A FRAMEWORK FOR RANKING AND CATEGORIZING MEDICAL DOCUMENTS

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

BY

MOHAMMED GH. I. AL ZAMIL

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

JUNE 2010

Approval of the Graduate School of Informatics:

_____

Prof. Dr. Nazife Baykal

Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Doctor of Philosophy.

_____

Assist. Prof. Tuğba Taşkaya Temizel

Head of the Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Doctor of Philosophy.

_____	_____
Prof. Dr. Nazife Baykal	Assist. Prof. Aysu Betin Can
Co-Supervisor	Supervisor

Examining Committee Members:

Assoc. Prof. Ilyas Çiçekli	(BILKENT, CS)	_____
Asst.Prof.Dr. Aysu Betin Can	(METU, IS)	_____
Assist.Prof.Dr. Reza Hassanpour	(Çankaya, CS)	_____
Assoc. Prof. Ünal Erkan Mumcuoğlu	(METU, MIN)	_____
Assist. Prof. Tuğba Taşkaya Temizel	(METU, IS)	_____

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, Surname: ***Mohammed GH. I. AL Zamil***

Signature :

ABSTRACT

A FRAMEWORK FOR RANKING AND CATEGORIZING MEDICAL DOCUMENTS

Mohammed GH. I. AL ZAMIL
Ph.D., Department of Information Systems
Supervisor: Assist. Prof. Aysu Betin Can
Co-Supervisor: Prof. Dr. Nazife Baykal

June 2010, 161 Pages

In this dissertation, we present a framework to enhance the retrieval, ranking, and categorization of text documents in medical domain. The contributions of this study are the introduction of a similarity model to retrieve and rank medical text-documents and the introduction of rule-based categorization method based on lexical syntactic patterns features.

We formulate the similarity model by combining three features to model the relationship among document and construct a document network. We aim to rank retrieved documents according to their topics; making highly relevant document on

the top of the hit-list. We have applied this model on OHSUMED collection (TREC-9) in order to demonstrate the performance effectiveness in terms of topical ranking, recall, and precision metrics.

In addition, we introduce ROLEX-SP (**R**ules **O**f **LEX**ical **S**yntactic **P**atterns); a framework for automatically inducing rules to build text classifiers based on lexical syntactic patterns as a set of features to categorize text-documents. The proposed method is dedicated to solve the problem of multi-class classification and feature imbalance problems in domain specific text documents. Furthermore, our proposed method is able to categorize documents according to a predefined set of characteristics such as: user-specific, domain-specific, and query-based categorization which facilitates browsing documents in search-engines and increase users ability to choose among relevant documents. To demonstrate the applicability of ROLEX-SP, we have performed experiments on OHSUMED (categorization collection). The results indicate that ROLEX-SP outperforms state-of-the-art methods in categorizing short-text medical documents.

Keywords: Medical document, Ranking, Categorization

ÖZ

TIBBİ BELGELERİN KATEGORİLENDİRİLMELERİ VE

SIRALANMASI İÇİN BİR ÇERÇEVE

Mohammed GH. I. AL Zamil
Doktora, Bilisim Sistemleri Bölümü
Tez Yöneticisi: Yard. Doç. Dr. Aysu Betin-Can
Ortak Tez Yöneticisi: Prof. Dr. Nazife Baykal

Haziran 2010, 161 sayfa

Bu tezde, tıbbi alandaki yazılı dökümanların bilgi erişim sistemlerinde sıralanması ve sınıflandırılmasını için bir model çerçeve sunulmaktadır.Bu çalışmanın katkıları tıbbi metin belgelerinn erişim ve sıralanması için bir benzerlik modelin ve bu belgelerin sözlüksel sentaktik kalıp özelliklerine dayanan bir kural- tabanlı sınıflandırma yönteminin öne sürülmeleridir.

Benzelik modelini üç özelliği birleştirerek,belgeler ve belgelerin yapıları arasındaki ilişkileri biçimlendirmek için formüle edilmiştir. Erişilen metinleri, konularına

dayanarak; yüksek derecede ilgili belgeleri listenin en üst sıralarına yerleştirmek sıralama amaçlanmıştır. Bu model OHSUMED (TREC-9) koleksiyonunda konuya ait sıralama,anımsama ve hassaslık metriklerine dayanarak fayda performansını göstermek için uygulanmıştır.

Ayrıca, Rolex-SP (Rules of Lexical Syntactic Patterns) adında, metin belgelerini sınıflandırmak için özellikler dizisi olarak sözlüksel sentaktik kalıp özelliklerine dayanan kural tabanlı metin sınıflandırıcıların otomatik sonuç çıkarabilecekleri bir metod sunmaktayız. Önerilen yöntem çok sınıflı tasnif problemlerini çözmek ve alana özel metin belgelerindek Idengesizlik alanındaki dengesizlik problemlerini aşabilmek için önerilmiştir. Bunun yanında sunulan metod önceden tanımlanabilen karakteristiklere dayanarak metinleri kategorize edebilir. Bu karakteristikler içinde kullanıcıya özel tanımlamalar, alana özel tanımlamalar, ve arama motorlarında belge taramaya yardımcı olup sorguyla ilgili belgeler arasında kullanıcının seçim yapabilmesine yardımcı olan sorguya dayalı sıralamala tanımları sayılabilir Rolex-SP uygulanabilirliğini göstermek için, OHSUMED üzerinde deneyler yapılmıştır. Sonuçlar, Rolex-SP'nin kısa tıbbi metin belgelerinin sıralandırılmasında, var olan teknolojilerin son durumundan daha iyi verim verdiğini göstermiştir.

Anahtar kelimeler: Tıbbi belge, Sıralaması, Sınıflama

To The Memory of My Father

Ghazi I. AL Zamil

ACKNOWLEDGEMENT

I would like to take this opportunity to express my thanks to many people, who helped me in a way or another, including my research advisor, co-advisor, instructors in informatics institute, my friends in Ankara, and my family.

First of all, I would like to express my heartfelt thank to ***Dr. Aysu Betin-Can*** for her support through out my study. Her enthusiasm, comments, and advices made me energetic, productive, and hardworking. I will never forget the courses she taught me, especially Software Verification and Design Patterns. "*I am looking forward to work with you on the application of Model Checking*". I expect to continue working with her till the end of my academic life.

I would also like to thank our wonderful Dean and my co-supervisor Professor ***Nazife Baykal***. I did never expect to meet a nice person like her. I will never forget her words at the end of our first meeting; "*I believe in you Mohammed*". Her directions helped us to stay on the right track.

I would like to thank my friends in **Saudi** and **Kuwaiti** embassies in Ankara and a special thank to the ambassadors of **Jordan** and **Oman** sultanate. I would also like to thank my close friends in Ankara: Assoc. Prof. **Thabit Abdel Jawad** (Çankaya University), **Maher Durnaika**, **Ayman Hijazi**, **Hisham Nasser**, **Wesam Abdullah**,

Ahmad Kaddoura, and **Mohammed Gharamti**. We really spent and enjoyed good and unforgettable years in Turkey. I would also like to thank my friend **Songül Gelişken** and her nephew **Bahar Çetin** who translated the abstract section into Turkish.

Last, but not least, I would like to thank my mother ***Layla***, my brother ***Muhanned***, and my Sisters ***Luma***, ***Ruba***, and ***Haya*** for supporting me. You all were great especially after the death of my father "May Allah have Mercy Upon Him". Thank you so much my dearest family; thank you for being my family.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS AND ABBREVIATIONS

ASP	: Active Server Page
DSN	: Document Semantic Network
ID3	: Iterative Dichotomiser
IIS	: Internet Information Server
ILP	: Inductive Logic Programming
LP	: Label Propagation
LSA	: Latent Semantic Analysis
LSP	: Lexical Syntactic Patterns
MCC	: Multi-Class Classification
MeSH	: Medical Subject Headings
MIR	: Medical Information Retrieval
NP	: Noun Phrase
OHSUMED	: Oregon Health Sciences University MEDLINE
PAC	: Probably Approximately Correct
RC	: Reference count
ROLEX	: Rules of Lexico
RS	: Representative Set
TREC	: Text Retrieval Conference
TREC-EVAL	: TREC Evaluation Program
UMLS	: Unified Medical Language System

CHAPTER 1

INTRODUCTION

A major problem in biomedical informatics involves the contextual retrieval and ranking of medical documents. Many medical information retrieval systems restrict their services to medical experts. However, common people tend to be more informed to the decision processes related to their health problems. Thus, such ordinary users search the Internet for the purpose of locating relevant information. This situation has led to a plenty amount of medical queries on the Internet by users who are not able to specify their needs using medical jargons. Therefore, there is an increasing demand for effective medical information retrieval techniques and tools to help people with no medical training in searching for health information on the Internet [1].

As the web grows rapidly, the task of locating relevant information from multiple sources is becoming hard. Medical search engines have been proposed to overcome

this problem by limiting the searching process to medical and health domain, such as MedicoPort [1], PubMed [2], and WEBMD [3]. In this domain, studies in [4, 5] showed that searching biases affect the decision of health care information consumers. These biases resulted from the weak experience of users in medical concepts.

Traditional information retrieval features, such as terms and phrases, have been widely used to find similarities between documents and queries. Recent research [6] shows that applying combined semantic features resulted in an effective retrieval process in domain-specific information retrieval systems.

Although improvements on retrieval and ranking tasks affect the usability, categorization of documents into classes of knowledge play a significant role to help users choose relevant documents according to their needs. Text categorization is a method that is capable to assign a text document under one or more class of knowledge on the basis of its constituent text. Many machine learning methods and techniques have been widely used to build classifiers for text categorization, using labeled training set of data, such as K-nearest neighbors, neural networks, Bayesian, and SVM. Detailed description of these methods can be found in [7, 8, 9].

Multi-class classification [10, 11] and feature imbalance [12, 13] are central problems in machine learning methods to address classification of free text with minimal labels description. Rule-based classification algorithms such as [14], [15],

[16], [17], and [18] have been used to handle these problems, but restrict features on a vocabulary of terms and the specifications of structured labels in training dataset.

In this research, we are proposing an effective similarity model to create document networks in a domain-specific environment; specifically in medical domain. The proposed model is able to create a robust connectivity among documents within the network; making the search process and ranking more effective. The strength of the proposed mapping features resulted from the incorporation of domain ontologies to construct the connection among documents.

In addition, we introduce ROLEX-SP (**R**ules **O**f **LEX**ical **S**yntactic **P**atterns); a method for automatically inducing rules to build text classifiers relies on lexical syntactic patterns as a set of features to categorize text-documents. The proposed method is dedicated to solve multi-class classification and feature imbalance problems of domain specific text documents. Furthermore, our proposed method is able to categorize documents according to a predefined set of aspects such as: user-specific, domain-specific, and query-based categorization which facilitates browsing documents in search-engines and increase users ability to choose among relevant documents. For example, a medical search engine might display retrieved documents under user-specific categories such as symptoms, diagnosis, treatment, and medication categories.

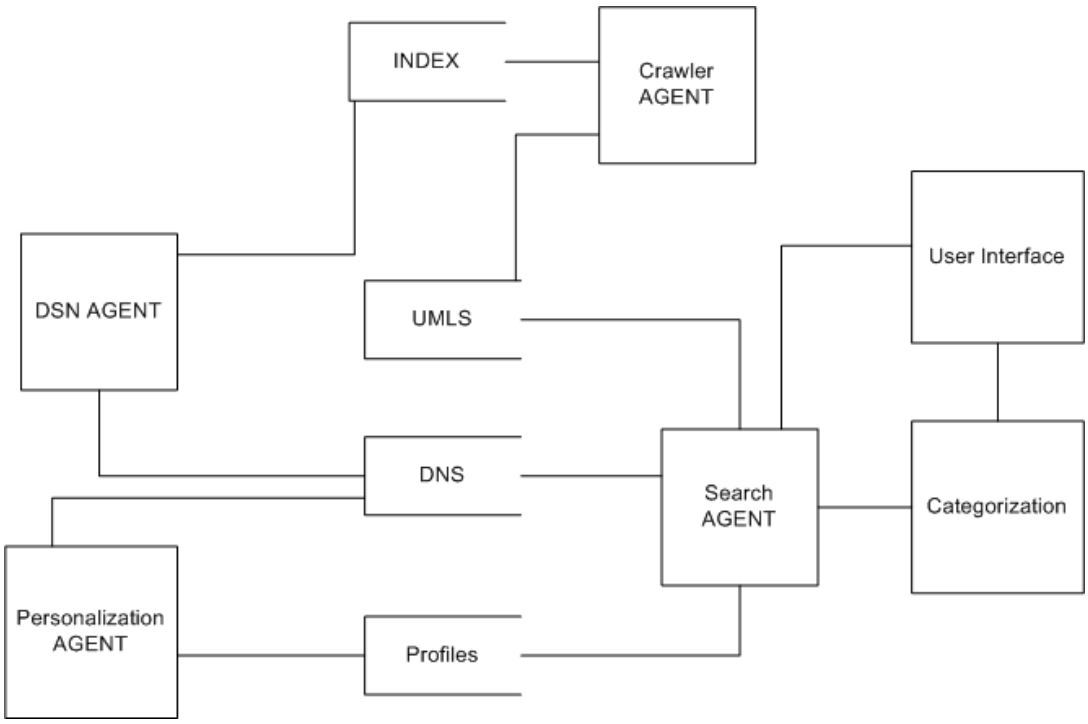


Figure 1.1- Multi-Agent Design of the Proposed Framework

We have re-developed MedicoPort [1] to realize our proposed methods. Figure 1.1 shows a general view of a multi-agent design framework of MedicoPort. Our contribution in this framework can be summarized in two phases including: document semantic network (DSN) agent and Categorization agent.

