.

Weighted Distances with Alexa

In this interpretation of experimental results, we compared each method with respect to the weighted score. Weighted score is defined to map each method into one of the integer values in the set {1, 2, 3, 4, 5} with respect to their correlation ranking. The method producing the most correlated results with Alexa's gets 5 points and the next one gets 4, and so on. We calculated the average score of each method over the results for the top 5 categories. The bars in Figure 3.4 show weighted score of each case.

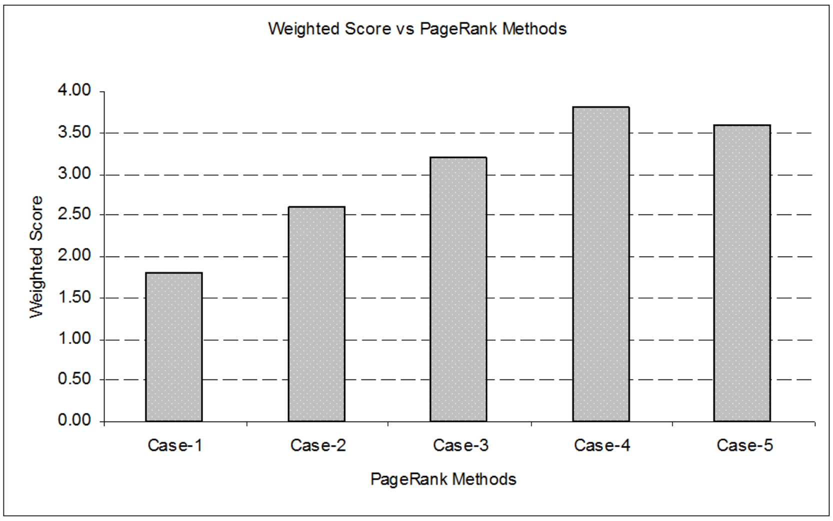


Figure 3.4: PageRank Cases versus corresponding weighted scores against Alexa lists.

In this interpretation, the aim is to see the distinctive characteristics of different PageRank cases more clearly. As seen in Figure 3.4, Case-4 outperforms the other PageRank variants by combining our weighting scheme and spam page elimination with modified TrustRank algorithm. We observed in Case-5 that eliminating multiple URLs serving for the same IP does not yield better results in the Turkish domain.

A sample of Kendall Tau distances and the corresponding mappings with weighted scores is shown in Table 3.4.

The important result of our experiments is that the methods using external usage information like trust-rank with Alexa as its seed list produce better results than others. In both Case-4 and Case-5 we have used external usage information about set of URLs which corresponds to trusted set of webpages. Our results show that this type of external usage information improves the results of ranking.

Table 3.4: An example showing the Kendall Tau distances and their corresponding integer mappings.

	PageRank Case-1	**PageRank Case-2**	**PageRank Case-3**	**PageRank Case-4**	**PageRank Case-5**
General	Distance: 0.11 Mapping: 1	Distance: 0.12 Mapping: 2	Distance: 0.21 Mapping: 3	Distance: 0.36 Mapping: 4	Distance: 0.37 Mapping: 5
Regional	Distance: 0.14 Mapping: 2	Distance: 0.13 Mapping: 1	Distance: 0.19 Mapping: 3	Distance: 0.30 Mapping: 5	Distance: 0.27 Mapping: 4
News	Distance: 0.12 Mapping: 1	Distance: 0.21 Mapping: 3	Distance: 0.20 Mapping: 2	Distance: 0.34 Mapping: 5	Distance: 0.32 Mapping: 4
Sports	Distance: 0.09 Mapping: 1	Distance: 0.10 Mapping: 2	Distance: 0.19 Mapping: 3	Distance: 0.29 Mapping: 4	Distance: 0.34 Mapping: 5
Shopping	Distance: 0.15 Mapping: 1	Distance: 0.15 Mapping: 2	Distance: 0.22 Mapping: 3	Distance: 0.40 Mapping: 5	Distance: 0.32 Mapping: 4
Overall	Avg. Distance: 0.12 Weighted Score: 1.2	Avg. Distance: 0.14 Weighted Score: 2.0	Avg. Distance: 0.22 Weighted Score: 2.8	Avg. Distance: 0.34 Weighted Score: 4.6	Avg. Distance: 0.32 Weighted Score: 4.4

We observe that the Case-4 proves the best performance amongst all of the cases. This result suggests that the sample Web model we depicted in Figure 3.2 is the ideal Web graph providing the best performance over our experimental setup.

3.5 Conclusion

In this study, we addressed two main problems that search engines encounter today: the elimination of spam pages and the difficulty to create a Web model revealing the true value of hyperlinks between pages. The experimental results suggest that refining the link weights on a trusted set increases the PageRank performance remarkably. The cooperative work of PageRank and TrustRank algorithms is another key point in this success. With the integration of the usage data of real Web surfers, TrustRank provides the PageRank algorithm with a safer environment to work on.

The WPR weights the links by the importance of the pages that they point to. In this approach, important pages get more PageRank scores. In the WLPR, weights of links are determined by their relative position and textual structure in a webpage. The study by Ding et al.(2004) covers reshaping of link normalization scheme by taking indegree values of pages into consideration along with outdegrees. Our approach differs from these studies by concentrating on the hierarchical relationship between pages focusing on their host and subdomains. Therefore, we can say that our work is mutually exclusive with the previous work on the PageRank link weighting scheme and combinations of these work may be evaluated to improve PageRank accuracy.

Our study also offers an alternative and efficient way of selecting a trusted seed list from real users' surfing experience that helps to eliminate spam pages.

However, the Web is a very complex structure. In this study, we conducted the experiments on a usage data which covers Turkish users' surfing experience. Therefore, the crawl data were restricted to the Turkish domain except the foreign pages as leaf nodes in the Web graph. In this regard, our results may not be applicable to other domains due to different navigational and hierarchical structures of websites from different cultures and domains. Different link topologies might require different link weighting schemes.

CHAPTER 4

EFFECTS OF STEMMING IN INFORMATION RETRIEVAL PRECISION SCORE AND WEB RELEVANCE FOR TURKISH WEB SEARCH

In many Information Retrieval (IR) tasks, stemming is applied to transform words to their root forms before indexing. Applying stemming increases the retrieval effectiveness; however, it lowers the Web relevance (i.e. whether the list of returned webpages is relevant with user query or not, for Web search). Stemming requires additional processing resources at indexing and querying time; therefore, extra computational requirements shall be satisfied to avoid a decrease in the overall performance of the Web search engine.

In this study we propose a framework that increases IR precision score and Web relevance by using a stemmer/lemmatizer for Turkish Web search engine domain. We first show that for Turkish, which is a morphologically rich language, applying stemming increases retrieval effectiveness while decreasing Web relevance. We also show that the best precision and Web relevance pair is obtained by combining the results, retrieved by sending unstemmed query to unstemmed content and sending stemmed query to stemmed content.

Another contribution of this study is that we adopt the MapReduce programming model for the stemming process during the indexing phase of the webpages. We show that Mapreduce overcomes the performance overheads of the stemming process.

4.1 Introduction

Web search engines are used as a major tool in daily life for gathering information by showing a list of hits to users. Users spend time following links, reading a page to check their relevance, trying other suggested pages or searching by means of improved queries, and the process goes on like this. Finding relevant pages is usually time consuming and difficult for users. One of the most important issues of a Web search engine, i.e. effectiveness measure, is that user queries are most of the time not best formulated to get optimal results.

Accurate formulation of a query requires users to predict the word form used in the documents that best suit their actual needs, which is tedious even for experienced users (Peng et al., 2007). One of the widely used techniques to solve this problem is stemming of document and query terms (Porter 1980), which is the process of mapping singular and plural forms of the same word on a single stem so that a search term will match documents that contains all forms of the term. Kraaj and Pohlmann (1996) showed that traditional stemming increases recall by matching word variants. However, stemming may reduce unrelated words to the same stem and it may fail to reduce related words to a common stem, which results in stemming to reduce Web relevance.

Another problem with stemming is the performance overheads. Stemming requires additional computation resources while search engine stems all of the words to their root forms within the content of webpages indexed (which ends up millions of webpages).

To increase the IR precision score for Web search we use a language specific lemmatizer-based stemmer (for Turkish). To avoid the negative effects of stemming on Web relevance, we propose an innovative method for the querying of content: retrieving half of the results from sending unstemmed query to unstemmed content and the rest from sending stemmed query to stemmed content. In order to overcome the performance problem we use the MapReduce paradigm with distributed computing architecture.

In the next section, we summarize the working principle of some Turkish stemmers and the other methods developed to improve IR effectiveness by stemming the documents and query terms in the literature. In Section 3, we continue with the detailed descriptions of the Turkish stemmer/lemmatizer and the methods used in Web search engine to have the best precision score and Web relevance pair. Section 3 concludes with the work done for the stemming performance problem. In Section 4, we report the results of perform extensive experiments to support our claim via detailed discussions. Finally, in Section 5, we conclude the chapter by giving future suggestions.

4.2 Related Work

Stemming is a commonly used technology for many applications and domains, such as IR, domain analysis and commercial products. One of the most widely used stemmers in IR is the Porter stemmer (Porter, 1980) due to its simplicity and effectiveness. In most query systems, such as Web search engines, stemming is used to increase IR effectiveness (Hull, 1996). The effects of stemming on IR effectiveness are analyzed for different languages. Figuerola et al. (2006) showed that IR effectiveness by using stemmer could not improve for Spanish. However, for German (Braschler and Ripplinger, 2004) and English (Hull, 1996; Krovetz, 1993) the stemming has been reported to have positive effects on IR effectiveness.

For Turkish, a number of IR studies have taken place to analyze the stemming effects. Solak and Can (1994) analyzed the effects of stemming on Turkish text retrieval and showed a small amount of improvement in retrieval effectiveness by showing precision and recall values. Similarly, Sever and Bitirim (2003) discussed analysis and evaluation of a Turkish stemming algorithm and concluded that stemming improves retrieval effectiveness.

Ekmekçioğlu and Willett (2000) compared the retrieval effectiveness using the stemmed and unstemmed query terms without stemming documents. They concluded that by using stemmed words with unstemmed content, retrieval effectiveness is increased.

Analytic languages like English, have limited morphological forms of words than agglutinative languages like Turkish. Therefore, analytic languages' stemming process is relatively unsophisticated. On the other hand, it is a difficult task to stem words for agglutinative languages due to their rich morphological forms and syntactic relations between words or concepts through discrete suffixes and complex word structures. For example, Turkish an agglutinative language, has words derived by using inflectional and derivational suffixes linked to a root. An example can be seen in the following table:

Table 4.1: A sample Turkish word and its derivations.

Turkish	English
araba	car
araba-m	my car
araba-lar	cars
araba-lar-ım	my cars
araba-lar-ım-da	in my cars

For Turkish, one of the developed Turkish stemmer is available under Zemberek[10,] which is an open source Natural Language Processing (NLP) framework for Turkic languages. Zemberek aims to provide a generic NLP framework not only for Turkish but also for other Turkic languages. The framework provides basic NLP operations

[10] http://code.google.com/p/zemberek/

like stemming, spell checking, word suggestion, converting words written only using ASCII characters and extracting symbols. In Zemberek, for stemming, a morphological parser basically finds the possible root and suffixes of a given word. Dinçer and Karaoğlan (2003) developed a probabilistic stemmer for Turkish and they claimed that it may also be applied to other agglutinative languages, such as Finnish, Hungarian, Estonian and Czech.