The DSN agent represents the first phase in our study. The main goal behind adopting this module is to construct a document network that model the relationship among documents. Such network, then, used to rank the retrieved documents based on their topics. In this context, the proposed similarity model has been applied to compute the similarity among documents in the DSN.

Categorization agent is used to classify documents using little description of class labels. We aimed to classify the retrieved hit-list into categories to facilitate browsing relevant documents. In this context, we introduce a rule-based categorization method that is based on lexical syntactic patterns.

Finally, we applied a personalization technique to our framework to complete the retrieval process. The application of the personalization method takes into consideration the need to rank classes of information into useful order according to user interests. We track user clicks and keep information about users' behaviors in order to fulfill this task.

1.1 Summary of Contributions

This study aims to address the problem of ranking medical documents according to their topics based on document semantic network structure. Our proposed solution is a similarity model to rank documents according to their topics. Consequently, the precision at the top documents in the hit list increased; expanding the number of relevant documents in the hit list.

Furthermore, we introduce a rule-based categorization method to address multi-class classification and feature imbalance problems. The goal is to categorize medical documents after retrieval task completes into simple categories with minimal description such as: Symptoms, diagnosis, treatment, etc. Categorizing documents

into these categories facilitate locating relevant documents, and therefore, reduces the searching biases.

Finally, we applied a personalization technique for the purpose of ranking classes of knowledge according to user interests. The application of this method relies on tracking user browsing behaviour and extracting information about users browsing patterns. The main goal is to complete the cycle of the retrieval process to include all relevant tasks.

1.2 Scope of this Dissertation

Recently, numerous research studies have been conducted to address different aspects of domain specific information retrieval systems; specifically in medical domain. Chapter 2 provide a review of methods, techniques, and systems that provide solutions to different aspects of medical information retrieval problem.

In this dissertation, we studied the effect of formalizing a similarity model to construct a documents network based on the semantic enrichment of domain concepts. The document network, then, ranks medical documents according based on their relevancy to a specific query and according to their topic. The improvement of the proposed model has been measured using recall and precision metrics.

Furthermore, we proposed a rule-based categorization method to classify medical documents using minimal label information. In this context, our technique addresses

Multi-class classification and feature imbalance problems to categorize medical documents. We studied the enhancement on the classification performance resulted from applying lexical syntactic patterns.

In summary, we performed experiments on OHSUMED (Oregon Health Sciences University MEDLINE) benchmark to evaluate the proposed techniques. In the first experiment, we applied our similarity model to evaluate the retrieval and ranking in terms of precision. Also, we compared our findings with the top five baselines reported in TREC. The results indicate that the proposed model outperformed other models. We also performed a comparison based on interpolated precision metric with a similar method that expands concepts using MeSH metathesaurus and latent semantic analysis. In addition, we have distributed a questionnaire among two classes of users; medical experts and non-experts, in order to evaluate the ranking and the relationship among successive results in the hit lists. The feedback data from participants shows that our proposed model performs well and serves the needs of participants.

In the second experiment, we have evaluated the proposed categorization method on OHSUMED categories. We compared the results with state-of-the-art methods in the literature. In addition, we performed sensitivity analysis to understand how classification parameters affect the categorization process. The results indicate that ROLEX-SP outperformed other methods, such as C4.5 and OLEX, when applied to free-text medical documents with minimal label description.

1.3 Dissertation Outline

The rest of this dissertation is organized as follows: Chapter 2 exhibits background information on the main topics covered in this dissertation. This Chapter is divided into three parts. The first part presents the problem of information retrieval. In addition, it provides detailed information about related theoretical work in this field. The second part describes an overview of text categorization approaches, major issues in text categorization, and an overview of learning algorithms. The third part of Chapter 2 provides an overview of existing personalization methods and tools in the literature.

Chapter 3 introduces the proposed similarity model. It includes detailed description of the main goals to introduce this model. Furthermore, we extensively explain the method and the design of the proposed retrieval and ranking techniques. Next section, in this Chapter, provides detailed information about the experiment including: experiment setup, corpus, results and comparison with related methods. Moreover, we describe the application of personalization on MedicoPort to complete the retrieval process. The last two sections provide a discussion of output performance issues and a conclusion statement of this research.

Chapter 4 includes information about the categorization method introduced in this dissertation. Section 1 introduces the concept of lexical syntactic patterns. Section 2 discusses the framework of the proposed method. Sections 3 and 4 provide detailed algorithmic description of the induction and validation phase of the proposed method. Section 5 describes the time complexity of the proposed learning method.

Section 6 and 7 describe the experiments performed to show the effectiveness of the proposed technique as well as a discussion of the research outputs.

Chapter 5 represents the conclusion of our research. It includes a discussion about the contributions of this research in academic and practical fields. Furthermore, it discusses comments on results in addition to the research limitations and future research.

CHAPTER 2

BACKGROUND

This Chapter provides theoretical and technical information of the main topics covered in this dissertation. It is divided into three sections: Section 1 gives theoretical information on information retrieval and domain specific information retrieval problems. Section 2 has particularized to text categorization approaches and techniques, specifically rule-based categorization. Finally, section 3 provides a literature review of existing personalization methods and tools.

2.1 Information Retrieval Problem Definition

Information Retrieval (IR) is an application of natural language techniques that is focuses on modeling, indexing, designing, and retrieving widely distributed information chunks [19]. The goal behind the application of IR systems is to facilitate obtaining interested and relevant information by IR users. Technically, the information retrieval system is responsible to collect information that expected to be useful according to the user's description; user query.

In general, two elements might affect the overall performance of the information retrieval process: user task and the logical view of a document as a basic entity. User task include the formalization of query and browsing documents. In other words, the performance of an IR output increased as users formulate their needs by specifying a set of terms, query, which reflect the semantics of the requested information.

The logical view of a document represents the definition of the key attributes of a text document such as word, phrase, or paragraph. In the jargon of information retrieval and data mining, researchers used the term *Feature* or *Feature Attributes* to refer the key searching attribute.

In brief, the basic elements of the design of IR systems are: the formulation of user query, documents attributes, retrieval process, and browsing of relevant documents. These four elements represent an abstract view to the information retrieval problem.

2.2 Classical Information Retrieval Models

In this section, we provide description to the classical information retrieval models. In addition, we summarize the basic mathematical definitions in order to clarify the functionality of retrieval models in capturing correspondence between a given user query and a set of documents.

2.2.1 Boolean Model

The Boolean model is a plain method to retrieve information where a query is formalized as a Boolean expression. The definition of Boolean model obeys the semantic of set theory and Boolean algebra. An indexed term, in Boolean model, is considered either available or not available in a document and the weight of indexed terms takes on of the two values, 1 or 0.

The similarity between queries and indexed documents is computed as follows [19]:

"Given binary weights of indexed terms $w_{i,j} \in \{1,0\}$ *and a query q as Boolean expression, let* $\vec{q}_{dnf}$ *be the disjunctive normal form for the query q. Further, let* $\vec{q}_{cc}$ *be any of the conjunctive components of* $\vec{q}_{dnf}$*. The similarity of a document* d_j *to the query q is defined as":*

$$sim(d_j, q) = \begin{cases} 1 & if\ \exists \vec{q}_{cc} \mid (\vec{q}_{cc} \in \vec{q}_{dnf}) \wedge (\forall_{ki}, g_i(\vec{d}_j) = g_i(\vec{q}_{cc})) \\ 0 & otherwise \end{cases}$$ **(EQUATION 2.1)**

where

$$g_i(\vec{d}_j) = w_{i,j}.$$

In other words, if $sim(d_j, q) = 1$ then the model considers that document d_j is appropriately close or similar to the query q; binary choices: relevant or irrelevant.

In [20], a method of generalizing Boolean model has been introduced in order to address the problem of producing partially related set of documents. The proposed enhancement is based on attaching numeric weights to terms instead of binary ones. Then, the similarity is computed as fractional of the number and weight of terms in a given document. Finally, the similarity is compared with a threshold to foresee the relevancy to a given query.

Boolean model is more efficient as a data model in comparison with informational models. But it suffers from major problems in the field of information retrieval since Boolean model requires exact matching; this situation leads researchers to more sophisticated models such as vector and probabilistic models. Commercially, Boolean models used to implement structured query languages to represent Boolean expressions.

2.2.2 Probabilistic Model

The probabilistic information retrieval model was proposed by Robertson and Jones [21]. It attempts to use probabilistic framework in order to solve IR problems. The main idea is to create two sets of documents on the basis of a given probabilistic formula. The first set represents documents that have been considered to be probably relevant to a given user query, while the other one represent irrelevant documents according to the given formula; clustering documents into relevant and irrelevant ones.

The similarity between a user query and a document is calculated as follows [19]:

"The index term weight variables are all binary $w_{i,j} \in \{0,1\}$, $w_{i,q} \in \{0,1\}$. *A query q is a subset of index terms. Let R be the set of documents known to be relevant. Let* $\overline{R}$ *be the complement of R. Let* $P(R \mid \vec{d}_j)$ *be the probability that the document* d_j *is relevant to the query q and* $P(\overline{R} \mid \vec{d}_j)$ *be the probability that* d_j *is non-relevant to q. The similarity of the document* d_j *to the query q is defined as the ratio":*

$$sim(d_j, q) = \frac{P(R \mid \vec{d}_j)}{P(\overline{R} \mid \vec{d}_j)} \qquad \textbf{(EQUATION 2.2)}$$

$$sim(d_j, q) = \frac{P(\vec{d}_j \mid R) \times P(R)}{P(\vec{d}_j \mid \overline{R}) \times P(\overline{R})}, \qquad \textbf{(EQUATION 2.3)}$$

where $P(\vec{d}_j \mid R)$ refers to the probability of choosing arbitrary document d_j from the set R of related text documents. Further, $P(R)$ refers to the probability that an arbitrary document chosen from the set of all collected documents is relevant to q.

The main advantage of this model as compared to Boolean one is its capability of sorting documents according to their relatedness to a user query. However, the probabilistic model still suffers from many problems. The main one is the ignorance of term importance among other terms; the model gives all terms the same weight.

In addition, the model has to guess the initial separation, relevant and irrelevant documents.

2.2.3 Vector Model

The vector model (or vector space model VSM) is widely implemented in information retrieval systems. It relies on representing documents as a vector of term weights and computing the similarity on the basis of the distance between two vectors; or the cosine angle. VSM was first introduced in [22]. VSM addresses the problem of retrieving partially relevant documents by using non-binary weights to index terms.

According to VSM, the similarity of document d_j to query q is calculated as follows [19]:

"The weight $w_{i,j}$ associated with a pair (k_i, d_j) is positive and non-binary, where k_i represents term i in a given document d_j. Further, the index terms in the query are also weighted. Let $w_{i,q}$ be the weight associated with the pair (k_i, q), where $w_{i,q} \geq 0$. Then, the query vector $\vec{q}$ is defined as $\vec{q} = (w_{1,q}, w_{2,q}, \ldots, w_{t,q})$ where t is the total number of index terms in the system. The vector for a document d_j is represented by $dj = (w_{1,j}, w_{2,j}, \ldots, w_{t,j})$. That is":

$$sim(d_j, q) = \frac{\vec{d}_j \bullet \vec{q}}{|\vec{d}_j| \times |\vec{q}|}. \qquad \textbf{(EQUATION 2.4)}$$

$$= \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w^2_{i,j}} \times \sqrt{\sum_{j=1}^{t} w^2_{i,q}}} \qquad \textbf{(EQUATION 2.5)}$$

The model represents the document and the query as two vectors. Then, it computes the angle between these two vectors. A small angle indicates high similarity while a large one indicates large distance between vectors.

Our work, in this research, is based on vector model as a base to compute the similarity between vectors. Unlike traditional definition of VSM, we define more than one feature (see Chapter 3) associated to an entity (document or query) and, then, we compute the similarity between features using traditional vector model definition.

2.3 Categorization of Text Documents

Categorization is the process of classifying entities into classes of knowledge. Some resources distinguish between the classification and categorization concepts since categorization describe the classification of text documents after the retrieval task completes. There are three major approaches for automatic text classification Text categorization, document clustering, and document classification [23].

1. Text categorization is the process of classifying text into classes of information based on a training sample set. The sample training set is used to

learn semantic and/or syntactic characteristics of text documents. Then, a classifier is constructed to automatically categorize new incoming text documents. In machine learning jargon, text categorization is called supervised learning because the learning process supervised by a labeled training set of documents.

2. Document clustering is an unsupervised learning process that does not requires a training set. A document clustering algorithm tries to learn how to separate similar document by, for example, measuring the distance among them such as k-nearest neighbors algorithm

3. The library classification approach based on intellectually created controlled vocabulary. Documents classified under categories using simple algorithms that detect the general topic covered by them.

There are some major similarities among these approaches including text document preprocessing and utilization of text features. On the other hand, the major differences among them are the learning algorithm (supervised or unsupervised) and the feature selection process.

This dissertation concerns text categorization with supervised learning algorithm to induce text features. Next sections provide background information about common learning techniques, major issues in learning algorithms, lexical syntactic patterns as classification features of free text, and rule based categorization.

2.3.1 Learning Techniques

In this section, we provide an overview of classification techniques. Decision-Tree based, Bayesian, and classification by Backpropagation represent common techniques in which recent research on data mining has built on. In this dissertation, we are focusing on text categorization rather than other types of data. Thus, we describe classical techniques learning methods to understand learning algorithms of text based features.

2.3.1.1 Classification by decision tree

Decision tree (DT) is a data structure consists of a set of nodes and arcs. In the jargon of decision-tree classification, non-leaf nodes represent attributes or features while leaf nodes hold class labels. The arcs in a decision tree represent attributes values, which control the way a learning algorithm set up a path from the root node to one of the leaves.

The process of building a decision tree requires no domain knowledge [24]; this fact makes decision-tree based classification very popular with respect to other classification methods. In addition, the learning algorithm of DT-based classification handles high dimensional data efficiently in terms of memory requirements and time complexity.

For text categorization, decision trees have been widely used to extract rules from training sets. DT learning algorithms implement many attribute or feature selection

algorithms such as Information Gain and Gain Ration, which facilitate selecting attributes to represent class labels. Consequently, the rules are constructed as logical conditions of attribute values on paths chosen by attribute selection algorithms.

2.3.1.2 Bayesian classification

Bayesian methods rely on Bayes' theorem; a statistical model to predict to which class a given data record is belong to. There are two widely used versions of Bayesian models: Naïve and belief network. Both techniques implements Bayesian statistics but they deal with attribute dependency in a different way [25].

Naïve classifiers assume that the effect of one attribute is independent from the value of other attributes in the training set. This assumption simplifies the classification process but makes naïve classifiers inefficient in domains where attributes (or features) are naturally dependent. In text categorization, features (such as terms) are considered naturally dependent. For instance, synonyms and antonyms are examples of such dependency.

Bayesian belief networks are graphical models consists of variables and conditions. The network is trained by modeling a set of example cases from the training set. In contrast with naïve Bayesian approach, belief networks assume that attribute are not all independent; "*a variable is conditionally independent of its non-descendants in the graph*".

2.3.1.3 Classification by Backpropagation

Backpropagation learns classes of data using neural network algorithm. The network is a set of input and output nodes and the connections among these nodes are weighted. The basic idea of learning in neural network is that learning by adjusting the weights to predict classes of the input data.

A Backpropagation classifier learns correct classes by performing weights computations in order to reduce prediction error (mean squared error) in comparison with actual classification. The weight's modification takes place from the data-generation level back to the first level that accepts input; which increase the accuracy of classification [23].

2.3.2 Overview of Major Learning Issues in Text Categorization

In this section, we describe major issues that have been noticed in domain-specific text categorization. In contrast with other data types, these issues clearly affect the categorization of text documents since text entities might holds semantic connections.

2.3.2.1 Multi-Class Classification (MCC)

In data mining literature, there are two types of classification methods: binary and multi-class classifications. Binary classification techniques assign a tuple to one of two classes on the bases of whether it has a specific property or not. For example, a

medical testing to determine whether a patient infected by a specific disease or not; the property in this example is the disease.

On the other hand, multi-class classification is the process of assigning tuples to their appropriate class among many other classes. In the common case, given ***n*** classes, trains ***n*** classifiers, one for each class, a data-record is categorized under the closest class in terms of positive distance [24]. In other words, the tuple is assigned to only one class among many available ones.

As a special case, in some problems such as text categorization, an entity might be related to more than one class at the same time. For example, suppose we want to classify a medical document that describes the symptoms and treatment of certain disease. Suppose, in our domain, we define three classes: Symptoms, Treatment, and Tests. In this case, the document has to be classified into two classes at the same time. After the learning task completes, a classifier associated with each class will be build. In this case, a relaxed procedure is required to allow more than one classifier to adopt this document.

In the literature of text categorization, a useful solution, in terms of performance, to this problem is the application of rule-based categorization in which a classifier is defined as a set of representative rules.

2.3.2.2 Feature Imbalance

Many feature selection algorithms have been proposed for the purpose of enhancing the performance of text-based classifiers [24]. In text-based categorization, the

dimensionality of text data is normally high. Traditional term-based features suffer from the problem of feature imbalance [12, 13], in which the classifier biased toward frequent terms.

Feature imbalance negatively affects the performance of text classifiers; increase the number of misclassified documents. The application of semantic features to categorize text documents reduces the effect of feature imbalance problem. In this research, we study the effect of applying lexical syntactic patterns as a classification feature.

2.3.3 Lexical Syntactic Pattern

A lexical syntactic pattern (LSP) is a natural language expression consisting of noun-phrases and domain lexicon concepts. LSP have been proposed by [26] to extract relations among concepts such as synonyms. In this research, we used to extract Hearst-like patterns to construct rule-based classifiers. LSP is more robust in representing class properties than term-based since LSP do not bias toward frequent terms [27]; resulted in feature imbalance problem. For instance, the terms "Symptoms" and "Diagnosis" are infrequently appear in medical documents. On the other hand, LSP are capable to represent multi domain concepts in the same pattern just like phrase-based representation. Unlike phrase-based, LSP is not affected by concepts positions.

Our hypothesis, in this research, is measuring the capability of LSP to increase the coverage as well as the accuracy of rule-based classifiers and, at the same time, reduce classification error resulted in term and phrase-based feature.

In this dissertation, we propose to implement lexical syntactic patterns as a classification feature in a rule-based categorization method. Our method, called ROLEX-SP, is the first method to apply lexical-syntactic patterns as a feature to represent free text. Lexical syntactic patterns were able to represent single terms, conjunctions of terms, and more complex relation among many domain concepts that cannot be represented as a phrase.

2.3.4 Rule-based Categorization

Rule-based categorization is the process of sequentially classifying text documents into classes of knowledge. It is sequential since classifiers attempt to learn one rule at a time. Thus, a learning algorithm is defined to induce rules from training sets based on some performance criteria such as: F-measure, coverage metric, or accuracy metric.

A rule, in this context, is an expression of the form *If* (*condition*) *Then* (*conclusion*) where, the *condition* part represent a logical condition consists of set of attributes; features. The *conclusion* part represents the consequent of a rule.

A learning algorithm is responsible to scan training sets for the purpose of constructing rules. Thus, the learning algorithm must have enough information about the classification features before the learning process starts. This information include description of the available categories in the training sets; information that, hopefully, uniquely identified these categories.

2.4 Personalization

Personalization is the process of tailoring search results based on a specific user's interests. In this section, we provide a review of common personalization methods and tools. The literature includes two types of methods: Re-ranking and personalization.

2.4.1 Personalization Methods

Documents re-ranking is defined as the process of sorting the set of documents retrieved by an IR system based on a specific feature. In this subsection, we review existing methods and algorithms of re-ranking text documents in the domain of information retrieval and search engines.

2.4.1.1 Query-Specific Document Clustering

This method relies on the cluster assumption, which states that related documents tend to appear in the same cluster of documents. If a specific collection of

documents satisfied this assumption, then related documents will be apart from irrelevant ones.

Relevant documents that might be ranked at the bottom of the retrieved hit-list (based on inverted file) will be assembled together with other related documents (in the same cluster), which improve the effectiveness of ranking in IR systems. The actual effectiveness of cluster-based search is retrieving the cluster that best fit the query [28]. The point is to consider the query as a representation of user's interest.

The main drawback of this technique is to find the relevant cluster to a specific query. Query-specific document clustering requires queries to be expressive; full description of user's needs. Thus, this technique is not effective for short-queries or for queries that cover more than one topic (topic-overlapping).

2.4.1.2 Labeled Propagation (LP)

Labeled propagation is a semi-supervised learning algorithm to re-rank text documents. It is semi-supervised since the learning algorithm must know few relevant documents to a given query. The algorithm, then, propagate by exploring the similarities among all retrieved documents; the similarity between know and unknown ones [29].

Unlike KNN (K-nearest Neighboring), LP breaks the nearest-rule when a set of unknown documents are close (similar) enough to each other. Specifically, the algorithm considers unknown documents in the high-density area to cover the same labels of each individual's nearest known neighbor [29].

For well known queries in some IR systems, such as question-answering systems, this method is effective to re-rank relevant documents without users' intervention. But, in general IR systems, it is hard or even impossible to determine relevant documents since user-query can not be predicted.

2.4.1.3 Graph-based Method

Graph-based re-ranking is a technique to construct a graph in which nodes represent documents and links represent the relationship among documents. Previous works focus on developing models to measure the relationship among documents.

Kurland and Lee [30, 31] performed re-ranking based on measures of centrality in the graph formed by arcs induced by language model scores, through a weighted version of PageRank algorithm and HITS-style cluster-based approach. In [32], authors introduced a method to improve search based on a combination of results from text search and authority ranking techniques. The proposed graph in [32], which is called affinity graph, based on Kurland and Lee's research with links induced by a modified version of cosine similarity using vector space model.

In [33], authors used score regularization to adjust document retrieval re-rankings from an initial retrieval by a semi-supervised learning method. In [34], authors built a latent space graph based on content and explicit links information. Explicit information is extracted through latent analysis or similar statistical methods to represent links between documents.

2.4.1.4 Structural Re Ranking Method

A structural re-ranking approach to ad-hoc information retrieval: The idea is to re-sort the retrieved set of documents by investigating the asymmetric relationships among them. They assumed the language-model arcs derived from a document assigns high probability to the text of another document. One purpose of this method is to avert bias toward long-text documents. The proposed re-ranking criterion based on measures of centrality in the graphs is formed by generation links [30]. This method requires using structured text documents.

2.4.1.5 Maximal Marginal Relevance Method

This method focuses on re-ranking documents retrieved by short-queries. The idea is to expand short-queries according to user feedback to improve the retrieval effectiveness of these queries. In [35], experiments have been performed to refine the set of documents used in feedback; i.e. after retrieving the hit list, the system shows candidate relevant documents to the user (they might be related to different topics). Authors performed these experiments using Boolean filters constructed manually as well as closeness restrictions. Next, they predict the relation among terms by automatically extracting term co-occurrence information. Experimental findings indicated that refining the collection of documents used to expand queries improved the overall performance in terms of average precision and precision at the top twenty retrieved documents.

2.4.1.6 Latent Semantic Analysis (LSA)

Latent semantic analysis (LSA) [36] has been used for document representation. Latent Dirichlet Allocation (LDA) was first introduced by [37], is a probabilistic-based modeling relies on extracting relations among tuples in different entities. It has been implemented on different areas including: text classification and clustering [38], information discovery, [39] [40] and information retrieval [41]. In this model, each topic is represented by a set of concepts in which each concept is corresponding with a weight to quantify its contribution to the topic. In [41], authors described large-scale information retrieval experiments by using LDA.

LSA-based method suffers from an incremental build problem. Normally adding new documents to the corpus needs to "be folded in" to the latent representation. Such cumulative addition is unable to catch the co-occurrences of incoming new documents. In some cases, the algorithm disregards newly incoming terms. Thus, the performance of the applying LSA is decreased as more documents are coming and required a lot of computations; high complexity.

2.4.2 Personalization Tools

In this subsection, we provide a brief description of tools that have been widely used to personalize searching results.

2.4.2.1 Letizia

Letizia [42] is a web explorer that tracks browsing behavior of users. Letizia relies on implicit user feedback; tracking user browsing of web pages. The tool detects user's clicks, and then uses a set of heuristics to construct preferences for users. For instance, book-marking a web page is a strong indication for the user's concerns in that page.

2.4.2.2 NewsDude

NewsDude collects news essays using a "*speech interface*". The tool harvests news from *Yahoo! News* as a single source of information. The tool has a preliminary training set of articles in which each user is interested in. This set is huge enough to handle different cases. The length of time a user listening to articles provides implicit indication of interest to the tool. For short-time sessions, the similarity computations are based on classical vector space model, and for long-time sessions, a naïve Bayes classifier is used [43].

2.4.2.3 Syskill & Webert

Syskill is software that relies on the classical profiling technique of creating users profiles. Users provide their preferences information and the tool identifies pages of interests according to them. The learning process starts by converting HTML source into positive and negative indicators. Then, the tool uses a binary classifier to separate relevant ones, such as k-nearest neighbor algorithm [44].

2.4.2.4 LIBRA

LIBRA is a classification method of structured text. LIBRA uses to construct its contents by extracting product description from Amazon. LIBRA relies on structured text; the product's specifications are partitioned into segments in order to deal with information in disunite manner. Such segments represent title, abstract, and authors' information. For more details about the advantages of applying structured contents, see the work in [45].

2.4.2.5 IfWeb

IfWeb keeps information about users by creating a network of weighted terms. The nodes in the network point to concepts and the links represent occurrence of these concepts in indexed documents. ifWeb implements explicit feedback in which users specify their needs explicitly. The tool, also, differentiate between two types of interests: positive and negative. This way, the tool enhances the accuracy of stored information since it considers interests as well as disinterests of users [46].

2.4.2.6 SiteIf

SiteIf is software that induces user's concerns from multilingual sources of information. The learner analyzes the retrieved web pages in order to modify profiles associated to system users. By this framework, the tool predicts the set of documents that match the interests of a given user. SiteIF is implements a semantic network of profiles and documents just like ifWeb. The difference between both tools is that in SiteIF user interaction is not allowed [47].