Can et al. (2006) did large scale IR experiments on Turkish texts by implementing and applying simple to sophisticated stemmers and various query-document matching functions. They concluded that truncating words at a prefix length of 5 creates an effective retrieval environment for Turkish. Besides, they showed that lemmatizer-based stemmers provide significantly better effectiveness over a variety of matching functions.

Krovetz (1993) showed a precision up to 45% by comparing a small number of documents (from 400 up to 12K). However, Krovetz used limited number of tested queries and the collected data were not large enough. Therefore, the results cannot be generalized for Web search.

Velez et al. (1997) used query expansion and Anick (2003) used query reformulation to improve the performance of stemming in query phase. Generally, these two techniques aim to reformulate or expand the query by using feedback techniques. These techniques include sending query to a Web search engine and retrieving top relevant documents. After retrieving the top documents, query is reformulated or expanded and sent back to the search engine. Therefore, this requires sending a query multiple times to a search engine, and it is time consuming to process huge amount of queries in a Web search. Besides, query expansion and reformulation may change the intent of the original query (Peng et al., 2007).

4.3 System Detail

4.3.1 Overview

The system has been fed by a focused crawler that is specialized to gather Turkish webpages and has language identifier capability. The parser and the indexer component are responsible for parsing the crawled pages to their pure text forms, and indexing them as unstemmed and stemmed forms via inverted index structure. To have stemmed content Turkish a stemmer/lemmatizer component is used. The proposed system architecture shall be seen in Figure 4.1.

When a user query comes to the search engine, the top results are retrieved from both sending unstemmed queries to unstemmed content, and stemmed queries to stemmed content index in an evenly distributed manner. Half of the results are retrieved from unstemmed content, and the other half is retrieved from stemmed index. Query is stemmed by using the Turkish stemmer/lemmatizer, too. Finally, the results are shown from both contents.

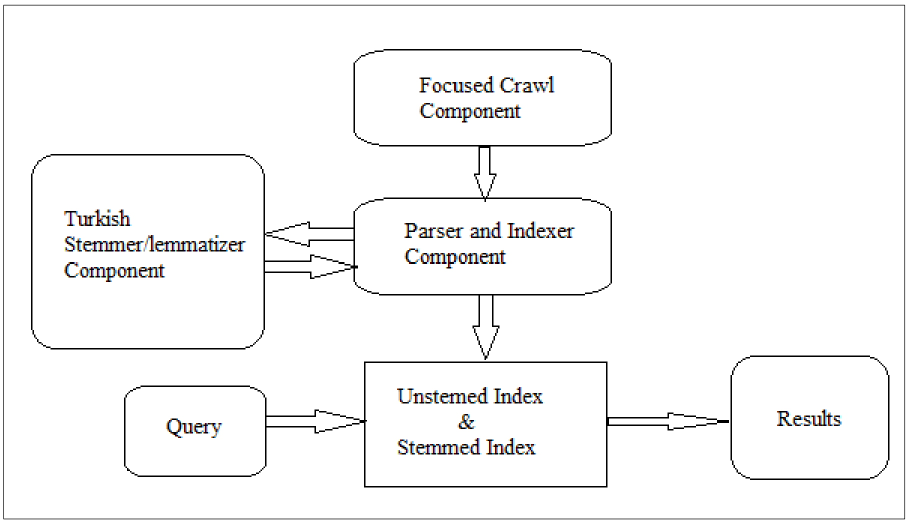

Focused Crawl Component
Turkish Stemmer/lemmatizer Component
Parser and Indexer Component
Query
Unstemed Index & Stemmed Index
Results

Figure 4.1: System overview.

4.3.2 Bilgi.com Search Engine

Throughout this study, the required tasks are implemented on the top of a Web search engine, i.e. Bilgi.com[11], basic components. Bilgi.com, which is funded by the Scientific and Technological Research Council of Turkey (TUBITAK) as a research project, is a scalable, commercial Web search engine that is localized for the Turkish Web domain, such that it has specialized modules for the Turkish language. These modules provide Turkish NLP operations like spell checking, language identification, morphological analysis, sentence boundary detection, etc.

As for ranking quality, Bilgi.com uses variants of PageRank algorithm (Page et al., 1999) that are specialized for the Turkish domain. In this study, although different ranking algorithms are available in Bilgi.com, since the main goal is to measure IR effectiveness and Web relevance, Term Frequency (TF) - Inverse Document Frequency (IDF) is used to calculate the IR scores. Besides being a commercial search engine Bilgi.com provides a research environment for this study.

4.3.3 Turkish Stemmer/Lemmatizer

Stemming is the process of removing inflectional elements and certain other endings from a word (Porter, 1980). For a stemming algorithm, the resulting form is not guaranteed to be a word. Lemmatization, on the other hand, is the process of finding the most basic linguistically valid form of a word. As specified earlier, Turkish is an agglutinative language with a productive inflectional morphology. In Turkish, a word may, theoretically, have an infinite number of inflected forms and typically have 10-20 different forms in a moderately sized corpus.

It has been reported that lemmatizer–based Turkish stemmer provides better effectiveness over a variety of different stemmers (Can et al., 2006). In order to

[11] http://www.agmlab.com/AGMLab_eng/bilgi_com.html

increase IR effectiveness a stemmer and a lemmatizer for Turkish is adapted for our study. Both implementations use a morphological model based on Oflazer (1994). The morphological model combines phonological variations and morphotactics as single deterministic finite automata. The resulting automaton is a recognizer for Turkish morphology. Basic properties of the stemmer are as follows:

- No lexicon is required.
- The morphological model is used in reverse to get a reduced form.
- Certain stem voice changes can be handled.

Basic properties of the lemmatizer are as follows:

- A lexicon is used to determine basic forms of the items.
- The morphological model is used together with lexical categories to determine the correct form of the stem.
- Most of the stem voice changes are handled correctly before the transition to the morphological model.
- When there is an ambiguity (e.g. adam, ada+m) the longest stem is preferred.
- When a stem cannot be determined, the whole form is returned as the stem.
- It can process around 100K words per second.

Here is a sample output for the Turkish lemmatizer:

Table 4.2: Sample outputs of Turkish stemmer.

Word	**Stem**
beklenen	bekle
bekleniyor	bekle
bekleyen	bekle
bekliyor	bekle
adım	adım
adına	ad
Adını	ad

The lemmatizer does not handle certain types of exceptional stem form changes, such as hakkı, yiyor, etc.

4.3.4 Turkish Lexicon

A Turkish lexicon is used within stemmer/lemmatizer implementation. This lexicon is created based on the entries gathered from the TDK's[12] (Türk Dil Kurumu, Turkish Language Institute) lexicon. The lexicon is stored as the Open Lexicon Interchange Format (OLIF[13]) file having nearly 96,000 entries.

The structure of an OLIF entry in the body of an OLIF file accordingly reflects a hybrid representation, with neither the explicit lemma-orientation of many lexicons, nor the explicit concept-orientation (with formal concept and term levels) of many terminology management models. To accommodate both the lexical and terminological models, the OLIF developers have opted for a flexible structure based on word-sense orientation

The OLIF word sense is itself defined as a semantic unit that is identified uniquely by a set of five key data categories:

- Canonical form: The entry string, represented in canonical form in accordance with OLIF guidelines.
- Language: The language represented by the entry string.
- Part of speech: The part of speech, or word class, represented by the entry string.
- Subject field: The knowledge domain to which the lexical/terminological entry is assigned.
- Semantic reading: The semantic class identifier used to distinguish readings for entries with identical values for canonical form, language, part of speech, and subject field.

[12] http://www.tdk.gov.tr/

[13] http://www.olif.net/index.htm

Generated sample OLIF file is as follows:

Table 4.3: Generated sample OLIF file.

```
<entry>
    <mono MonoUserId="113741348764324328092">
        <keyDC>
            <canForm>kireçsizleştirmek</canForm>
            <language>tr</language>
            <ptOfSpeech>verb</ptOfSpeech>
            <subjField>general</subjField>
            <semReading>86</semReading>
        </keyDC>
        <monoDC>
            <monoAdmin>
                <entrySource>TDK</entrySource>
            </monoAdmin>
            <monoMorph>
                <inflection>agmlab-default</inflection>
            </monoMorph>
            <monoSem>
                <definition>Kireçsiz duruma getirmek.</definition>
            </monoSem>
        </monoDC>
        <generalDC>
            <updater>guven.fidan@agmlab.com</updater>
            <modDate>2006-01-16T14:11:27.643+02:00</modDate>
        </generalDC>
    </mono>
</entry>
```

4.3.5 IR scoring Calculation

In this study, we use Lucene[14], which is an open source IR library that has inverted index structure, for indexing documents. It is shown that inverted index is the state-of-the-art data structure for efficient retrieval (Zobel, 2006). To measure IR scores, we use Lucene's TF-IDF based scoring models. In this model, Boolean Model[15] (BM) and Vector Space Model[16] (VSM) are combined in such a way that the queries and the documents are represented as weighted vectors, and the weights are TF-IDF values. The scoring calculation is as follows:

$$score(q,d) = coord(q,d) \times queryNorm(q) \times \sum_{t\ in\ q}(tf(t\ in\ d) \times idf(t)^2 \times termboost \times norm(t,d)) \quad \text{(Equation 4.1)}$$

where:

tf(t in d): Term frequency, the number of times term t occurs in document d.

$$tf(t\ in\ d) = \ frequency^{1/2} \quad \text{(Equation 4.2)}$$

*idf(t):*Inverse document frequency, the number of documents in which the term t appears.

$$idf(t) = 1 + \log(\frac{numDocs}{docFrequency+1}) \quad \text{(Equation 4.3)}$$

coord(q,d): A score factor based on how many of the query terms are found in the specified document.

[14] http://lucene.apache.org/
[15] http://en.wikipedia.org/wiki/Standard_Boolean_model
[16] http://en.wikipedia.org/wiki/Vector_Space_Model

queryNorm(q): Normalizing factor that used to make scores between queries comparable. It does not affect document ranking, it is a search time factor.

$$queryNorm(q) = \frac{1}{sumofSquaredWeights^{1/2}} \qquad \text{(Equation 4.4)}$$

where sumofSquaredWeights for query term is calculated as:

$$sumofSquaredWeights = queryBoost^2 \times \sum_{t\ in\ q} (idf(t) \times termBoost)^{\ 2}$$

(Equation 4.5)

termBoost and queryBoost: Search term boosts for terms and queries.

norm(t,d): Function that encapsulates some boost and length factors at indexing time. docBoost is document boost and fieldBoost is field boost. lengthNorm is computed when the document is added to the index in accordance with the number of tokens of this field in the document.

$$norm(t,d) = docBoost \times lenghtNorm(field) \times \prod_{field\ f\ in\ d\ named\ as\ t} fieldBoost$$

(Equation 4.6)

4.3.6 Scalability

Dinçer and Karaoğlan (2003) state that as the level of morphological knowledge use increases in a stemmer, the level of computational complexity increases to a point close to NP-Hard. Since the stemmer/lemmatizer used in this work uses morphological knowledge and stemming at indexing and querying time causes performance problem. To overcome performance problem we use MapReduce (Dean and Ghemawat, 2004) programming model for stemming the webpage's content while indexing. MapReduce, a programming paradigm first introduced by Google in 2004, expresses a large distributed computation as a sequence of distributed operations called map and reduce tasks on data sets of <key, value> pairs (Dean and Ghemawat, 2004).