CHAPTER 3

A MODEL BASED ON MULTI-FEATURES TO ENHANCE MEDICAL DOCUMENT RETRIEVAL

Web-wide search engines are growing in popularity as they offer a mechanism to locate information in the Internet. Large number of Internet users accesses the abundant web content to locate medical and health information that is of interest to them. Recently, medical information retrieval systems gain an increased attention, many specialized databases and tools, such as UMLS [48] (Unified Medical Language System), have been offered to public research providing a source to ontologies and metadata about medical terms, which opened the doors for developing search engines targeted to medical and health domain.

In this Chapter, we present a model for extracting the semantic relation among medical and healthcare documents. The purpose is to maximize the contextual retrieval and ranking performance with minimum input from users. We developed and evaluated a medical search engine that relies on a multi-features similarity

model. The indexed documents are represented as a network that reflects the semantic relations among documents to assess topical ranking. The proposed technique based on expanding terms with related MeSH concepts in order to extract relations among medical and healthcare documents

The evaluation measurements include: Recall, Precision and R-Precision. We used OHSUMED collection to evaluate our work with runs submitted to TREC-9. We provide a comparison with the top 5-runs, which achieved highest average precision scores and a similar method in terms of expansion concepts with MeSH related terms. In addition, we used a questionnaire-based evaluation for measuring the effectiveness of the ranking task.

The results indicated that the proposed model achieved higher average precision in compare with top-scored runs submitted to TREC-9 and achieved higher interpolated average precision per query as compared to KELSI (Knowledge-Enhanced Latent Semantic Indexing). Furthermore, a questionnaire-based experiment showed that the retrieved hit-lists by the proposed model satisfied the requests of participants. In addition, the questionnaire's participants assessed the relationship among successive documents as strong.

This Chapter is organized as follows: section 3.1 explicates the objectives of the proposed model. Section 3.2 explains the methodology and mathematical foundations of the model. Section 3.3 shows the design of MedicoPort. Section 3.4, describes in details the experiment setup, results and comments experimental

findings. Section 3.6 provides a discussion on the experimental results. Section 3.7 describes the contribution and a summary of the enhancements achieved.

3.1 Model Objectives

We propose an effective mapping model dedicated to the medical and healthcare domain to create document network that relates medical documents according to the semantic relations among them. We called this a medical information retrieval (MIR) model. The model was inspired by the definition of gravitational force among bodies [49]. The goal is to develop a searching method to serve the needs of non-expert users in medical domain.

The contributions of this Chapter can be summarized as follows:

- The similarity model combines multiple semantic features to model the relationships among documents containing medical and healthcare information. The purpose is to overcome the frequency anomaly of traditional methods and retrieve more accurate results by shrinking the hit-list via reduction of the maximum number of relevant documents, which results in high precision.
- The system facilitates medical and non-medical searching by expanding user queries with related concepts through the use of a specialized medical lexicon and a metathesaurus. The system then attaches user queries to a network of documents and computes similarity based on a set of predefined semantic features.

- The ranking method sorts highly relevant documents toward the top of the hit-list. The ranking task is implemented on top of a semantic document network created to rank documents according to their topics.

3.2 Method

In this section, we provide a detailed description about the assumptions and the features that have been used to develop the similarity model. Furthermore, we also illustrate the translation of the features into vector-space representation.

3.2.1 Assumptions

We have developed a set of assumptions to model the relationship among documents; these assumptions have been extracted from the work of other researchers and our observations during experiments. They can be summarized as follows:

1. Documents that share medical concepts in their identification text (such as the title) approximately belong to the same topic. This observation has been tested by [50, 51]. It is approximate because these anchors might not be available or might refer to partial relations among topics.
2. The weight of medical concepts in a document reflects the importance of the document in the collection. A document that includes large numbers of medical concepts as compared to non-medical terms seems more professional and

relevant to the medical domain and, therefore, deserves a higher ranking place than a document with fewer medical concepts.

3. Documents that share a set of medical concepts are connected to each other in a direct relation. The higher is the number of common medical-domain concepts, the higher is the similarity among documents. Unlike the first assumption, here we measure the similarity among domain concepts in the whole documents; the first assumption measures the similarity among identification text only.

To evaluate the effectiveness of the proposed model, we assume that all documents in the sample collection belong to a single category. Whenever a new query comes, the system considers the query as a single node and links it to its related documents.

3.2.2 Basic definitions

Definition (1) *Let* t *be the number of all terms stored in the inverted file and* k_i *be a generic index term. Let* $K = \{k_1,...,k_t\}$ *be the set of all index terms in a collection of documents. A weight* $w_{i,j} > 0$ *is assigned to every index term* k_i *of a document* d_i. *Document* d_i *is associated an index term vector* $\vec{d}_j$ *represented by* $\vec{d}_j = \{w_{1,j}, w_{2,j},..., w_{t,j}\}$, *where* $w_{i,j} = 0.5 + \frac{0.5 \times freq_{i,q}}{\max freq_q} \times \log \frac{N}{n_i}$ *(tf-idf scheme), where* $freq_{i,q}$ *is the frequency of the term* k_i *in the query q,* n_i *is the number of documents in which the term* k_i *appears, and N is the total number of documents [19].*

For the purpose of our work, we want to extract terms relevant to the medical domain; in other words, we want to distinguish medical terms from other generic terms or terms from different domains. This situation leads us to define a Boolean set to identify the class of indexed terms, medical or not.

Definition (2) *Let* $M_j = \{m_{1,j}, ...m_{t,j}\}$ *be a set of Boolean values (0, 1) associated with document* j *in which* $m_{i,j} = 1$ *indicates that term* k_i *is a medical term and* $m_{i,j} = 0$ *indicates that term* k_i *is a generic term.*

To facilitate the implementation of the first feature, we define the set C_i associated with document *j* as containing anchor terms, that is, terms that have used in the identification text to documents as follows.

Definition (3) *Let* $C_j = \{c_{1,j}, ..., c_{t,j}\}$ *be a set of Boolean values (0, 1) associated with document j in which* $c_{ij} = 1$ *indicates that term* k_i *is an identification text of document j; otherwise,* $c_{ij} = 0$. *Term* k_i *is part of the anchor text, such as the title.*

Finally, to facilitate the mathematical description of the proposed features, we define the function ρ that takes a set of sets as input and produces a vector that represents the multiplication of elements from different sets relying on the same position. This function has been used to identify medical terms and anchor terms mathematically by combining the set of weights *d* with the Boolean sets *M* and *C*.

Definition (4) *Let* $v = \{e_1, \ldots, e_t\}$ *be a set of t-elements and* $v(i) = e_i$ *denotes the* i^{th} *element in the set v. The function* ρ *is defined as follows:* $\rho(v_1, ..., v_n) = \vec{g}$, *where* $\vec{g}$ *is a vector in t-dimensional space such that* $\forall i \le t, g(i) = v_1(i) \times v_2(i) \times ... \times v_n(i)$, *where* i *refers to the* i^{th} *element in the sets* $\{v_1, \ldots, v_n\}$ *and the* i^{th} *component of vector* $\vec{g}$.

3.2.3 Degree of Attractiveness Feature

The first part of the model analyzes how close topics that belong to two different documents are to each other. According to the first assumption, two documents are considered close to each other if their titles and/or links (i.e., anchors) share some common knowledge. For this purpose, we want to construct a vector that represents medical identification text. For document j, we define the vector g as follows.

$$\vec{g}_j = \rho(C_j \times M_j \times d_j) \qquad \textbf{(EQUATION 3.1)}$$

The resulting vector $\vec{g}_j$ represents the weights of medical terms in the identification text in t-dimensional space associated with document j. Notice that if a term k_i is a medical term in the document title, this implies that the values of $M_{i,j}$ is 1 and $C_{i,j}$ is 1 as well. Otherwise, if k_i is not an anchor term and/or if it is not a medical term, the value associated with term k_i in the vector $\vec{g}_j$ is 0.

Given two documents d_i and d_j, the degree of attractiveness, which is represented as $G(d_i, d_j)$, is defined as follows.

$$G(d_i, d_j) = 1 + (\vec{g}_i \bullet \vec{g}_j) \qquad \textbf{(EQUATION 3.2)}$$

which represents the Euclidian distance between vectors $\vec{g}_i$ *and* $\vec{g}_j$ in t-dimensional space.

3.2.4 Mass of Document Feature

A document's mass indicates the importance of the document in the collection. Unlike the weirdness factor defined in [52], which is directed to measure the differences in the distribution of a specific term in domain-specific and generic text, document's mass measures the weight of domain terms in a specific document rather than how the distribution of these terms affects term-weighting. According to the second assumption, we define the scalar mass to represent this feature. For a given document d_i, we obtain the vector of medical-terms weights from the following function:

$$\vec{Ms}_i = \rho(M_i \times d_i) \qquad \textbf{(EQUATION 3.3)}$$

The mass of document d_i, which is represented as $Mass(d_i)$, is computed using the following formula.

$$Mass(d_i) = \frac{\sum_{k=1}^{t} Ms_{k,i}}{\sum_{k=1}^{t} w_{k,i}} \qquad \textbf{(EQUATION 3.4)}$$

The goal behind this feature is to give professional documents or documents with frequent medical terms a higher score than non-professional ones or documents with few medical terms.

3.2.5 Distance Feature

Finally, we want to compute the distance between two given documents. Therefore, we represent every document by a medical feature vector consisting of only medical concepts. Then, we compute the cosine value between vectors. Therefore, to compute the distance between two vectors from documents d_i and d_j represented in t-dimensional space, we apply the following formula.

$$\vec{D}_i = \rho(M_i \times d_i) \qquad \textbf{(EQUATION 3.5)}$$

$$\vec{D}_j = \rho(M_j \times d_j)$$ **(EQUATION 3.6)**

$$Distance(d_i, d_j) = 1 - (\vec{D}_i \bullet \vec{D}_j)$$ **(EQUATION 3.7)**

Because we want to create an inverse relation between overall similarity and distance, we subtract the distance from one to compute dissimilarity. Therefore, the lower the distance between the two domains vectors is, the higher is the similarity between these vectors.

3.3 MedicoPort Design

In this section, we provide a description of MedicoPort [1] design. As stated before, we have redeveloped MedicoPort in order to integrate it with our proposed method. In particular, the design of the modified version of MedicoPort consists of seven modules that are responsible for receiving user queries as input, performing text operations, expanding queries with medical related concepts, searching the inverted file for relevant documents, and ranking the hit-list.

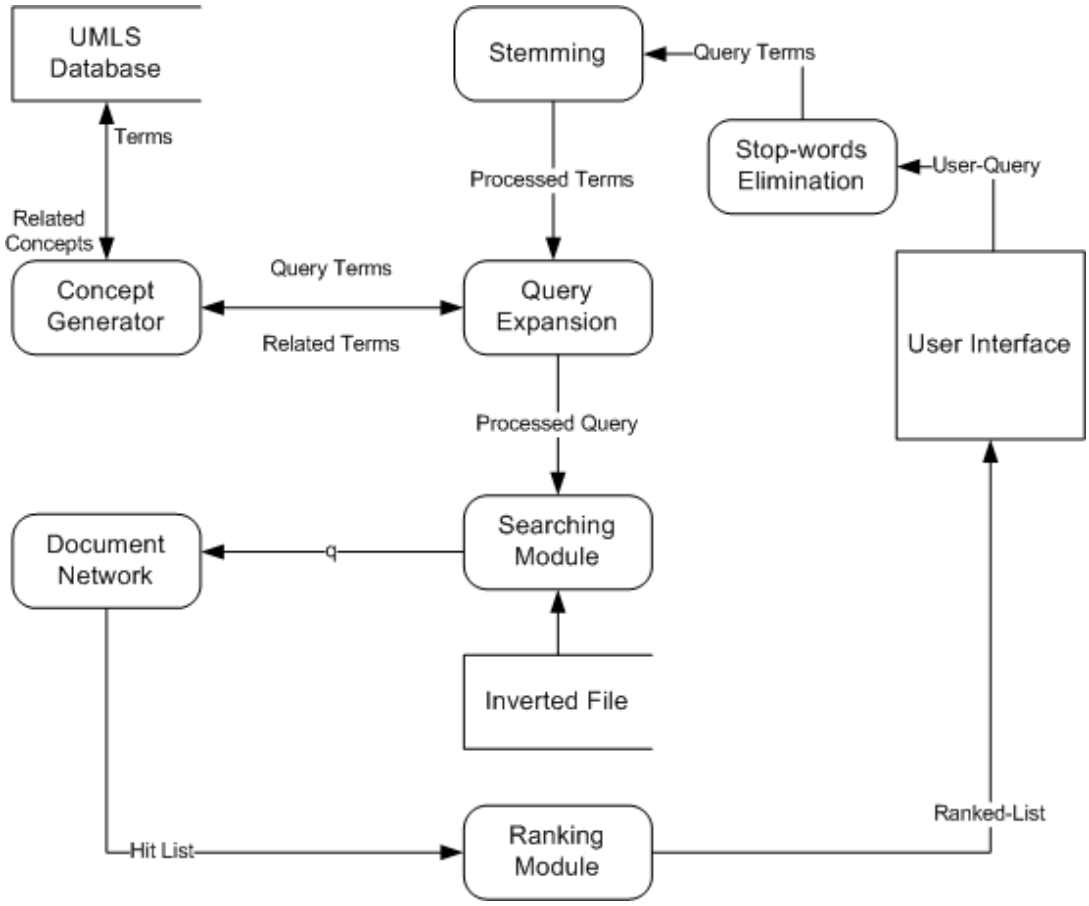


Figure 3.1- MedicoPort Design Modules

3.3.1 Design Issues

MedicoPort has been design to take advantage of the Unified Medical Language System (UMLS). UMLS is a specialized database that maintains medical concepts and their semantic relationships. The main constituents UMLS are: specialist lexicon of medical concepts, semantic network, and metathesaurus. The system uses UMLS for crawling medical documents, indexing, and query expansion.