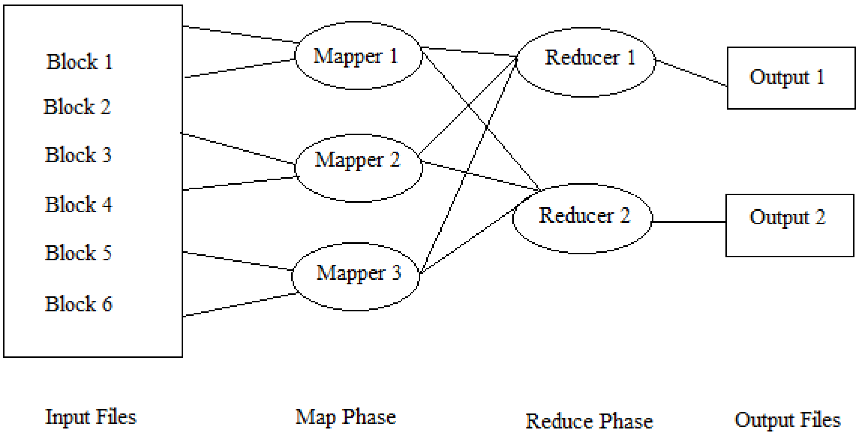


Figure 4.2: MapReduce programming model.

In our study webpage's content is stemmed in a distributed manner by partitioning the content into small blocks which are processed by different MapReduce nodes. Figure 4.2 shows the MapReduce processing flow. The input files are split into set of blocks and then processed in parallel by different map jobs. The map function stems the tokens within each document and emits a sequence of <stemmedWord, docID> pairs. The reduce function accepts all pairs for a given word, sorts the corresponding docID's and emits a <stemmedWord, list(docID)> pair. The set of all output pairs forms a simple inverted index.

The stemmer MapReduce job is tested on Bilgi.com's cluster having more than a hundred computers.

4.4 Experimental Evaluation

4.4.1 Evaluation Metrics

We have basically measured Web relevance by using IR metrics, namely Precision at K (Jarvelin and Kakelainen, 2000) and Normalized Discounted Cumulative Gain (NDCG).

Precision at K: It shows portions of documents ranked in the top K results that are labeled as relevant. In our case, we classify a page as PERFECT, EXCELLENT, GOOD, FAIR, and BAD. In order for a page to be considered relevant it must be classified as GOOD, EXCELLENT or PERFECT. By using this metric, we measure overall user satisfaction with the top K results.

Normalized Discounted Cumulative Gain (NDCG): It is a retrieval measure designed specifically for Web search evaluation (Jarvelin and Kakelainen, 2000). NDCG is efficient and effective to measure Web relevance. The advantages of NDCG can be given as follows:

- The degree of relevance of documents and their rank are combined.
- An estimate of the cumulated gain is given as a single measure no matter what the recall size.
- The cumulated gain is focused from the beginning of the results.
- The gain received through documents found later in the ranked results is weighted down.
- Modeling user persistence in examining long ranked result lists is allowed by adjusting the discounting factor.

NDCG is computed as follows for a given query (q), where the results are examined from top ranked to down:

$$Nq = Mq \sum_{j=1}^{K} ((2^{r(j)} - 1)/\log_2(1 + \mathrm{j})) \qquad \text{(Equation 4.7)}$$

where:

Mq: A normalization constant that is calculated such that a perfect ordering obtains NDCG of 1.

r(i): An integer label between 0 and 4. (Namely: 'BAD' = 0, 'PERFECT'=4) for results returned at position i.

It can be seen that BAD documents do not contribute to the sum, but reduces the NDCG for the query pushing down the relevant labeled documents. NDCG is a well suited metric to evaluate Web search results since relevant documents in the top ranked results are rewarded more heavily than those ranked lower.

Also, to show the performance gain by using the MapReduce programming model, we measure the time elapsed for stemming and indexing the content with different number of map tasks on a cluster of 4 computers.

4.4.2 Data Preparation

We run a focused crawler to collect Turkish webpages. These webpages are parsed and then indexed with both unstemmed and stemmed forms via inverted index structure. To index content as stemmed form, Turkish stemmer/lemmatizer is used. The total number of webpages indexed is 583,771 and about 3,2GB in size. The number of overall terms is around 16 million where 4,1 million of terms are indexed as unstemmed content and 3,78 million of terms are indexed as stemmed forms. No stop words removal is used during indexing.

In order to compare the performance of stemming with MapReduce programming model we run crawler several times with different number of map tasks.

For experimental setup, randomly 50 queries from a one month query log are selected. Among these 50 selected queries, all the misspelled queries and one word length queries are eliminated. The main reason of one word queries' removal is that they are not efficient at evaluating the Web search IR score and Web relevance, namely stemming one word queries has risk of changing query intent (Peng et al., 2007). Finally, 25 correctly spelled queries, and 18 among them are stemmed by using Turkish stemmer/lemmatizer. 7 of them are originally stem forms of words; therefore, they are not affected.

4.4.3 Experimental Setup

After having 25 correctly spelled queries, we sent unstemmed queries and stemmed 18 queries to indexes three times and retrieved top 10 ranked pages each time. Queries and their stemmed versions are given in Table 4.4.

The experiment setups shall be described as follows:

- In the first experiment, we retrieved results from indexes with unstemmed content.
- In the second experiment, queries are sent to the search engine again but results are retrieved from stemmed-content.
- In the third experiment, queries are sent to both stemmed and unstemmed index. Half of the results are retrieved from unstemmed content, and half of them from stemmed-content.

These three experiments are performed for unstemmed forms of queries and repeated for stemmed forms of queries in the same way. We did a final experiment to support our claim that retrieving half of the results by sending unstemmed queries to unstemmed content and half of them by sending stemmed queries to stemmed content gives best <Precision at K, Web relevance> pair. Overall, 7 experiments are conducted and the experiment setups shall be seen in Table 4.5.

Table 4.4: Queries used and their stemmed versions.

Query	Stemmed Query	English Version
Uzaktan kumanda	Uzak kumanda	Remote control
Son dakika haber	Son dakika haber	News flash
Tablet bilgisayar	Tablet bilgisayar	Tablet pc
Türkçe ingilizce sözlük	Türk ingiliz söz	Turkish English dictionary
Ankara yetkili servis	Ankara yetki servis	Ankara authorized service
Damacana su	Damacana su	Carboy water
Leyla ile mecnun	Leyla ile mecnun	Leyla and mecnun
Tek taş yüzük fiyat	Tek taş yüzük fiyat	Monolith ring price
Yılbaşı hediyesi	Yılbaşı hediye	New year gift
Bedelli askerlik	Bedel asker	Paid-for military service
Sayısal loto sonucu	Sayı loto sonuç	Lottery result
Online film izle	Online film izle	Watch online movie
Online radyo dinle	Online radyo dinle	Listen online radio
Hava durumu	Hava durum	Weather forecast
Iddia maç sonuçları	Iddia maç sonuç	Bets match result
Ucuz uçak bileti	Ucuz uçak bilet	Cheap plane ticket
Ankara sinemalar	Ankara sinema	Ankara movies
Tatil fırsatları	Tatil fırsat	Holiday opportunities
ODTÜ mezunlar derneği	ODTÜ mezun dernek	METU alumni club
Yılbaşı programları	Yılbaşı program	New year programs
Istanbul müzeleri	Istanbul müze	Museums of Istanbul
Galatasaray uefa kupası	Galatasaray uefa kupa	Galatasaray uefa cup
Telekomünikasyon hizmetleri	Telekom hizmet	Telecommunication services
Sigorta bayiileri	Sigorta bayii	Insurance company
Kurabiye tarifi	Kurabiye tarif	Cookie recipe

Table 4.5: Experiment setups.

Experiment Name	Query Type	Content Type
Experiment 1	Unstemmed	Unstemmed
Experiment 2	Unstemmed	Stemmed
Experiment 3	Unstemmed	Stemmed/Unstemmed
Experiment 4	Stemmed	Unstemmed
Experiment 5	Stemmed	Stemmed
Experiment 6	Stemmed	Stemmed/Unstemmed
Experiment 7	Stemmed/Unstemmed	Stemmed/Unstemmed

In all of the experiments, 10 different users ranked the top 10 results for the same queries. The subjects were selected randomly from METU[17] students who have familiarity with search engines like Google. 10 participants aged between 20 and 28 participated in the user study. Ranking part was to mark results as *PERFECT*, *EXCELLENT*, *GOOD*, *FAIR* and *BAD*. We use these user rankings to calculate Precision at K, NDCG values.

4.4.4 Results and Discussions

IR and Web Relevance

We summarize the overall results for unstemmed queries in Table 4.6 and the stemmed queries in Table 4.7. The results of retrieving half of the results by sending unstemmed queries to unstemmed content and half of them by sending stemmed queries to stemmed content is shown in Table 4.8. In all the tables, the first column depicts the name of the index type that the queries are sent to. The second column depicts Precision at K over all tested queries and the third column depicts the NDCG gain over all tested queries.

[17] http://www.metu.edu.tr

Table 4.6: Precision at K and NDCG results for sending unstemmed queries to different types of indexes.

Content Type	Precision at K	NDCG
Unstemmed Content	0.75	0.53
Stemmed Content	0.62	0.42
Mix Content	0.71	0.51

Table 4.6 shows the results for sending unstemmed queries to indexes. The best Precision at K and NDCG values are retrieved from unstemmed content. Sending unstemmed queries to stemmed content results in a **13%** decrease in the precision score and **11%** decrease in the NDCG value. Although these are statistically significant decreases, they are as expected, since the stemmed content may not match the words with suffixes in this case.

Table 4.7: Precision at K and NDCG results for sending stemmed queries to different types of indexes.

Content Type	Precision at K	NDCG
Unstemmed Content	0.64	0.36
Stemmed Content	0.86	0.48
Mix Content	0.77	0.44

Table 4.7 shows the results for sending stemmed queries to indexes. The best Precision at K and NDCG values are retrieved from stemmed content. Sending stemmed queries to stemmed content results in a **22%** increase in the precision score and a **12%** increase in the NDCG value over sending stemmed query to unstemmed content. Although these are statistically significant improvements, they are expected results too, since unstemmed content cannot match some word stems in this case.

Table 4.8 shows the results for sending unstemmed queries to unstemmed content, stemmed queries to stemmed content and mix results. The best precision score is obtained by sending stemmed queries to stemmed content; however, we can see that the NDCG, which corresponds to Web relevance, is lower than unstemmed content.

Sending stemmed queries to stemmed content results in an **11%** improvement in precision score and a **5%** decrease in the NDCG value over sending unstemmed query to unstemmed content. It shows that although stemming improves retrieval effectiveness for Turkish text it may reduce the Web relevance for Web search. By using mix content we have a **6%** increase in precision score and the NDCG value is decreased only by **1%** which is statistically insignificant. These results are supporting our claim which was improving IR precision score and avoid the decrease in Web relevance for Web search by retrieving search results from the mix-content.