As a medical domain search engine, MedicoPort built on the top of a topical crawler, called Lokman [92], which is responsible for crawling medical documents. The design of Lokman restricts the crawler to harvest only medical documents by choosing domain relevant hyperlinks during the crawling phase with the assistant of UMLS. Thus, Lokman filters out irrelevant web pages, which increase the quality of indexed documents

3.3.2 MedicoPort Structure

In this section, we provide a description of MedicoPort structured modules. The description highlights the general specification of the design modules.

3.3.2.1 Text Operations

To reduce the set of representative words, the system eliminates stop-words using the Princeton English stop-word list [53]. Furthermore, the stemming module is used to diminish distinct terms to their common grammatical stems or roots. Notice that

the inverted-index has been cleaned using stop-words during construction and contains terms and their stems.

3.3.2.2 Query Expansion

To facilitate the searching process to non-medical users, the query expansion module is responsible for automatic reformulation and expansion of a query with domain concepts. The goal behind query expansion and reformulation is to make the search engine capable to retrieve relevant documents with small amount of information provided by a user and minimize the effect of language ambiguity encountered by non-medical users.

As a result of applying this task, the system can retrieve documents similar to original user's query in addition to documents that are close to be related to the search topic. This way, MedicoPort enables retrieving relevant documents with a small amount of knowledge input by user's query.

3.3.2.3 Concept Generator

The concept generator module consists of a set of functions that enable contacting the UMLS (Unified Modelling Language System) database [48] to retrieve specific information about these terms such as the type of terms (i.e., medical or non-medical), synonyms, contextually-related terms, or partially-related terms.

In particular, concept generator module accepts incoming terms from query formulation module and formulates a UMLS query using XML format. Figure 3.2 shows an example query to retrieve the concept identifier of the concept “Breast Cancer”. Figure 3.3 show an example query to retrieve the synonym terms.

```
<?xml version="1.0"?>
<findCUI version="1.0">
<conceptName> breast cancer </conceptName>
<language > ENG </language >
<exact/>
<noSuppressibles />
</ findCUI>
```

Figure 3.2 "FindCUI" Query

```
<?xml version="1.0"?>
<getRelations version="1.0">
<cui> C0033572 </cui >
<rel > SYN </rel >
</ getRelations>
```

Figure 3.3 "GetRelations" Query

The following table shows the concept relations retrieved from UMLS relevant for the query “breast cancer.”

Table 3.1 - UMLS Term Relations for The Query "breast cancer"

Concepts obtained from UMLS	*Relation*
Breast Carcinoma	SYN
Cancer of Breast	SYN
Mammary Carcinoma	SYN
Carcinoma of Breast	SYN
Malignant Neoplasm of Breast	PAR
Malignant Tumor of Breast	CTX

3.3.2.4 Searching Module

Searching for relevant documents is performed on a collection of medical documents that have been harvested by Lokman crawler. The purpose behind searching topical documents is to maximize the quality of the retrieved hit-list.

A query might consist of a single word or multiple words. After expanding the query with related concepts, the searching module computes similarity using the MIR model. Then, the system sends the list of relevant documents to the ranking module to rank the hit-list according to their relevancy to the given query.

Notice that, terms weights are computed using the TF/IDF formula. For query terms, MedicoPort searching module uses a weight factor in order to give original terms higher priority over other related concepts such as synonyms and partially related terms. The weight factor attached to exact query term is 1; the same TF/IDF weight.

While the weight factor given for synonym concepts is 0.9, for contextually related concepts is 0.6, and for partially related concepts is 0.3.

3.3.2.5 Document Network and Ranking

A document network is a network in which the members are documents, and the links represent similarity among them. MedicoPort constructs the network during pre-processing phase. Thus, the system computes the similarity between retrieved documents with respect to user queries by attaching the user query to a network of links between relevant documents, which are based on the similarity scores.

$$Similarity(d_i, q) = G(d_i, q) \times \frac{Mass(i) \times Mass(q)}{Distance(d_i, q)^2} \qquad \textbf{(EQUATION 3.8)}$$

The system then ranks retrieved document according to their position in the network. Figure 3.4 shows part of the document network created during experiments; circles represent documents, while squares refer to the similarity percentage among different documents.

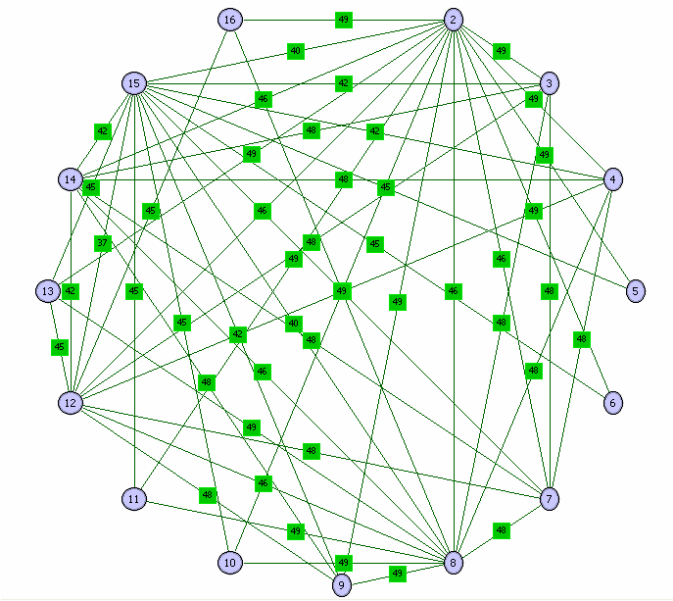


Figure 3.4- Document Network Created using MIR Model

3.3.3 Application of Personalization

Every user has different background and a specific goal when searching for medical information on the web. Search personalization is the process of particularizing the list of search results based on a specific user's interests. In medical domain, personalizing searching results facilitate browsing relevant documents of ordinary users.

In this subsection, we describe the application of a personalization technique that track users' browsing habits. The goal behind this task is to rank the classes of information according to user interest. We chose this technique since it provides a good performance in comparison with other ones in the literature.

Figure 3.5 describes the design of user-profiling task. The design consists of six entities each of which has its own functionality. The following diagram explains the connections among these entities.

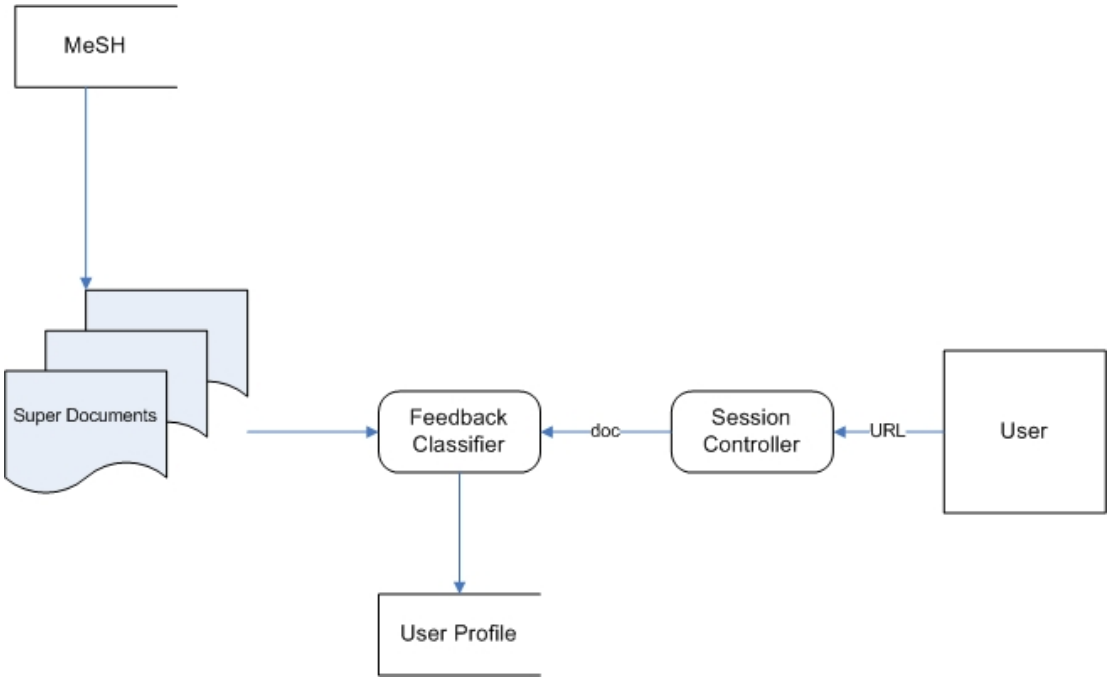


Figure 3.5- Personalization Task

In the following subsections, we explain the design, methodology and implementation issues of the profiling method described in [57]. As we mentioned before, we chose this method since it shows better performance and applicability in comparison with other methods.

3.3.3.1 User Interface

The user interface module is responsible for receiving user's choice (click) to a specific document and, then, passing the chosen URL to the session controller module. Thus, this module is intended to pass URLs without performing any checking operation.

Every user interface is considered as a separate thread. It defines a special web session associated to a specific user identity. We chose to identify users through the IP address of their machines.

The use of IP address as an identification attribute may affect the accuracy of the personalization task since a user might access a specific web page from different machines (different IPs) or many users might use the same machine; the same IP address. For simplicity, we assumed that every visitor uses different IP address. For future expansion of MedicoPort, a more sophisticated user management interface that relies on cookies management might be applied.

3.3.3.2 Session Controller

This module is responsible for controlling session variables. In this case, session controller computes how much time a user has spent reading a document. The document will be considered as an interested one if the time exceeds a specific threshold (the original description of OBP [57] assumes that 5 seconds is the minimum amount of time to decide that the visitor is interested in a given document). Otherwise, the controller ignores the document.

In addition to time consideration, session controller keeps information about user and documents variables such as URLs, time, and access date. After processing session variables, the controller passes document's information to the feedback classifier module.

Moreover, session controller is responsible to make decision about the quality of the page or link contents. During experiments, we noticed that documents that contain only graphics or URLs, such as subject menus, should be ignored. The controller checks the content and decides whether the document is qualified to be logged or not. Notice that, some implementations use to remove documents with size less 1KB, for example.

3.3.3.3 Medical Subject Headings (MeSH) Database

The Medical Subject Headings (MeSH) metathesaurus is a specialized vocabulary of medical concepts. It has been introduced by the National Library of Medicine and used for identifying, categorizing, and searching for medical domain information.

The Cataloging Section at NLM uses MeSH levels for associating a medical category to text documents that are represented in different formats. In this context, we applied the MeSH description of medical categories as special entities to represent categories.

3.3.3.4 Super Documents

A super document is a collection of identification concepts relevant to specific category in medical domain. MeSH (Medical Subject Headings) provides a list of key concepts to distinguish medical category. In order to model the similarity between a given document (from session controller) and these concepts, we used to collect identification concepts into a special form of documents called "*Super Documents*" [57].

A super document might consist of any number of concepts to represent a category of knowledge. In this dissertation, we followed the methodology in [57] in which a super document consists of the top relevant 20 concepts in MeSH headings. So that, every incoming document will be compared with every super document.

3.3.3.5 Feedback Classifier

A log analysis method is applied to process log information and to classify user needs. In this dissertation, our goal is to discover potential correlation between medical categories and user behavior. We applied content-based filtering where the analysis process is tailored to users' interest individually; the system traces every users' attitudes and proposes them classes that are similar to categories the user liked previously [58].

The analysis phase includes applying statistical techniques in addition to data mining methods in order to detect interesting relations. Thus, the module computes

the similarity between chosen document and super documents. The classifier, then, re-weights the categories in the user profile if the similarity exceeds a specific threshold.

For example, suppose user *X* clicks on document *D* and spends more than 5 seconds reading document *D*. The feedback classifier computes the similarity between document *D* and all available super documents. Suppose that the classifier discover that document *D* is relevant to category *Y*. Then, the classifier will update the weight of category *Y* in the user profile.

In detail, the classifier computes the similarity between a document and a super document (as a set of concepts) as follows:

$$Similarity(c_j, d_k) = c_j \circ d_k = \sum_{i=1}^{n} w_{i,j} d_{i,k} \qquad \textbf{(EQUATION 3.9)}$$

where

- n is the number of unique terms in the super document
- w_{ij} is the normalized weight of term *i* in concept *j*
- $d_{i,k}$ is the un-normalized weight of term *i* in document *k*

The un-normalized weight of terms in an incoming document is calculated with respect to the document's mass. This way, terms in highly rated documents receive

higher weights. The un-normalized weight of term *i* in document *j* is calculated as follows:

$$uw_{i,j} = tf_{i,j} \times idf_i \qquad \textbf{(EQUATION 3.10)}$$

where $tf_{i,j}$ = number of occurrences of term *i* in document *j*

$$idf_i = \log \frac{\#\ of\ documents\ in\ the\ collection}{\#\ of\ documents\ in\ the\ collection\ that\ contain\ term\ t_i} \qquad \textbf{(EQUATION 3.11)}$$

To compute the normalized weight of term *i* in document *j*, we used the parameter Mass as follows:

$$d_{i,j} = \frac{uw_{i,j}}{Mass(d_j)} \qquad \textbf{(EQUATION 3.12)}$$

Finally, the classifier stores the sum of similarities relevant to specific category's concepts. Later, categories will be displayed to users according to the descending order or these similarities.