Table 4.8: Precision at K and NDCG values for combining the unstemmed query sent to unstemmed content and stemmed query to stemmed content.

Content Type	**Precision at K**	**NDCG**
Unstemmed Content	0.75	0.53
Stemmed Content	0.86	0.48
Mix Content	0.81	0.52

Solak and Can (1994) and Sever and Bitirim (2003) showed that using stemming improves effectiveness for retrieval of Turkish texts. Our experiments show that using stemming increases IR score precision, which is conceptually in parallel with the findings of previous works, but decreases Web relevance (NDCG).

Stemming Performance

The results of distributed stemming of webpages are shown in Figure 4.3. Figure clearly shows that using the MapReduce paradigm reduces the stemming cost. Besides, we observed that increasing the size of the data processed by increasing the number of nodes in the cluster, the performance problem may be handled.

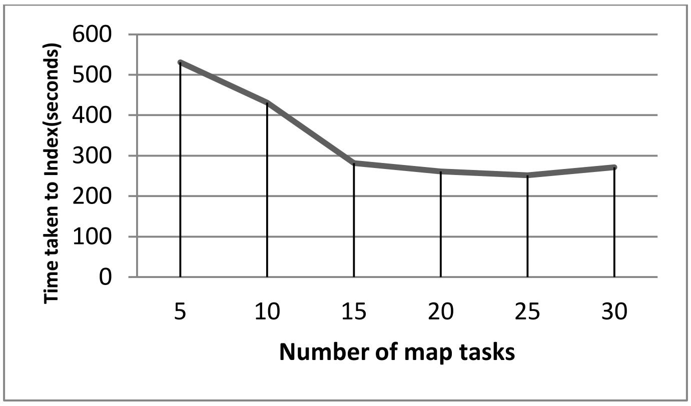


Figure 4.3: The effect of varying the number of map tasks over indexing time (seconds) of Turkish webpages.

4.5 Conclusion

We present a simple but efficient and effective way to increase IR score precision and to avoid a decrease in Web relevance for Web search by using stemming and we use the MapReduce paradigm to handle the performance problem during stemming. We show that by combining unstemmed and stemmed content, the best precision score and Web relevance pair is examined, and by using distributed programming, the stemming cost is reduced. For future work, the effect of long and short queries and the type of queries (informational, navigational, transactional) may be analyzed separately. Also, the query feedback mechanism may be handled with the processing of user clickthrough data, namely the query logs.

CHAPTER 5

ANALYZING USER BEHAVIOR ON THUMBNAILED WEB SEARCH

This study examines the effect of thumbnailed results on user behavior in Web search by interpreting users' eye tracking and clickthrough data. In this study, we used Bilgi.com[18] which is a domain-specific search engine operating on Turkish Web domain. Bilgi.com was chosen due to its capability of bringing thumbnailed search results. This study concentrated on finding the answer to the question: 'Can thumbnailed search results help users find the most relevant document on search engine results pages (SERPs)?' In order to answer this question, we added a mini snapshot of the corresponding webpage, called thumbnail, in each search result and we traced the eye movements and clicking behavior of the users to see if thumbnails have an effect on users' searching and decision making behavior.

Contrary to what was expected, a huge gap between gaze fixation duration on abstracts and thumbnails was observed. However, it does not mean that thumbnails prove no use in Web search. When the ranking quality on a SERP was intentionally decreased, users were observed to be able to find the most relevant results by using thumbnails without being able to find the most relevant results in the absence of thumbnails.

[18] http://www.agmlab.com/AGMLab_eng/bilgi_com.html

5.1 Introduction

The visual characteristics of a webpage may give ideas about the content and the quality of that page. Addressing this issue, we analyzed the impact of thumbnails on user behavior in Web search. In this study, we propose an analysis of the issue by examining user behavior on a search engine (Bilgi.com) which has the capability of bringing thumbnailed search results.

This is a study of how users interact with SERPs in both the presence and the absence of thumbnails. Specifically, we focused on two page components residing on a SERP. The first one is the thumbnail part and the other component is the abstract part of search results. When gathering eye tracking information, we ignored the gaze fixations on other page elements like related search or ads. The argument we had before starting this study was that thumbnails may help users evaluate search results more comprehensively so that they can find the most relevant search result.

In order to analyze whether thumbnails are useful components or not, we collected users' searching data to understand the actual user behavior. We collected the user behavior data using two methods: The first one was using eye tracking technique and the second one was processing user clickthrough data.

Eye tracking is an effective technique used in other search engine-user interaction studies as well (Joachims et al., 2005; Dumais, 2010; Ostergen, 2010; Buscher et al. 2010). In this method, a special device tracks users' eye movements by tracing the pupil-center using infrared light. The output of eye tracking is the gaze fixation durations on the defined components which are thumbnail and abstract elements in our study.

The second method we used was processing clickthrough data, namely the query logs. Clickthrough data tells us the final decision made by user on a SERP. Previous studies on search engine clickthrough data (Joachims, 2002; Radlinski et al., 2005) mainly concentrate on extracting a training data for learning algorithms to optimize search engine rankings. In our study, clickthrough data serves as relevance metric between a query and the corresponding search results assuming users make informed decisions proven by Joachims et al. (2005). The aim of their study is to devise a learning algorithm for training search engines on user clickthrough data to improve the ranking performance. They propose a technique to interpret the user clickthrough data into training data. The user behavior in his work is recorded as triplets including query, the suggested search results by search engine and the set of clicked results by users. Finally, a new set, the ideal set, is formed based on the clicked results set such that if users click the 1^{th}, 3^{rd} and 7^{th} results, then in the ideal set 3^{rd} result should be ranked higher than 2^{nd} result and 7^{th} should be ranked higher than 2^{nd}, 4^{th}, $5^{th}, and\ 6^{th}$ results for the given query. The intuition behind this approach is simple that a search engine should always rank the most relevant results higher since users cannot scan all the search results to find the most relevant ones.

In our study, we followed a similar technique to measure the thumbnailed search performance. We intentionally distort the quality of search rankings and analyze user behavior to see if thumbnails help users to find the most relevant search results. As the performance measure, we used Kendall's τ distance (Kendall, 1955) to find the distance between the ideal result sets of natural and distorted orderings based on user clickthrough data. For instance, the 10^{th} link is the most clicked one in the distorted set, and then it should be ranked as 1^{st} in the ideal set. The comparison of ideal sets helps us to analyze the effect of thumbnails in such a way that if the Kendall Tau correlation value between ideal sets of natural and distorted rankings is high (greater than 0 and closer to 1) then that means users are successful in finding the most relevant documents even if the ranking quality is low. The highest Kendall Tau correlation achieved in this study is achieved by correlation between the normal and reversed orderings in thumbnailed search.

Our results suggest that thumbnailed search is more useful when the quality of search rankings is extremely low. The eye tracking results show that users' spend most of their time on reading abstracts compared to gazing at thumbnails. However, this does not prove that thumbnails are not helpful for users in finding the most relevant results; the experimental results show that with the natural ordering of search results, the effect of thumbnails was not very absolutely clear because users usually clicked on the topmost ranked results. In both Joachims' (2002) and our study, the observed behavior agrees on that users have a trust bias towards search engine rankings. For instance, when the natural ordering of the search rankings was reversed (where the last result becomes the first and the first result becomes the last), the users were still inclined to click on the topmost search results. They thought that the topmost ranked documents were the most relevant ones even with the manipulated ordering. On the other hand, in thumbnailed search, when we intentionally distorted the ranking quality (reversal of natural ranking), the users were still able to find the most relevant result, which they weren't able to do so without the thumbnails.

5.2 Related Work

The previous work on user search behavior analysis concentrates on two basic questions: why and how do people search on the Web? Starting from the point of why people search on the Web, Broder (2002) introduced 3 types of search goals: informational, navigational and transactional search. In this taxonomy, informational search yields specific information, navigational search finds a specific webpage or site and transactional search seeks for a site to make transactions such as online gaming and shopping.

In another user behavior research, Downey et al. (2008) proposed a link between the commonality of a query and the success of a Web search such that popular queries yield more desirable search results for users.

User-computer interaction studies come into scene in understanding how people search on the Web, namely, how they react to and interact with search engines. Clickthrough log analysis is the starting point in analyzing user behavior. Joachims

(2002) presented an approach to create a training data on clickthrough data for learning retrieval functions. In this approach, the clickthrough data is considered as triplets consisting of query, the corresponding results set and the clicked result. Using SVM approach, he proposes a learning algorithm for clickthrough behavior. User behavior is converted into training data in such a way that the search result rankings are re-ordered by using the clicked results.

The eye tracking method, another popular method for analyzing user behavior, enables researchers to examine key insights of searching behavior. Joachims et al. (2005) examined the reliability of implicit feedback by interpreting eye tracking data and by direct judgment of clicking behavior by human evaluators (2005). They reported that users' choices are informed but biased that they are inclined to click on the topmost results disregarding the ordering of the results.

Inspired by eye tracking studies, Huang et al. (2011) argued that cursor movements correlate with the eye gaze. This method offers a wider range of users to analyze since eye tracking studies are only applicable on a limited number of participants. Another aspect that they evaluate with cursor movement analyzing is the abandonment issue. Sometimes users find the information they seek for on SERP so they do not need to click any results (good abandonment). Similarly, when users feel no interest in the results on SERP they left without clicking (bad abandonment).

Buscher et al. (2010) and Dumais et al. (2010) conducted eye tracking studies on SERPs to analyze the users' interest on a particular group of components like related search or ads.

Lagun et al. (2011) developed a system called ViewSer that allows only a single result to be seen at a time and blurs the rest of the search results. Similar to cursor position study, the goal of their study was to analyze large-scale remote user data on SERPs.

The rest of this study is organized as follows: the details about the user study we conducted, the analysis of the user data we get and finally the conclusions on the results we get.

5.3 User Study

In order to understand the effectiveness level the thumbnailed search, we analyzed user behavior while searching with and without thumbnails. Since we were interested in determining the influence of thumbnails in users' decision making process, we analyzed the amount of time users spent on individual search results as well as the percentage of time that they paid attention on thumbnails. Our interpretations of ordinary and thumbnailed search gave us key insights on the users' behavior of Web search under the influence of graphical enhancements. The next section describes the general structure and the scope of this study.

5.3.1 Task, Participants and Conditions

The task we assigned to participants was designed to test all the characteristics of our thumbnailed Web search. The participants were given 10 questions to search on Bilgi.com search engine. Since this study covered only the Turkish domain, the questions were selected so that the answers could be found in the Turkish Web domain. Considering Web search characteristics introduced by Broder (2002), we chose half of the questions to be navigational whereas the other half was informational. The complete list of these questions is presented in Table 5.1 and Table 5.2. The navigational questions asked the participants to find a specific webpage, and the informational questions asked them to reach a specific piece of information.

Table 5.1: Navigational questions.

Navigational Questions
Find the personal webpage of writer ‘Mehmet Yılmaz’.
Find the webpage showing Ankara subway map.
Find the home page of ‘Antalya Otium Hotel’.
Find the webpage of Beyazıt Öztürk’s TV Show.
Find the webpage showing accommodation information for METU students.