3.3.3.6 User Profile Data Store

User profile data store (log) keep profiling information associated to every user. As stated before, the primary key field (user identity) is the IP address of users' machines. The database keeps all required information to identify user interests. The data log record holds information that enables the server site to keep information about the user and tracks users' actions at the client side. Notice that, this method relies on implicit profiling; collecting visitors information without any users interventions.

The data store includes the following information:

- The visitor's identification IP address
- Session ID: an identifier that uniquely identify user's sessions
- Category of Selected Document
- Time: the amount of time a user spend reading a document
- Access Date

3.3.3.7 Implementation of User-Profiling Task

The purpose of applying this method is to customize the retrieved medical documents categories according to users' interests by analyzing the acquired information from the analysis of users' navigational actions and the usage of retrieved document. Also, the system correlates other information gathered such as documents category.

We have implemented this technique on MedicoPort. At the first phase, we built the data store as a database file with appropriate fields to keep users browsing data. Next step, we prepared the super documents consisting of the MeSH description of medical categories. Also, we implemented the session controller and feedback classifier according to the design description in the previous sections.

3.3.3.8 Profile Management

For long term management of users' profiles, there are mainly two issues that should be considered for the purpose of increasing the accuracy of the method described in the previous sections. The first issue is the profile stability to accurately represent user interests. The second one concerns pruning non-relevant actions that might not reflect user interests.

3.3.3.8.1 Profile Coverage

Profile convergence is defined as [57] the state in which a profile becomes steady and reflects accurate user interests. The goal is to determine how much information required building a stable profile. In this context, a stable profile is defined as the one which requires little or no changes in its information.

In our design, the convergence of a user profile depends heavily on the exhibits of users. But user exhibits are dynamic and can not be predicted in advance. However, since users choices vary in their browsing, the strongest convergence, according to

[57], was detected when profiles were built not based on time but on the number of documents collected. Thus, the systems will consider a profile as a stable one after collecting *N* selected documents. In [57], experiments show that 70 pages (*N*=70) are enough to predict user browsing exhibits. At this number of documents, the profile requires no or little changes.

3.3.3.8.2 Concept Pruning

The next important issue to be considered is the threshold value of the similarity between an incoming document and super documents. In other words, specifying the lower level of similarity in order to assign a category rank in a user profile. In our previous work on MIR model, we computed the similarity at threshold of 5%. At retrieval task, it is better to assign small value to threshold parameter to make sure that most or even all relevant documents were retrieved.

In the context of user profiling, we compute the similarity between chosen documents and super documents in order to recognize whether a user is interested in a specific category or not. This situation leads to a strict threshold policy. Meaning that, the value of the threshold parameter should be large to guarantee a strong similarity that reflects real relation between selected documents and interested categories.

We choose a threshold value of 15%; if the similarity between an incoming document *d* and a given super document *C* is exceeds or equal 15% then the

classifier will modify the similarity of *C* in the user profile by adding the similarity value to the previous one.

$$Similarity(UserID, C) = Similarity(UserID, C) + Similarity(S_c, d_k)$$ **(EQUATION 3.13)**

where:

$Similarity(UserID, C)$: is the degree of interest between a user and category *C*.

$Similarity(S_c, d_k)$: is the similarity between incoming document d_k and a super document S_c that represent category *C*.

3.4 Experiment Setup and Results

We have performed two experiments to evaluate the performance of our model in terms of retrieval recall/precision and topical ranking. In the first experiment, we compared the performance of the model with results reported in TREC-9 (Text Retrieval Conference - filtering track) in addition to KELSI (Knowledge-Enhanced Latent Semantic Indexing) method. In this track, participants submitted different runs to evaluate recall/precision metrics related to 63 queries. Notice that, OHSUMED offers 106 queries that cover the document collection. We tested the results from the implementation of our model using TREC_EVAL program and compared the performance with top TREC results and KELSI.

In our second experiment, we distributed a questionnaire to a group of medical experts and non-experts to evaluate user ranking derived from the MIR model. This experiment relies on a collection of 1,340 medical documents. The collection consists of full-text specialized documents from medical portals and documents from forums and blogs containing medical information, which represent the experience of ordinary people. Furthermore, 603 (45%) documents of this collection cover five medical topics, including insomnia, pregnancy, diabetes, the flu, and HIV.

Finally, we re-implemented MedicoPort [1] based on the MIR model. MedicoPort was implemented based on the .NET framework development environment using Active Server Page (ASP.NET) programming technology. For the purpose of the experiment, MedicoPort was installed on a Pentium machine with 2.0 GHz processor and 1.0 GB RAM. Furthermore, we installed Internet Information Services (IIS 6.0), which is a web server process that offers a set of internet based communication services for servers built on the top of MS Windows.

3.4.1 TREC-based Experiment

In the TREC-9 [54] filtering track, the OHSUMED [55] collection was used as a test dataset. OHSUMED is a combination set of about 348,500 abstracts that referenced form MEDLINE. The collection consists of documents' titles and abstracts from more than 270 medical sources of information over a period of five years (1987-1991). The fields to classify documents include document's title, MeSH indexing

terms, summary section, information source, text author, and publication type. The following figure shows a sample document.

```
.I 4
.U
87049090
.S
Am J Emerg Med 8703; 4(6):504-6
.M
Adolescence; Adult; Aged; Blood Glucose/*ME; Diabetes Mellitus/BL;
Emergencies; Female; Glucose/*AD; Human; Hypoglycemia/*TH; Male;
Middle Age; Prospective Studies; Solutions.
.T
Serum glucose changes after administration of 50% dextrose
solution: pre- and in-hospital calculations.
.P
JOURNAL ARTICLE.
.W
A prospective clinical trial was conducted to estimate the rise in
serum glucose level after an intravenous bolus of 50 ml of 50%
[...]
predicted after a single intravenous bolus of D-50.
.A
Adler PM.
```

Figure 3.6 - Sample OHSUMED document

53 runs were submitted to provide evaluation datasets for different retrieval methods over a set of queries. In this context, we provide the results reported by five runs, which represent the top runs reported by TREC-9 participants. In addition, we provide the evaluation of the results derived from applying the MIR model on the same dataset at threshold value of 5%. Table 3.2 shows a description of these runs and the methods used to retrieve relevant documents.

Table 3.2 - Description of TREC-9 runs and methods

Group	*Run-ID*	*Method*
Carnegie Mellon (CMU-C)	**CMUDIR**	**Incremental Rocchio Algorithm**
University of Toulouse (IRIT)	**Mer9**	**Profiles for non-relevant documents**
Microsoft Research (Cambridge)	**Ok9**	**Limited term selection**
University of Nijmegen	**KUN**	**Score distribution with Rocchio algorithm**
Informatique CDC	**S2RN**	**Neural Network without hidden neurons**

Table 3.3 exhibits the performance of the retrieval task in terms of average precision. The table shows the average precision at every run and the precision at *N* retrieved documents. Notice that we obtained the results in Table 3.3 from running TREC_EVAL program on the datasets listed in the restricted area of TREC-9 server.

Table 3.3 - Retrieval Performance on TREC filtering Track (Top 5-runs)

Run ID	*Average Precision*	*P@5*	*P@10*	*P@15*	*P@500*	*P@1000*
CMUDIR	0.202	1.000	0.800	0.867	0.518	0.501
Mer9	0.213	0.600	0.800	0.600	0.330	0.275
Ok9	0.354	1.000	1.000	0.933	0.594	0.514
KUN	0.364	1.000	1.000	0.933	0.714	0.596
S2RN	0.463	1.000	1.000	1.000	0.656	0.561
MIR	**0.577**	**1.000**	**0.800**	**0.800**	**0.750**	**0.690**

The average precision is computed using the following formula:

$$Average\ Precision = \frac{\sum_{i}^{Nq} Precision(Q_i)}{N_q} \qquad \textbf{(EQUATION 3.14)}$$

where $Precision(Q_i)$ indicates the precision for query Q_i, and N_q is the number of queries.

The results in Table 3.3 show that MIR outperforms the other runs, because it shows a higher average precision. The improvement of MIR over other methods is statistically significant. Thus, the observed differences between MIR and other methods reflect a real difference not due to chance (p-value < 0.0001). Furthermore, the average precision at a large *N* number of documents (i.e., P@500 and P@1000) is higher than other runs. The high average precision of the MIR model results from applying the semantic features of detecting anchor-terms. This feature guarantees the detection of relevant documents even if the query terms infrequently appear in the text, as the degree of attractiveness parameter and the frequency of index terms are independent.

The precision at *N* documents (P@N) indicates the ability of the model to rank relevant documents in the top *N* hits. Our model achieved relatively high precision at small and large *N*. The ranking technique, which relies on ranking documents

according to their topics through the implementation of the document network, plays a significant role in achieving this result.

R-precision is a single value summary of the ranking by computing the precision at the R^{th} position in the ranking, where R is the total number of relevant documents for the current query [19]. Table 3.4 shows the overall average R-precision reported by TREC_EVAL program.

Table 3.4 - R-Precision value: Precision After R Documents Retrieved

Run ID	*R-Precision*
pircT9U2	0.2544
KUNa2T9U	0.2887
KUNb	0.2712
Mer9r1	0.2228
KUNr2	0.3477
S2RNr2	0.4039
MIR	**0.5874**

The results shown in Table 3.4 support our previous conclusion regarding the ranking task. MIR achieved a high R-precision at compared to other runs. The attractiveness and documents mass parameters guarantee that a higher rank is assigned to relevant documents. Figure 3.7 shows the recall/precision curve across all runs. The MIR model achieves higher precision among all standard recall levels

(11-levels). This result implies that MIR performs better than other runs at different recall levels.

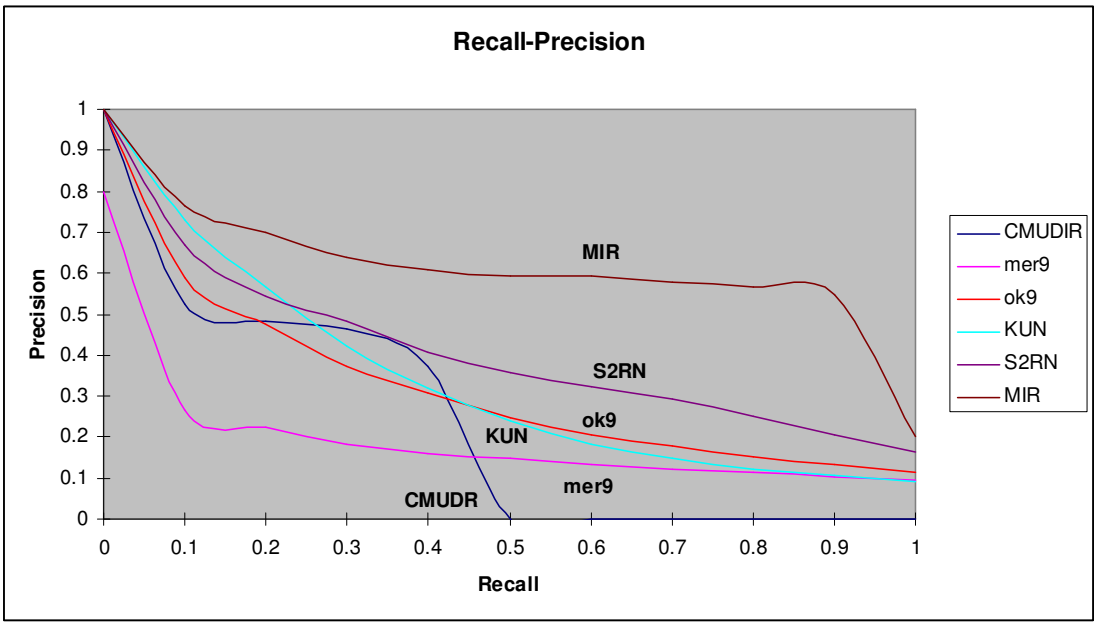


Figure 3.7 - Recall-Precision Curve

Finally, we performed one more experiment to report the interpolated precision (the 11-point average precision) for every query. The goal behind this experiment is to compare the average interpolated precision reported by MIR model with Knowledge-enhanced latent semantic indexing (KELSI) [56]. Both MIR and KELSI rely on UMLS (MeSH lexicon) to expand queries and indexed concepts. The difference is that MIR uses direct indexing and query expansion while KELSI applies singular value decomposition to extract the semantic relation among terms in the documents collection. Notice that, in this experiment we report the interpolated precision at threshold value of zero.