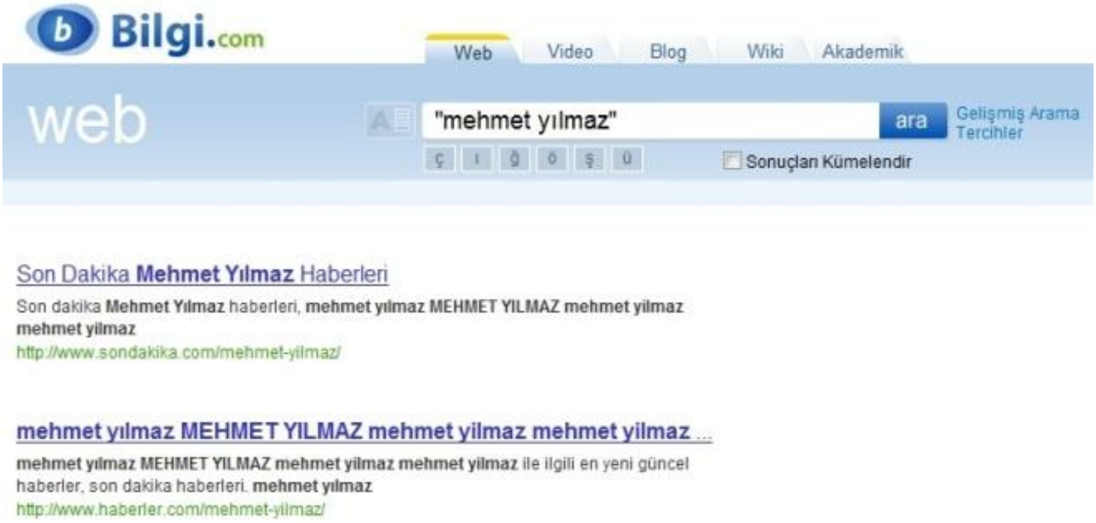
Table 5.2: Informational questions.

Informational Questions
Which is the largest dam lake in Turkey?
What is the date of the next local elections?
Who starred the character ‘Güdük Necmi’ in the first volume of movie ‘Hababam Sınıfı’?
Who carries on the ministry duty after the previous minister Erkan Mumcu’s resignation from his political party?
Who invented the hooked needle?

The participants were asked to answer the questions in the given order and they were told not to use any search engines other than Bilgi.com. They were informed that the researchers were interested in their behavior on SERPs. The subjects were also reminded that they were expected to search for the answer even if they knew the answer; it was explained to them that, the focus was not on the answer itself, but instead, on their behavior in searching for the answer. They were told that there were no restrictions on the queries they used, but they were asked not to open a new page for results.

The participants were selected randomly from METU students who have familiarity with search engines like Google. 28 participants (22 males and 6 females) aged between 20 and 31(Mean=24) participated in the user study.

Half of the subjects used ordinary search interface with no thumbnails. A snapshot of ordinary search interface is shown in Figure 5.1. The other half of the participants were presented a thumbnailed search interface shown in Figure 5.2.



Son Dakika **Mehmet Yılmaz** Haberleri

Son dakika **Mehmet Yılmaz** haberleri, **mehmet yılmaz MEHMET YILMAZ mehmet yilmaz mehmet yilmaz**

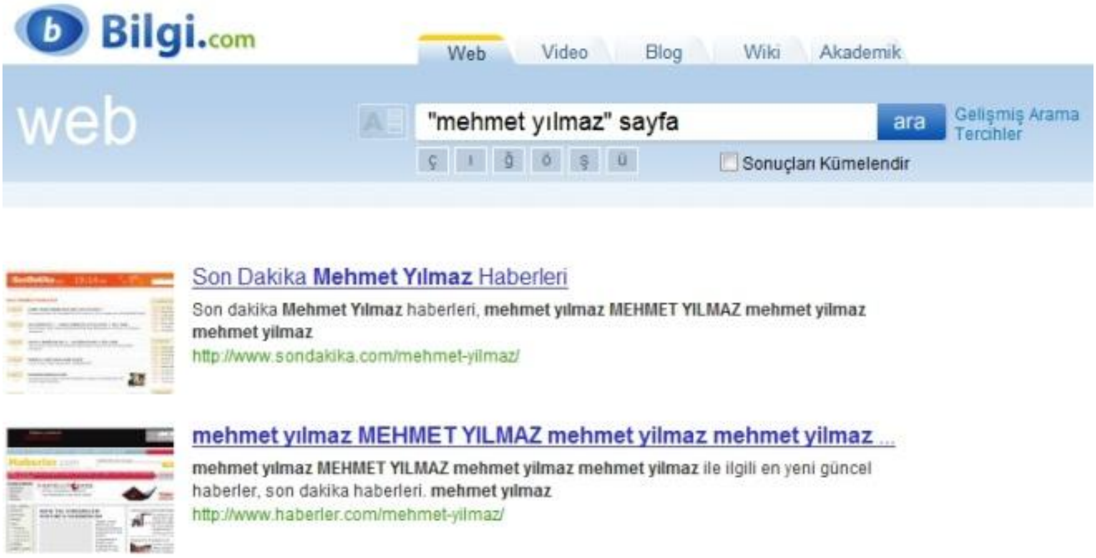
http://www.sondakika.com/mehmet-yilmaz/

mehmet yılmaz MEHMET YILMAZ mehmet yilmaz mehmet yilmaz ...

mehmet yılmaz MEHMET YILMAZ mehmet yilmaz mehmet yilmaz ile ilgili en yeni güncel haberler, son dakika haberleri. **mehmet yılmaz**

http://www.haberler.com/mehmet-yilmaz/

Figure 5.1: Bilgi.com search interface with no thumbnails.



"mehmet yılmaz" sayfa

Son Dakika **Mehmet Yılmaz** Haberleri

Son dakika **Mehmet Yılmaz** haberleri, **mehmet yılmaz MEHMET YILMAZ mehmet yilmaz mehmet yilmaz**

http://www.sondakika.com/mehmet-yilmaz/

mehmet yılmaz MEHMET YILMAZ mehmet yilmaz mehmet yilmaz ...

mehmet yılmaz MEHMET YILMAZ mehmet yilmaz mehmet yilmaz ile ilgili en yeni güncel haberler, son dakika haberleri. **mehmet yılmaz**

http://www.haberler.com/mehmet-yilmaz/

Figure 5.2: Bilgi.com search interface with thumbnails.

There were two phases in each interface to distinguish the user behavior between high and low-quality search rankings. In Phase I, the search results are in their natural order. In Phase II, the natural ordering of the results is manipulated by the search engine itself. The order was manipulated in two ways: swapping and reversing. Table 5.3 shows all the ordering configurations used in this study. The aim in manipulating the ordering of results was to assess users' reaction to low-quality rankings. In other words, we wanted to see if users made random selections on presented rankings or their clicks are informed.

Table 5.3: Different orderings on search result ranking.

natural ordering	The natural ordering of the results based on PageRank scoring is used.
swapped ordering	The first and the second results from the natural ordering are swapped.
reversed ordering	The first 10 results are put in the reverse order of their natural order.

The complete list of participants and the corresponding search conditions are given in Table 5.4. User and search condition matching was randomly selected.

Table 5.4: A complete list of participants

Subject #	Age	Sex	Thumbnail	Ordering
1	22	M	YES	Normal
2	23	M	YES	Normal
3	26	M	YES	Normal
4	27	M	YES	Normal
5	26	M	YES	Reversed
6	20	F	YES	Reversed
7	24	F	YES	Reversed
8	23	F	YES	Reversed
9	22	M	YES	Reversed
10	21	M	YES	Swapped
11	22	M	YES	Swapped
12	22	M	YES	Swapped
13	24	M	YES	Swapped
14	24	M	YES	Swapped
15	25	M	NO	Normal
16	22	F	NO	Normal
17	26	M	NO	Normal
18	24	M	NO	Normal
19	20	M	NO	Reversed
20	23	F	NO	Reversed
21	29	M	NO	Reversed
22	31	M	NO	Reversed
23	23	M	NO	Reversed
24	24	M	NO	Swapped
25	24	M	NO	Swapped
26	25	F	NO	Swapped
27	24	M	NO	Swapped
28	26	M	NO	Swapped

The next section continues with the explanation of our data collection process.

5.3.2 Bilgi.com Search Engine

Bilgi.com is a commercial Web search engine localized for the Turkish domain such that it has specialized modules for processing the Turkish language. We used Bilgi.com for this study because it has the capability of providing both ordinary (without thumbnails) and thumbnailed search results. Bilgi.com has a scalable architecture supporting Gecko[19] rendering. Gecko is a free and open source Web layout engine to render webpages. It is developed by Mozilla Foundation.

As for ranking quality, Bilgi.com uses variants of PageRank algorithm that are specialized for the Turkish domain. For an increased ranking quality, Bilgi.com uses spam page elimination algorithms as well as providing an interface for manual spam page detection. In this study, the different orderings (both natural and manipulated) on rankings are handled by Bilgi.com; no external proxy interceptors are used.

Besides being a commercial search engine Bilgi.com provides a research environment for this study. Thumbnailed search is one of the research-in-mind features. In addition to visual quality features, there are other features implemented for studying ranking quality, retrieval performance, natural language processing for Turkish, clickthrough logging and training on implicit user feedback.

5.3.3 Data Capture

In this study, we captured two types of user data: the eye tracking and the clickthrough data.

[19] https://developer.mozilla.org/en/Gecko

The eye tracking method enables us to examine two types of information. The first one is the percentage of time that users spend on reading abstracts and assessing thumbnails of the results presented by the search engine. The second type of information we get from the eye tracker is the percentage of time that users spend on individual search results.

The clickthrough data helps interpret the final decision of the users on the results. What we sought for by analyzing clickthrough data was the agreement between gaze movements and the final decision of participants. If the users were to make informed decisions on presented search results, then there should be an agreement between their eye tracking and clickthrough data; that is, the more a result is gazed, the higher its probability to be clicked will be. Such kind of a consistency is the key factor in interpreting implicit user feedbacks. Previous work (e.g., Joachims et al., 2005) suggests that there is such kind of relevancy between these two types of user feedback.

In data capturing process, we started by defining the areas of interest (AOIs), the visual components on a SERP on which we seek for user interactions either by gazing or by clicking.

After the AOIs were defined the Tobii's ClearView[20] software extracted the focus and click rates for the defined AOIs. By the end of the user survey, we had 1600 snapshots, 30 videos containing 15766 AOIs detected by ClearView. In the next section, the AOIs are explained in more detail.

AOIs

In this study, we defined two AOIs: the thumbnail and the abstract parts of search results. There are several AOIs defined in search engine studies. In some studies, an AOI is just the abstract part of search results (Lagun et al., 2011). In other studies,

[20] http://www.tobii.com/en/eye-tracking-research/global/

advertisements or related search sections of SERPs are considered as AOIs (Buscher et al., 2010; Dumais et al., 2010). Even a whole page can be an AOI if the aim of the study is to examine user behavior on different graphical patterns enhancing SERPs (Ostergen et al., 2010). The AOI selection we made was sufficient in analyzing the impact of thumbnails and the relationship between thumbnails and corresponding abstracts.

Eye Tracking

The eye tracking experiments were performed in The Human Computer Interaction Laboratory of Middle East Technical University[21] as shown in Figure 5.3.

Figure 5.3: A view from METU HCI Laboratory.

The recordings were done using Tobii commercial eye tracker setup. The setup consists of an infrared eye tracking hardware and its software component named ClearView. ClearView is a gaze analysis software that is able to convert raw data from eye tracker into statistical data. The statistical data is AOI-oriented; namely, the

[21] http://www.metu.edu.tr

finalized data shows gazing information on defined AOIs like which part of the page is gazed for how long.

Clickthrough Logging

Clickthrough logging is handled by the ClearView software. Bilgi.com brings 10 search results on each SERP. Each click on a SERP is labeled as either ‘content’ or ‘linkN’ where N may take any value between 1 and 10 inclusively. The content links are the internal navigation links of the search engine so we ignore them. What we were interested in the clickthrough data was the click counts of the search result links labeled from link1 to link10. To eliminate content links and gather the actual click counts, we performed some basic text filtering operations on the clickthrough data to remove unnecessary logging information.

The next section continues with the interpretations on the data we collected.

5.4 Analysis of User Behavior

In order to distinguish the impact of thumbnailed Web search, we had to understand user behavior on ordinary Web search first. In this regard, first the ordinary search results were evaluated to generalize users’ inclinations on a SERP.

In his study, Joachims (2002) defined the term, ‘ideal ranking’. Simply, the ideal ranking is re-ordering of the natural search engine rankings considering clickthrough data. For instance, for a given query, imagine that the search engine brings 5 links ranked as link#1, link#2, link#3, link#4, and link#5. After analyzing user clickthrough data, we realize that from the most clicked link to the least clicked one the links are ordered such that link#3, link#1, link#2, link#5 and link#4. And assume that we decide to re-rank the search results based on This ordering is what we call ideal ranking of documents in a way that the most relevant links should be ranked higher. After gathering experimental data, we re-ordered the rankings to get the ideal rankings for normal, swapped and reversed orderings. After we get the ideal rankings, we compared the Kendall Tau correlation (1955) values between the normal vs. swapped and normal vs. reversed rankings ordinary and thumbnailed search.

5.4.1 Searching Without Thumbnails

The user behavior we observed without thumbnails is in consistency with that in the study conducted by Joachims et al. (2005). They observed that users make informed but biased choices among ranked results.

On one hand, the number of clicks shows that, whether the rankings were manipulated or not, the topmost ranked documents were considered to be the most relevant search results by the participants. On the other hand, the click count of 10^{th} result increases significantly when the ordering is reversed. The click count for each ranking is given in Figure 5.4.

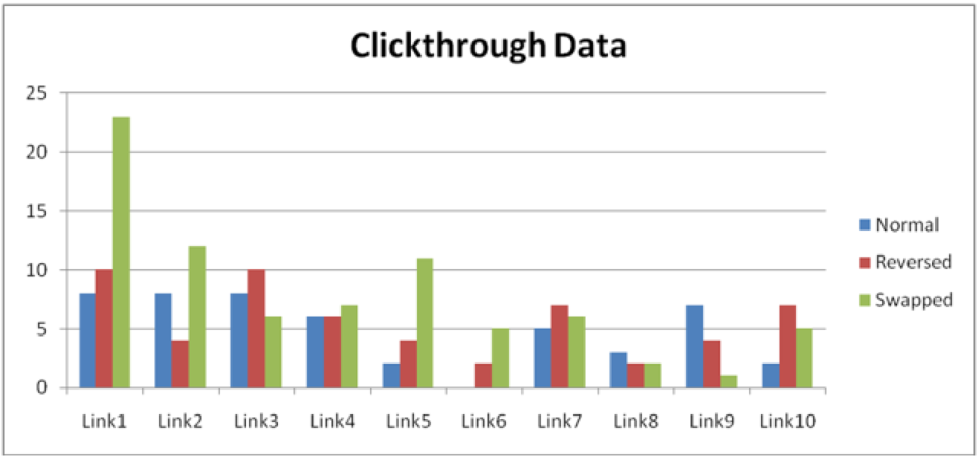


Figure 5.4: Clickthrough data of differently ordered result sets with no thumbnails.

Apparently, the searching behavior is informative but biased. Namely, the users do read and evaluate the abstracts before clicking on the links, but they also trust the search engine rankings in any ways that they are inclined to click on the top ranked documents in all cases.

In order to compare the previous results with thumbnailed search, we analyzed the results of different ordering by calculating the Kendall Tau correlation between them. Since the main concern of the comparison is to understand user behavior, we first obtain the ideal sets based on clickthrough data. As shown in Figure 5.4 we have 3 different orderings. First we sort the links according to their click rates. Table 5.5 shows the ideal rankings for normal, swapped and reversed result sets.

Table 5.5: Ideal rankings for ordinary search.

normal	1	2	3	9	4	7	8	10	5	6
swapped	1	2	5	4	3	7	6	10	8	9
reversed	1	3	10	7	4	2	5	9	6	8

The Kendall Tau correlation calculation is given in Equation 5.1.

$$\tau = \frac{2\mathrm{P}}{\frac{1}{2}\mathrm{n}(\mathrm{n}-1)} - 1 = \frac{4\mathrm{P}}{\mathrm{n}(\mathrm{n}-1)} - 1 \qquad \text{(Equation 5.1)}$$

Here, n is the number of items in each sequence, and *P* number of concordant pairs over all the items. It is calculated by counting the number of item pairs (a, b) for which *a* comes before *b* in both sequences. Higher values for P lead to higher values for the Tao correlations also. The resultant τ value is between -1 and 1. The values that are greater than zero imply correlation between orderings where less than zero values imply the opposite. Table 5.6 shows the correlation between normal vs. swapped rankings and between normal vs. reversed rankings. The aim here is to analyze that users really make informed decisions.

Table 5.6: The Kendall Tau correlation values in ordinary search.

Correlation	τ value
Normal vs. Swapped	0.33
Normal vs. Reversed	0.37

Here, we observe that there is an agreement between the ideal rankings. This suggests that even if the ordering is manipulated, users are able make relevant selections among search results, but as shown in Figure 5.4 it is also biased.

The results of thumbnailed search revealed rather different results that are explained in the next section.

5.4.2 Searching With Thumbnails

Our eye tracking data suggests that the users’ focus durations on thumbnails are extremely shorter compared to focus duration on abstracts. Intuitively, it is more probable that the human brain evaluates thumbnails faster than it does texts; however, such a huge gap between gaze durations was certainly not expected. The bars in Figure 5.5 shows the total duration of focus on two different AOIs: abstract and thumbnail. In Figure 5.5, pic# is the label (given by ClearView software) representing each search result.

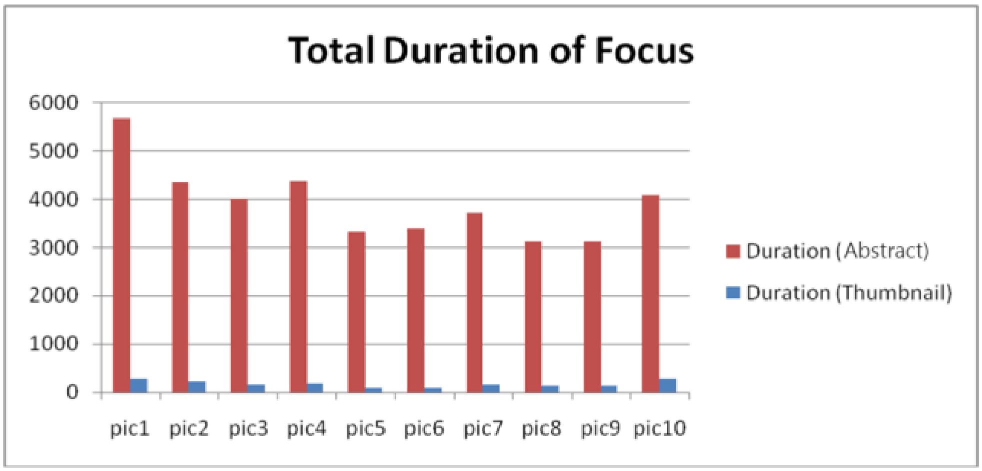


Figure 5.5: Total duration of focus on abstracts and thumbnails.

The next graphical representation tells more about the functionality of thumbnails in search results. Figure 5.6 shows the ratio between the clicks on thumbnails and the total number of clicks for each ranking. For instance, for the top ranked search result (linkp1) nearly 12% of the times the thumbnail part is clicked on where 88% of the times the abstract part gets clicked. Previous work (e.g. Huang et al., 2011) suggests that there is a strong relationship between the cursor position and the focused area on a page.

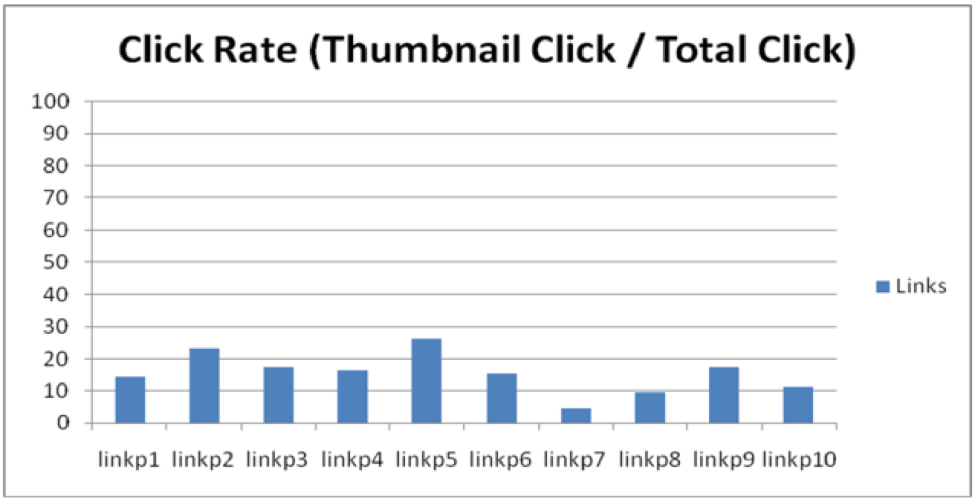


Figure 5.6: The ratio between the clicks on thumbnails and all the clicks.

Therefore the click rate data strongly implies that approximately 85% of the time, the final decision before clicking on a search result is influenced by the abstract part; not the thumbnail.

An important and the most surprising observation on thumbnailed search is that the thumbnails help users to partially break their trust bias towards search engine rankings. Figure 5.7 depicts the click counts for the first 10 rankings in thumbnailed search. Without thumbnails, the two most clicked links in reversed ordered search are the 1^{st} and the 3^{rd} links. In thumbnailed search, the most clicked links are the 10^{th} and 7^{th} links.