Table 3.5 Interpolated Precision of MIR and KELSI

Query	*Precision (KELSI)*	*Precision (MIR)*	*Query*	*Precision (KELSI)*	*Precision (MIR)*
1	0.4690	0.4810	**33**	0.0781	0.1076
2	0.0056	0.0073	**34**	0.1776	0.1790
3	0.1345	0.3015	**35**	0.0392	0.3117
4	0.0016	0.0109	**36**	0.0333	0.3333
5	0.1532	**0.1415**	**37**	0.0659	0.4000
6	0.1339	0.1807	**38**	0.0012	0.0921
7	0.0025	0.0113	**39**	0.5833	**0.4904**
8	0.0000	0.0000	**40**	0.0014	0.4471
9	0.0193	0.0309	**41**	0.0634	0.4228
10	0.0230	0.0273	**42**	0.0479	0.2032
11	0.0003	0.0094	**43**	0.0603	0.5906
12	0.0009	0.0012	**44**	0.0007	0.0914
13	0.2222	**0.2178**	**45**	0.0050	0.0812
14	0.0024	0.0203	**46**	0.2724	0.6955
15	0.3760	**0.3701**	**47**	0.0367	0.2949
16	0.1234	0.4075	**48**	0.0008	0.1705
17	0.0445	0.3901	**49**	0.0000	0.0000
18	0.0573	0.2160	**50**	0.0161	0.5102
19	0.0005	0.0027	**51**	0.0014	0.0048
20	0.3333	**0.2500**	**52**	0.0062	0.0771
21	0.0142	0.1009	**53**	0.0248	0.4208
22	0.4362	0.6703	**54**	0.3024	0.6955
23	0.0004	0.0141	**55**	0.1382	0.2700
24	0.0052	0.0096	**56**	0.0051	0.0104
25	0.0505	0.0919	**57**	0.0031	0.1000
26	0.0581	0.2245	**58**	0.2654	0.7106
27	0.0003	0.4102	**59**	0.0004	0.0058
28	0.0000	0.0000	**60**	0.0057	0.0062
29	0.0169	0.3690	**61**	0.3287	0.3352
30	0.0020	0.2122	**62**	0.0921	0.4501
31	0.0121	0.0778	**63**	0.1546	0.2700
32	0.0013	0.1981			

In Table 3.5, results illustrated by bold format indicate improvement of KELSI over MIR method. The findings, in this table, show that MIR outperforms KELSI in 58-queries while KHLSI reports higher interpolated precision in 5-queries. Also, we noticed that, for many queries, the reported interpolated precisions were close.

To make the comparison statistically significant, we applied paired t-test over these results to determine whether the difference between the means of MIR and KELSI can be considered real; not by chance. The following table shows the output of paired t-test analysis

Table 3.6 Two-tailed Paired t-test on MIR and KELSI

	MIR	*KELSI*
Mean	*0.226*	*0.087*
Standard Deviation	*0.206*	*0.134*
P-value	*0.0001*	
Mean(MIR) - Mean(KELSI)	*0.138*	
95% confidence interval of this difference	*[0.0971 , 0.180]*	
Standard error of difference	*0.021*	

The results in Table 3.6 indicate that the enhancement achieved by MIR model over KELSI is statistically significant.

Latent semantic analysis applied to capture patterns of terms occurs in a collection of documents. In other words, to capture relevant terms occur in different documents by inferring hyponyms from the occurrences of concepts within and among documents. In order to reduce the error resulted from associating irrelevant terms, KELSI augments term-by-document matrix with MeSH related concepts. Our method in MIR use a direct expansion of concepts using MeSH, without analyzing the relation among terms since this analysis might miss a significant relation or might relate irrelevant terms.

3.4.2 Questionnaire-based Experiment

In this section, we explain the performance results of the questionnaire-based experiments. After collecting data from target users, we analyze and report the performance in terms of topical ranking, precision, and recall.

We have distributed a questionnaire among two groups of users: 25 Experts and 31 non-experts in medical domain. Table 3.7 explores the classes of the quetionnaire's participats. The questions focus on evaluating three properties of the system: Ranking, Precision, and the relationship among successive documents appeared in the hit lists. In the following subsections, we provide detailed discussion about the evaluating procedure.

Table 3.7 - Questionnaire Participant Profile

Expert		***Non Expert***	
Nurses	11	Graduate Students	13
Medical Doctors	10	Undergraduate Students	12
Other	4	Other	6
Total	25	Total	31
Total Number of Participants		**56**	

The sample collection of documents covers 5 topics in medical domain. These topics include: Insomnia, Diabetes, Flu, Pregnancy, and HIV. We asked the users to inquire about these topics using their own queries. For example, some users could

inquire the term "Insomnia", "Sleeping disorders", or "Sleeping Problems" for the first query.

3.4.2.1 Topical Ranking

To evaluate the ranking task, we have distributed a questionnaire over two groups of participants, namely, 25 experts and 31 non-experts in the medical and health domain. The questions focus on evaluating the rank; the relationship among successive documents appear in the hit lists. We asked the users to provide us with the rank of their best choices, that is, the documents that satisfy their requests among the retrieved hit-list. The retrieved hit-list was designed to display 20 documents per page; each of them is assigned a unique number.

Table 3.8 - User Judgments (Medical and non-Medical)

Query	*Best Choice (Experts)*	*Best Choice (non-Experts)*	*Average Best Choice (Experts)*	*Average Best Choice (non-Experts)*
Insomnia	1 → 5	1 → 3	2.4	1.5
Diabetes	1 → 9	1 → 3	4.4	1.9
Flu	1 → 8	1 → 8	4.1	3.5
Pregnancy	1 → 6	1 → 6	3.9	2.4
HIV	1 → 10	1 → 10	6.5	2.2

The second column, Table 3.8, represents the best choice of expert users related to every query. These results indicate that the first choice of all expert users ranges from 1 to 10, meaning that the first 10 results satisfy expert user requests.

Furthermore, we asked the users to describe the relationship among successive displayed documents; 92% of expert users assess this relation as "Strong."

Similarly, the third column shows the best choices that satisfied non-expert users. According to user feedback, the best choice for these users ranges from 1 to 10. Moreover, 93.8% of users report that the relationship among successive documents is "Strong."

According to these results, we can conclude that the MIR model is able to rank documents according to their topics, as most users described the relation among successive documents as “Strong.” The robustness of the ranking process comes from using documents anchors and medical domain vectors to model retrieval relations. In addition, the expansion of medical concepts using medical-domain semantic relations enhances the quality of the ranking process.

Although the resulted hit-list satisfied the questionnaire’s participants, we noticed that the system produces different rank for some similar queries. Here, we provide an example of this anomaly and comment on the retrieved hit-lists. In this example, we used to retrieve the hit-list related to three relevant queries on a small dataset that covers information about “Pregnancy”:

1. Pregnancy
2. Getting Pregnant
3. Enjoying Pregnancy

These queries are relevant since all of them refer to the same topic. Figure 3.8 and 3.9 show the retrieved hit lists for queries “Pregnancy” and “Getting Pregnant”. The second query consists of two terms; one of them is medical and synonym to the term in the first query. Query “Getting Pregnant” receives lower Mass than query “Pregnancy” since it contains one domain term out of two.

Docurl	Doctitle
http://www.nlm.nih.gow/medlineplus/pregnancy.html#	Pregnancy
http://www.nlm.nih.gow/medlineplus/pregnancyandsubstanceabuse.html#	Pregnancy and Substance Abuse
http://www.nlm.nih.gow/medlineplus/teenagepregnancy.html#	Teenage Pregnancy
http://www.nlm.nih.gow/medlineplus/aidsandpregnancy.html#	AIDS and Pregnancy
http://www.nlm.nih.gow/medlineplus/diabetesandpregnancy.html#	Diabetes and Pregnancy
http://www.nlm.nih.gow/medlineplus/highriskpregnancy.html#	High Risk Pregnancy
http://www.nlm.nih.gow/medlineplus/infectionsandpregnancy.html#	Infections and Pregnancy
http://www.nlm.nih.gow/medlineplus/highbloodpressureinpregnancy.html	High Blood Pressure in Pregnancy
http://www.nlm.nih.gow/medlineplus/pregnancyloss.html#	Pregnancy Loss
http://www.nlm.nih.gow/medlineplus/ency/article/000895.htm#	Ectopic pregnancy
http://www.nlm.nih.gow/medlineplus/ency/article/001516.htm#	Adolescent pregnancy
http://www.nlm.nih.gow/medlineplus/ency/article/003778.htm#	Pregnancy ultrasound
http://www.skinsight.com/adult/striaeofPregnancy.htm#	Stretch Marks of Pregnancy (Striae of Pregnancy)
http://www.acog.org/publications/patient_education/bp090.cfm#	Early Pregnancy Loss: Miscarriage and Molar Pregnancy
http://www.nlm.nih.gow/medlineplus/ency/article/003264.htm#	Vaginal bleeding in pregnancy
http://www.mayoclinic.com/print/how-to-get-pregnant/PR00103	How to Get Pregnant
http://www.bt.cdc.gow/agent/smallpox/vaccination/preg-factsheet.asp#	Smallpox Vaccination Information for Women Who Are Pregnant or Breast
http://www.nlm.nih.gow/medlineplus/infertility.html	Infertility
http://www.mayoclinic.com/health/fertility/MC00023	Healthy Sperm: Improving Your Fertility
http://www.mayoclinic.com/health/symptoms-of-pregnancy/PR00102	Symptoms of Pregnancy

Figure 3.8 The Hit-list of The Query "Pregnancy"

Docurl	Doctitle
http://www.mayoclinic.com/print/how-to-get-pregnant/PR00103	How to get pregnant
http://www.nlm.nih.gov/medlineplus/pregnancy.html#	Pregnancy
http://www.mayoclinic.com/health/symptoms-of-pregnancy/PR00102	Symptoms of Pregnancy
http://www.nlm.nih.gov/medlineplus/ency/article/001516.htm#	Adolescent pregnancy
http://www.nlm.nih.gov/medlineplus/pregnancyandsubstanceabuse.html#	Pregnancy and Substance Abuse
http://www.nlm.nih.gov/medlineplus/teenagepregnancy.html#	Teenage Pregnancy
http://www.nlm.nih.gov/medlineplus/aidsandpregnancy.html#	AIDS and Pregnancy
http://www.nlm.nih.gov/medlineplus/pregnancyloss.html#	Pregnancy Loss
http://www.nlm.nih.gov/medlineplus/diabetesandpregnancy.html#	Diabetes and Pregnancy
http://www.nlm.nih.gov/medlineplus/highriskpregnancy.html#	High Risk Pregnancy
http://www.nlm.nih.gov/medlineplus/infectionsandpregnancy.html#	Infections and Pregnancy
http://www.nlm.nih.gov/medlineplus/ency/article/000895.htm#	Ectopic pregnancy
http://www.nlm.nih.gov/medlineplus/ency/article/003778.htm#	Pregnancy ultrasound
http://www.nlm.nih.gov/medlineplus/highbloodpressureinpregnancy.html	High Blood Pressure in Pregnancy
http://www.bt.cdc.gov/agent/smallpox/vaccination/preg-factsheet.asp#	Smallpox Vaccination Information for Women Who Are Pregnant or Breastfe
http://www.skinsight.com/adult/striaeofPregnancy.htm#	Stretch Marks of Pregnancy (Striae of Pregnancy)
http://www.acog.org/publications/patient_education/bp090.cfm#	Early Pregnancy Loss: Miscarriage and Molar Pregnancy
http://www.nlm.nih.gov/medlineplus/ency/article/003264.htm#	Vaginal bleeding in pregnancy
http://www.nlm.nih.gov/medlineplus/infertility.html	Infertility
http://www.mayoclinic.com/health/fertility/MC00023	Healthy Sperm: Improving Your Fertility

Figure 3.9 -The Hit-list of The Query "Getting Pregnant"

The reason behind having two different rankings for the first two queries is the parameters ***G*** and distance in the MIR model. In this example; “Getting Pregnant”, the parameter ***G*** gave the documents in which the title contains the term “Pregnant” higher rank than in the other query “Pregnancy”. Further, the distance is close to documents that have frequent “Pregnant” term since our system gives higher weight to exact terms as compared with similar ones.

The third query “Enjoying Pregnancy” contains the exact term of the first query “Pregnancy”. In this case, both queries receive the same value of parameter ***G*** (for this sample of documents) and distance. Figure 3.10 shows the retrieved hit-list of the query “Enjoying Pregnancy”

Docurl	Doctitle
http://www.nlm.nih.gov/medlineplus/pregnancy.html#	Pregnancy
http://www.nlm.nih.gov/medlineplus/pregnancyandsubstanceabuse.html	Pregnancy and Substance Abuse
http://www.nlm.nih.gov/medlineplus/teenagepregnancy.html#	Teenage Pregnancy
http://www.nlm.nih.gov/medlineplus/aidsandpregnancy.html#	AIDS and Pregnancy
http://www.nlm.nih.gov/medlineplus/diabetesandpregnancy.html#	Diabetes and Pregnancy
http://www.nlm.nih.gov/medlineplus/highriskpregnancy.html#	High Risk Pregnancy
http://www.nlm.nih.gov/medlineplus/infectionsandpregnancy.html#	Infections and Pregnancy
http://www.nlm.nih.gov/medlineplus/highbloodpressureinpregnancy.html	High Blood Pressure in Pregnancy
http://www.nlm.nih.gov/medlineplus/pregnancyloss.html#	Pregnancy Loss
http://www.nlm.nih.gov/medlineplus/ency/article/000895.htm#	Ectopic pregnancy
http://www.nlm.nih.gov/medlineplus/ency/article/001516.htm#	Adolescent pregnancy
http://www.nlm.nih.gov/medlineplus/ency/article/003778.htm#	Pregnancy ultrasound
http://www.skinsight.com/adult/striaeofPregnancy.htm#	Stretch Marks of Pregnancy (Striae of Pregnancy)
http://www.acog.org/publications/patient_education/bp090.cfm#	Early Pregnancy Loss: Miscarriage and Molar Pregnancy
http://www.nlm.nih.gov/medlineplus/ency/article/003264.htm#	Vaginal bleeding in pregnancy
http://www.mayoclinic.com/print/how-to-get-pregnant/PR00103	How to Get Pregnant
http://www.bt.cdc.gov/agent/smallpox/vaccination/preg-factsheet.asp#	Smallpox Vaccination Information for Women Who Are Pregnant or
http://www.nlm.nih.gov/medlineplus/infertility.html	Infertility
http://www.mayoclinic.com/health/fertility/MC00023	Healthy Sperm: Improving Your Fertility
http://www.mayoclinic.com/health/symptoms-of-pregnancy/PR00102	Symptoms of Pregnancy

Figure 3.10 -The Hit-list of The Query "Enjoying Pregnancy"

The system rank the documents exactly like the first query “Pregnancy” but gives similarity values lower than the one in the first query. The reason is that query “Pregnancy” receive higher Mass than query “Enjoying Pregnancy” and all other parameters remain fixed.