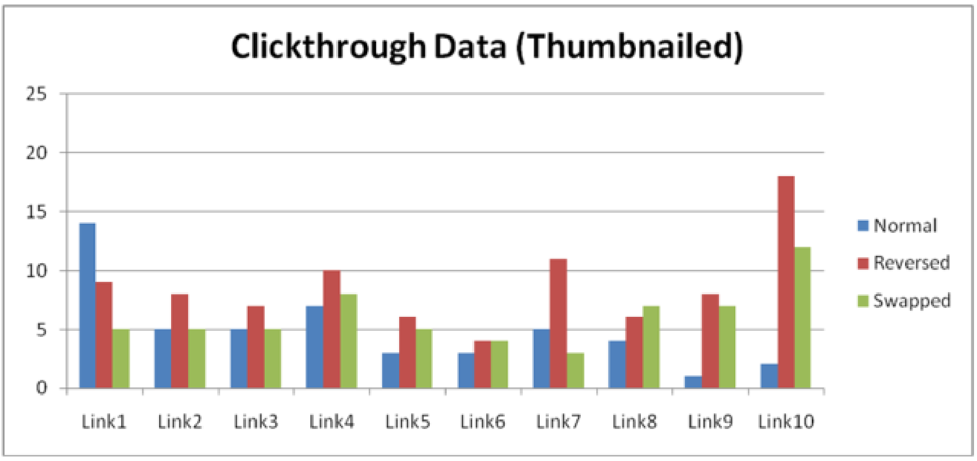


Figure 5.7: The click counts for the thumbnailed rankings.

The Kendall Tau correlation value between normal and reversed orderings supports our argument that the highest correlation value between normal and manipulated ordering is achieved in thumbnailed search, in particular, between normal and reversed ideal sets. Table 5.7 shows ideal rankings for thumbnailed search based on the data in Figure 5.7.

Table 5.7: Ideal rankings for thumbnailed search.

normal	1	4	2	3	7	8	5	6	10	9
swapped	10	4	9	8	1	2	3	5	6	7
reversed	1	4	7	10	2	9	8	3	6	5

The following Table 5.8 shows the Kendall Tau correlation values for thumbnailed search.

Table 5.8: The Kendall Tau correlation values in thumbnailed search.

Correlation	**τ value**
Normal vs. Swapped	-0.11
Normal vs. Reversed	0.42

When we consider that the ranking quality of swapped ordering is much better than the ranking quality of reversed ordering, we may conclude that thumbnailed search yields its best performance when the quality of search ranking is extremely low.

Obviously, thumbnails help user to find the most relevant results to their query. This inference supports our initial argument that the visual characteristics of a webpage may give clues about the relevancy of which the users judge.

5.5 Conclusion

In this study, we presented the results of eye tracking and clickthrough analysis experiments on the effectiveness of thumbnails in Web search. We analyzed the distribution of users' attention between thumbnails and abstracts. The results are surprising for us in a way that a huge gap between gaze durations on thumbnails and abstracts was not expected. The experimental results suggest that the attention of the majority of the users is focused on the abstract part. At first sight, this huge gap leads us to the conclusion that thumbnails are not as effective as abstracts when users evaluate search results. However, the experiments with reversed ordering of results suggest that thumbnails are useful in detecting the most relevant documents despite very short gaze durations.

This study reveals a link between visuals of a webpage and its quality (relevancy) of it with a given query. This study also confirms the results of the previous eye tracking study (i.e., Joachims et al., 2005) that users have a trust bias for search engine rankings that they have a tendency to click on the topmost results. Thumbnails prove to have an assistive role in breaking that bias.

With a natural and high quality ordering of search results, the usage of thumbnails does not seem practical since thumbnails seem most useful when the ranking quality is extremely low. This assistive role of thumbnails may be utilized in improving the effectiveness pagination where the relevancy of the rankings to the given query decreases as ranking increases. Thumbnail effect can be analyzed in other situations where the ranking quality is low. For example, Downey et al. (2008) argued that the search quality decreases as the popularity of queries decreases. What they suggest for future work is to increase the ranking of low ranked URLs when the popularity of query is low. Instead, thumbnails may be an effective component for low-popularity queries and may enable implicit feedback mechanism to the ordering of the low ranked webpages.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

Today search engines are the first address for information seekers to begin their Web activity by sending millions of query requests and expecting the most appropriate results to start with. Web search engines should be efficient in responding user queries within milliseconds, and they should be effective in making users satisfied with the relevant information provided through the query process. This study investigated a way to increase the effectiveness measures of a Web search engine with three main research areas:

- The impact of link quality and usage information on page importance calculation,
- The use of Turkish stemmer for indexing and query substitution, and
- The use of thumbnails for Web search engine result visualization.

First, we focused on the elimination of spam pages and the creation of a Web model for revealing the true value of hyperlinks between pages. The experimental results suggest that refining the link weights on a trusted set of pages increases the PageRank performance remarkably. The cooperative work of PageRank and TrustRank algorithms is another key point in this success. With the integration of the usage data of real Web surfers, TrustRank provides the PageRank algorithm with a safer working environment. The experiments were carried out on the Turkish Web domain, and the result of this study offers an alternative and efficient way of selecting a trusted seed list from the surfing experience of real users that helps eliminate spam pages. As for future work, this study shall be conducted on the different characteristics of Web domains with potentially different link structures to asses if this different link structure requires different weighting schemes.

Second, we focused on developing a framework to increase the IR precision score and Web relevance by using a stemmer/lemmatizer for Turkish Web search engine domain. In contrast to the previous studies that improved IR precision scores while decreasing Web relevance with stemming, this approach showed that by combining unstemmed and stemmed content, the best precision score and Web relevance pair could be observed. We also showed that by using the MapReduce programming model, the stemming/lemmatizing cost is notably reduced. For future work, this study shall be conducted by testing the effect of long and short queries and the intent of the informational, navigational or transactional query types.

Finally, we examined the effect of thumbnailed results on user behavior in Web search by interpreting the eye tracking and clickthrough data of users and tried to answer the question, Can thumbnailed search results help users find the most relevant document on search engine results pages? Previous studies (Joachims 2002; Joachims et al., 2005) reported that users’ choices are informed but biased, and they are inclined to click on the topmost results disregarding the ordering of the results. We analyzed the distribution of users’ attention between thumbnails and abstracts. We observed that when the ranking quality on a SERP was intentionally decreased, the users were able to find the most relevant results by using the thumbnails, which is something they couldn’t accomplish in the absence of thumbnails. This study reveals a link between visuals of a webpage and its quality (relevancy) with a given query. This study also confirms the results of the (Joachims, 2002; Joachims et al., 2005) previous eye tracking studies that users have a trust bias for search engine rankings, and they have a tendency to click on the topmost results. However, thumbnails prove to have an assistive role in breaking that bias. For future work, the usage of thumbnails shall be investigated as an effective component for low-popularity queries and may be tested as an implicit feedback mechanism to the ordering of the low ranked webpages.

REFERENCES

Ahlgren, P., Kekalainen, J. (2007). Indexing strategies for Swedish full text retrieval under different user scenarios. Information Processing and Management, 43(1), 81-102.

Anick, P. (2003). Using terminological feedback for web search refinement: A log-based study. In SIGIR, 2003.

Arasu, A., Cho, J., Garcia-Molina, H., Paepcke, A., Raghavan, S. (2001). Searching the web. ACM Transactions on Internet Technology, 1(1): August 2001.

Baeza-Yates, R., Davis, E. (2004). Web page ranking using link attributes. In Alt. track papers & posters, WWW Conf., pp. 328-329. New York. NY USA.

Baeze-Yates, R., Riberio, B. (1999). Modern Information Retrieval, Addison Wesley

Bar-Ilan, J., Gutman, T. (2003). How do search engines handle non-English queries? - A case study.WWW (Alternate Paper Tracks) 2003.

Bar-Ilan, J., Gutman, T. (2005). How do search engines respond to some non-English queries? J. Information Science 31(1): 13-28 (2005)

Belkin, N. J. (1993). Interaction with texts: Information retrieval as information-seeking behavior. In Information retrieval ’93, Von der Modellierung zur Anwendung (pp. 55–66). Konstanz, Germany: Universitaetsverlag Konstanz.

Berger, A., Caruana, R., Cohn, D., Freitag, D., Mittal, V. (2000). Bridging the lexical chasm: Statistical approaches to answer-finding. In Proc. Int. Conf. Research and Development in InformationRetrieval, 192-199, 2000.

Bharat, K., Broder, A. (1999). Mirror, mirror on the web: A study of host pairs with replicated content. In Proceedings of the Eighth International Conference on The World-Wide Web.

Braschler, M., Ripplinger, B. (2004). How effective is stemming and decompounding for German text retrieval? Information Retrieval, 7, 291-316. Brin, S. And Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. Comput. Netw. ISDN Syst. 30, 1-7, 107–117.

Broder, A. (2002). A taxonomy of web search. SIGIR Forum, 36 (2), 3-10, 2002.

Broder, A., Kumar, R., Maghoul, F., Raghavan, P., Rajagopalan, S., Stata, R., Tomkins, A., And Wiener, J. (2000). Graph structure in the web: Experiments and models. In Proceedings of the Ninth International Conference on The World Wide Web.

Buscher, G., Dumais, S., Cutrell, E. (2010). The good, the bad, and the random: An eye-tracking study of ad quality in web search. SIGIR 2010.

Can, F., Kocberber, S., Balcik, E., Kaynak, C., Ocalan, H. C., Vursavas, O.M. (2006). First large-scale information retrieval experiments on Turkish texts. In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '06).

Cho, J., Garcia-Molina, H. (2000b). The evolution of the web and implications for an incremental crawler. In Proceedings of the 26th International Conference on Very Large Data Bases.

Cho, J., Garcia-Molina, H.(2003) Estimating frequency of change. ACM Transactions on Internet Technology, 3(3): August 2003.

Craig, E., Wills, M. M. (1999). Towards a better understanding of web resources and server responses for improved caching. Computer Networks 31(11-16): 1231-1243, 1999.

Dean, J., Ghemawat, S. (2004). MapReduce: Simplified data processing on large clusters. In OSDI’04, 6th Symposium on Operating Systems Design and Implementation, Sponsored by USENIX, in cooperation with ACM SIGOPS, pages 137–150, 2004.

Dinçer, B.T., Karaoglan, B. (2003). Stemming in agglutinative languages: A probabilistic stemmer for Turkish. In Proceedings of ISCIS. 2003, 244-251.

Ding, C., He, X., Husbands, P., Zha, H., Simon, H. (2004). PageRank, HITS and a unified framework for link analysis. Proceedings of the ACM SIGIR 2002, Tampere, Finland, pp:353-354.

Ding, L., Finin, T., Joshi, A., Pan, R., Cost, R. S., Sachs, J. (2004). Swoogle: A search and metadata engine for the semantic web. Paper presented at the thirteenth ACM conference on information and knowledge management (CIKM), Washington DC (November)

Downey, D., Dumais, S., Liebling, D. & Horvitz, E. (2008). Understanding the relationship between searchers' queries and information goals. In Proceedings CIKM 2008, 449-458.

Dumais, S. (2010). Individual differences in gaze patterns for web search. IIiX 2010, August 18–21, 2010, New Brunswick, New Jersey, USA.

Ekmekcioglu, F.C., Willett, P. (2000). Effectiveness of stemming for Turkish text retrieval. Program, 34(2), 195-200, 2000.

Figuerola, C. G., Gomez, R., Rodriguez, A. F. Z., Berrocal, J., L. A. (2006). Stemming in Spanish: A first approach to its impact on information retrieval. Working Notes for the CLEF 2001 Workshop 3 September, Darmstadt, Germany, edited by Carol Peters.

Ghemawat, S., Gobioff, H., Leung, S.-T. (2003). The Google file system. SIGOPS Oper. Syst. Rev., vol. 37, pp. 29-43, 2003.

Gisbergen, M. S. V., Most, J. V. D., & Aelen, P. (2007). Visual attention to online search engine results. Market Research Agency De Vos & Jansen.

Gyöngyi, Z., Garcia-Molina, H., Pedersen, J. (2004). Combating web spam with TrustRank. Proc. 30th International Conference on Very Large Databases, Morgan Kaufmann, 2004, pp. 576–587

Huang, J.,White, R. & Dumais, S. (2011). No clicks, no problem: Using cursor movements to understand and improve search. CHI 2011, May 7–12, 2011, Vancouver, BC, Canada.

Hull, D. (1996). Stemming Algorithms - A case study for detailed evaluation. JASIS, 47(1):70–84, 1996.

Jansen, B. J. (2006). Using temporal patterns of interactions to design effective automated searching assistance systems. Communications of the ACM, 49(4), 72–74.

Jansen, B. J., Booth, D.L., Spink, A. (2008). Determining the informational, navigational, and transactional intent of Web queries. Information Processing and Management: an International Journal, v.44 n.3, p.1251-1266, May, 2008

Jarvelin, K., Kakelainen, J. (2000). IR evaluation methods for retrieving highly relevant methods. In SIGIR, 2000.

Joachims, T. (2002). Optimizing search engines using clickthrough data. SIGKDD 2002.

Joachims, T., Granka, L., Pan, B., Hembrooke, H., Gay, G. (2005). Accurately interpreting clickthrough data as implicit feedback. In Proceedings SIGIR 2005, 154-161.

Kendall, M. (1955). Rank correlation methods. Hafner, 1955.

Kettunen, K., Kunttu, T. & Jarvelin, K. (2005). To stem or lemmatize a highly inflectional language in a probabilistic IR environment? Journal of Documentation, 61(4), 476-496.

Kleinberg, J. M. (1998). Authoritative sources in a hyperlinked environment. In SODA'98: Proceedings of the ninth annual ACM-SIAM symposium on discrete algorithms (pp. 668–677). Philadelphia, PA: Society for Industrial and Applied Mathematics.

Kraaij, W., Pohlmann, R. (1996). Viewing stemming as recall enhancement. In SIGIR, 1996.

Krovetz, R. (1993). Viewing morphology as an inference process. In Proceedings of the 16th International Conference on Research and Development in Information Retrieval (ACM SIGIR'93) (pp. 191-202). Pittsburgh: ACM.

Lagun, D., Agichtein, E. (2011). ViewSer: Enabling large-scale remote user studies of web search examination and interaction. SIGIR'11, July 24–28, 2011, Beijing, China.

Langville, A. N., Meyer, C. D. (2004). Deeper inside PageRank. Internet Mathematics, 1(3), 335–400.

Leporini, B., Andronico, P. & Buzzi, M. (2004). Designing search engine user interfaces for the visually impaired. W4A at WWW2004.

Liu, Y., Gao, B., Liu, T. Y., Zhang, Y., Ma, Z., He, S., Li, H. (2008). BrowseRank: Letting web users vote for page importance. In SIGIR'08: Proceedings of the 31st annual international ACM SIGIR conference on research and development in information retrieval (pp. 451–458). New York, NY: ACM.

Liu, Y., Liu,T. Y., Gao, B., Ma,Z., L, H. (2010). A framework to compute page importance based on user behaviors. In Information Retrieval, Vol. 13, Nr. 1 (February 2010), p. 22-45

Moukhad, H., Large, A. Information retrieval from full-text Arabic databases: Can search engines designed for English do the job? Libri 51 (2001), 63-74.

Mujoo, A., Malviya, M. K., Moona, R., and Prahakar, T. (2000). A search engine for Indian languages. EC-Web 2000, Lecture Notes in Computer Science 1875 (2000), 349-358.

Nie, L., Wu, B., Davison, B.D. (2007). A cautious surfer for PageRank. In Proceedings of the 16th international conference on World Wide Web (WWW '07). ACM, New York, NY, USA, 1119-1120.

O'Brien, M., Keane, M. (2007). Modeling user behavior using a search-engine. IUI'07, January 28–31, 2007, Honolulu, Hawaii, USA.

Oard, D., and Diekama, A. R. (1998). Cross language information retrieval. Annual Review of Information Science and Technology, 33 (1998), 223-256.

Oflazer, K. (1994). Two level description of Turkish morphology. Linguistics and Literary Computing, 1994.

Ostergren, M., Yu, S., Efthimiadis, E. (2010). The value of visual elements in web search. SIGIR'10, July 19–23, 2010, Geneva, Switzerland.

Page, L., Brin, S., Motwani, R., & Winograd, T. (1999). The PageRank citation ranking: Bringing order to the web. Technical report 1999-66, Stanford InfoLab. Previous number = SIDL-WP-1999-0120.

Peng, F., Ahmed, N., Li, X., Lu, Y. (2007). Context sensitive stemming for web search. In Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '07).

Porter, M. (1980). An algorithm for suffix stripping. Program, 14(3):130-137, 1980.
Radlinski, F., Joachims, T. (2005). Query chains: Learning to rank from implicit feedback. KDD'05, August 21-24, 2005, Chicago, Illinois, USA.

Salton, G. (1988). Automatic text processing. Addison-Wesley Series in Computer Science. Addison-Wesley Longman Publ. Co., Inc., Reading, MA.

Sever, H., Bitirim Y.(2003). FindStem: Analysis and evaluation of a Turkish stemming algorithm. LNCS 2857: 238-251, 2003.

Solak, A., Can, F. (1994). Effects of stemming on Turkish text retrieval. ISCIS Conf., pp. 49-56, 1994.

Velez, B., Weiss, R., Sheldon, M. A., Glifford, D. K. (1997). Fast and effective query refinement. In SIGIR, 1997.

Witten, I., Moffat, A., And Bell, T. (1999). Managing gigabytes: Compressing and indexing documents and images. 2nd ed. Morgan Kaufmann Publishers Inc., San Francisco, CA.

Wu, B., Goel, V., Davison, B. D. (2006). Topical TrustRank: Using topicality to combat web spam. In Proceedings of the 15th international conference on World Wide Web (WWW '06). ACM, New York, NY, USA, 63-72

Würtz, E. (2005), Intercultural communication on websites: A cross-cultural analysis of websites from high-context cultures and low-context cultures. Journal of Computer-Mediated Communication, 11: 274–299. doi: 10.1111/j.1083-6101.2006.tb00313.x

Xing, W., Ghorbani, A. (2004). Weighted PageRank algorithm. Communication Networks and Services Research, 2004. Proceedings. Second Annual Conference.

Zobel, J. Mo, A. (2006). Inverted files for text search engines. ACM Computing Surveys, 38(2):Article 6.

CURRICULUM VITA

PERSONAL INFORMATION

Surname, Name: Fidan, Güven
Nationality: Turkish (TC)
Date and Place of Birth: 25 December 1975, Kırşehir
Marital Status: Married
Phone: +90 312 210 13 40
Fax: +90 312 210 13 41
email: guven.fidan@gmail.com

EDUCATION

Degree	**Institution**	**Year of Graduation**
MS	METU Computer Engineering	2001
BS	METU Computer Engineering	1998
Minor Programme	METU Industrial Engineering	1998
High School	Kırşehir High School, Kırşehir	1993

WORK EXPERIENCE

Year	**Company**	**Position**
2004- Present	AGMLab Ltd. , Ankara	CEO
2003- 2005	Bilkent University CTIS Department	Instructor
2002-2003	KoçSistem A.Ş., Ankara	Software Engineer
2002-2002	Havelsan A.Ş., Ankara	Software Engineer
1999-2001	Infopark A.Ş., Ankara	Software Engineer
1998-1999-	Likom Proje, Ankara	Software Engineer

PUBLICATONS

Schaal,M., Fidan, G., Muller, R.M., Dagli. O. (2010). Quality Assessment in the Blog Space. The Learning Organization (TLO), Special Issue on Web 2.0 Practical Implications.

Sinaci, A. A., Sehitoglu, O. T., Yondem, M. T., Fidan, G. (2010). SEMbySEM in Action: Domain Name Registry Service Through a Semantic Middleware. eChallenges e-2010 Conference Proceedings Paul Cunningham and Miriam Cunningham (Eds) IIMC International Information Management Corporation, 2010 ISBN: 978-1-905824-20-5

Bayir, M. A., Toroslu, I. H., Cosar, A., Fidan, G. (2009). SmartMiner: A New Framework for Mining Large Scale Web Usage Data. WWW 2009

Bayir, M. A., Toroslu, I. H., Cosar, A., Fidan, G. (2008). Discovering More Accurate Frequent Web Usage Patterns CoRR abs/0804.1409

Fidan, G., Kandiller, L., Oğuztüzün, H. (2002). A Decision Support System for Assembly Line Balancing. 30th International Conference on Computers and Industrial Engineering, 1, (2002), p.239--244.

CURRENT RESEARCH INTERESTS

Information Retrieval & Extraction, Natural Language Processing, Ontology Generation and Management, Semantic Web Mining, Data Mining, Distributed Computing Systems, Collaborative Systems, Web 2.0 and 3.0 Projects, Semantic Services.

VITA

Güven Fidan was born in Kırşehir in 1975. He graduated from METU, Department of Computer Engineering in 1998. He received his M.S. degree from METU, Department of Computer Engineering in 2001. As a professional, he has participated and managed numerous R&D projects funded by the Scientific & Technological Research Council of Turkey (TUBITAK) and European Union (EU). He has also given lectures in Department of Computer Technology and Information Systems at Bilkent University. His current research interests are information retrieval and extraction, web mining, recommender and collaborative systems.

TEZ FOTOKOPİSİ İZİN FORMU

ENSTİTÜ

Fen Bilimleri Enstitüsü □

Sosyal Bilimler Enstitüsü □

Uygulamalı Matematik Enstitüsü □

Enformatik Enstitüsü ☒

Deniz Bilimleri Enstitüsü □

YAZARIN

Soyadı: Fidan
Adı : Güven
Bölümü :Bilişim Sistemleri

TEZİN ADI (İngilizce):

IDENTIFYING THE EFFECTIVENESS OF A WEB SEARCH ENGINE WITH TURKISH DOMAIN DEPENDENT IMPACTS AND GLOBAL SCALE INFORMATION RETRIEVAL IMPROVEMENTS

TEZİN TÜRÜ: Yüksek Lisans □ Doktora ☒

1. Tezimin tamamından kaynak gösterilmek şartıyla fotokopi alınabilir. □
2. Tezimin içindekiler sayfası, özet, indeks sayfalarından ve/veya bir bölümünden kaynak gösterilmek şartıyla fotokopi alınabilir. ☒
3. Tezimden bir (1) yıl süreyle fotokopi alınamaz. □

TEZİN KÜTÜPHANEYE TESLİM TARİHİ: ....