3.4.2.2 Precision

Precision is defined as the percentage of the number of relevant documents retrieved by a search engine to the total number of retrieved documents. In this context, we measure the precision on 3 intervals; the first 10 retrieved documents, the first 20 retrieved documents, and all retrieved documents. High precision among the first 10 or 20 first retrieved documents indicates high performance retrieval process.

Table 3.9 - Precision (Expert Users)

Query	*Precision (First 10)*	*Precision (First 20)*	*Precision (All) Threshold = 0.1*
Insomnia	98.8%	96%	89.5%
Diabetes	96.4%	92.4%	91.9%
Flu	100%	96.8%	93.8%
Pregnancy	96.9%	95%	89.8%
HIV	95.3%	94.2%	92.7%
Average Precision	**97.48**%	**94.88**%	**91.54**%

For each query, we used to calculate the precision from the feedback of every user and then we calculate the average precision as:

$$Precision\ (Qj) = \frac{\sum_{i=1}^{N} Precision(i)}{N} \qquad \textbf{(EQUATION 3.15)}$$

where *N* is the number of participants in the group of expert users and *Precision* (i) is the user precision for query *j*. The first column in Table 3.9 represents the precision among the first 10 retrieved documents; this value is important as it reflects the robustness of the retrieval model. The values in the second column represent the precision among the first 20 retrieved documents. We asked five expert users to check all retrieved documents in order to provide us with the number of unrelated documents in the retrieved hit list. We found that at similarity greater than or equal a threshold of 0.1, the average precision is 91.54%. Finally, we calculated

the precision from the feedback of non-experts using the proposed formula for the first 10 hits and the first 20 hits, (see Table 3.10).

Table 3.10 - Precision (Non-Expert Users)

Query	*Precision (First 10)*	*Precision (First 20)*
Insomnia	100%	94%
Diabetes	98.6%	93.8%
Flu	100%	94.1%
Pregnancy	96.9%	96.3%
HIV	98.1%	97.7%
Average Precision	**98.72**%	**95.18**%

The precision metric tests the ability of the retrieval model to retrieve relevant results. It is common in information retrieval systems that high experimental threshold resulted in high precision since high threshold value eliminates weakly related documents. In addition to high threshold, the enrichment of concepts with medical domain ontologies increases the ability of the model to predict the semantic relations among anchors. Consequently, it enhances the relevancy among items in the retrieved list.

The relationship among the top 10 choices of experts and non expert users is depicted in Figure 3.11. The precision reported by non-expert users is always greater than or equal to the one reported by expert users for all queries. The variation among their choices seems to be normal as experts strictly selected relevant documents. For example, most of experts considered the document ‘Pregnancy and Cancer” as

irrelevant to the query 'Pregnancy' while most of non-expert users considered this document as relevant to the same query.

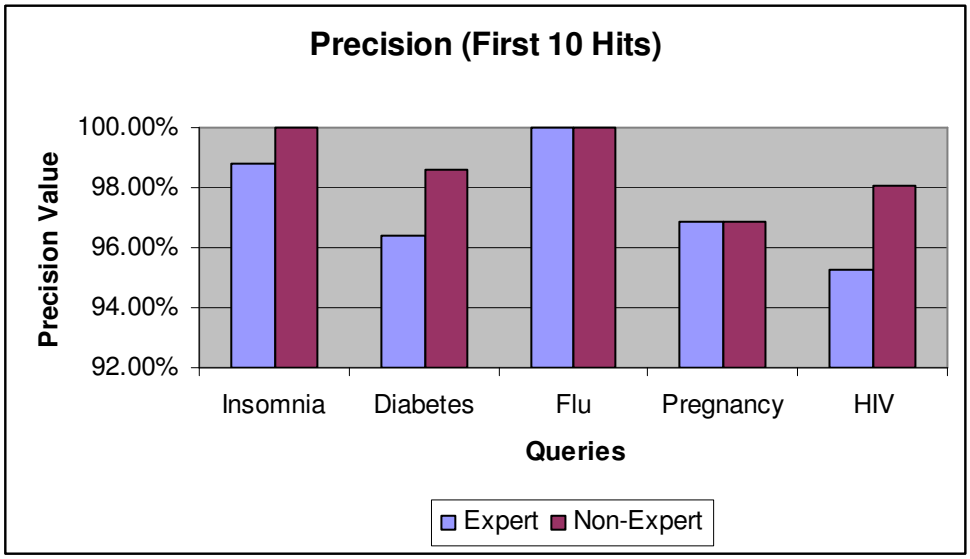


Figure 3.11 - Precision Comparison (First 10 Hits)

Equally important, in Figure 3.12 the relationship among the top 20 choices shows more fluctuations according to the reported precision. In the first 20 hits, many documents were not directly similar to the query terms but still relevant. For instance, experts consider the document 'Urine Test' as a relevant document to the query 'Diabetes' while all non-experts consider this document as irrelevant to the same query. On the other hand, some users, from both groups, consider 'Influenza' and 'TamiFlu' as irrelevant to 'Flu' query while many users consider them relevant.

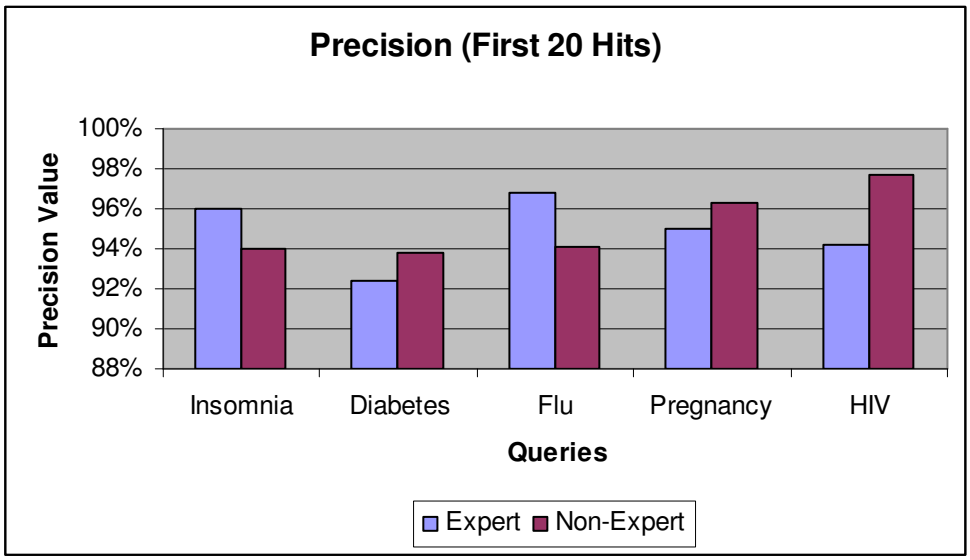


Figure 3.12 - Precision Comparison (First 20 Hits)

3.4.2.3 Recall

Recall is defined as the percentage of the number of retrieved relevant documents to the total number of related documents in the collection. Since it is quite hard to find all relevant documents in the entire collection, we define a set of benchmarks; a set of documents that have manually developed. The purpose is to check whether the system is able to retrieve these documents or not.

For every query, we plant 10 documents relevant to that query. These documents have been carefully selected to include professional and non professional information. In other words, part of this collection is developed using a variation of synonyms, contextually and partially related topics. For example, for the query insomnia, we developed documents that describe professional information about

insomnia, sleeping disorder, and sleeping problems. In addition, we put documents without titles representing documents from forums and blogs; these documents are important as they help in distributing people's experience. Non professional documents contain fewer domain terms and many typical human concepts to describe medical topics.

Title: Pregnancy ultrasound
Pregnancy sonogram; Obstetric ultra-sonography; Obstetric sonogram; Ultrasound - pregnancy
Definition
A pregnancy ultrasound is an imaging test that uses sound waves to see how a fetus is developing in the womb. It is also used to check the female pelvic organs during pregnancy.
How the Test is performed
You will lie down for the procedure. The person performing the test places a clear, water-based gel on your belly and pelvis area and then moves a hand-held probe over the area. The gel helps the probe transmit sound waves. These waves bounce off the body structures, including the developing fetus, to create a picture on the ultrasound machine.
In some cases, a pregnancy ultrasound may be done by placing the probe into the vagina. For information on this procedure, see transvaginal ultrasound.

This is a sample of a sowed document that represents professional information about pregnancy from a professional source. The document has a title and contains expert terminology to describe the topic. On the other hand, we sowed non professional documents that represent forum or blogs information. This type of documents is important as well since it may contains the experience of people; in the following example, the experience of a pregnant seems to be important as she provides

practical solutions to some health problems. The example shows a text that contains a set of advices about nutrition habits in pregnancy.

Hi guys

This is my advice to you:
The first trimester is crucial for your baby's development and really is the most important nutritionally. During the first trimester your baby will be developing all its major organs as well as finger nails, eyebrows and of course a little beating heart. All of that is made from what you have eaten! Important nutrients during this time are Folic acid, B vitamins as well as Essential Omega Oils and Zinc. Your body will also need plenty of Iron as blood volume increases and the placenta is formed.
…….
…….
……..

After inserting these artificial documents in the collection, we ran the system and checked the number of retrieved testing documents. This way, we can predict the recall metric with respect to the experimental threshold. The success of this experiment depends on the type of sowed documents. We believe that this technique is able to give us a clear picture about the performance recall of MIR model. The following table shows the recall metric based on benchmarking experiment at two different experimental thresholds:

Table 3.11 - Recall Values at Threshold of 0 and 0.1

Query	*Recall at threshold=0*	*Recall at Threshold=0.1*
Insomnia	100%	100%
Diabetes	100%	80%
Flu	100%	90%
Pregnancy	100%	80%
HIV	100%	90%
Average	**100**%	**88**%

As shown in Table 3.11, the average recall of all queries at threshold value of zero is 100%. This result indicates that the model is able to retrieve every relevant document even if it is not crawled from specialized sources. Forum and blogs documents receive lower rank than professional documents. Besides, the average recall is 88% at threshold value of 0.1. The average recall at high threshold goes down because documents that receive low similarity have been omitted from the resulted set.

3.4.3 Comparison with other related models

In this section, we provide a comparison between MIR model and two other models: vector and cosine similarity based on the co-occurrence of MeSH terms models (MeSH co-occurrence model); the latter has been implemented to compute paper-paper similarity in [10]. For this purpose, we used to compare the models understudy using reference-count analysis, which have been introduced in [93].

Reference-count is an information retrieval measure to compare the effectiveness of information retrieval systems based on documents' ranking. Given a specific number of top documents retrieved by an information retrieval system, the evaluation algorithm sums up the ranking differences of these documents in the results of all other retrieval systems understudy as compared to an ideal ranking; in this case, the questionnaire's participants ranking. Thus, each retrieval system assigned a reference-count score using the following formula:

$$S_i(N) = \sum_{j=1}^{N} w\ o(d_{i,j}) \qquad \textbf{(EQUATION 3.16)}$$

where S_i represents the reference-count score of system i with respect to the top N documents returned by S_i. The sub-formula $w\ o(d_{i,j})$ is computed as:

$$w(d_{i,j}) = K - n \qquad \textbf{(EQUATION 3.17)}$$

where K is a constant value that is set to (N+1). For more details about this method and the proof of the above equations, see [93].

First, we implemented vector and MeSH co-occurrence models to retrieve documents from the same collection that have been used in the implementation of MIR model. Also, we used to provide the same conditions for the implementation of these models including query expansion and stemming. Then, we ran the system to retrieve the hit-lists related to the same set of queries that have been included in the study of the MIR model. The following figures show the hit-lists of the query 'Diabetes' that are related to the three models:

	doctitle
	American Association Diabetes Educators, A Core Curriculum For Diabetes E
	Information Regarding Diabetes - Diabetes Articles - ArticleDoctor.com
	What Causes Diabetes? - Diabetes Articles - ArticleDoctor.com
	Diabetes Education - Diabetes Articles - ArticleDoctor.com
	Exercises To Control Diabetes - Diabetes Articles - ArticleDoctor.com
	Diabetes Symptom
	Diabetes Types Are Type2, Type 1 And Gestational Diabetes. - Diabetes Articl
	Beginning Signs Of Diabetes - Diabetes Articles - ArticleDoctor.com
	Diabetes In Children - Diabetes Articles - ArticleDoctor.com
	Diabetes Related Problems - Diabetes Articles - ArticleDoctor.com
	Canine Diabetes Is A Common Diagnosis In Dogs. - Diabetes Articles - Article
	American Association Diabetes Educators, A Core Curriculum For Diabetes
	Diabetes Medication
	Causes Of Diabetes Mellitus
	Controlling Type II Diabetes Is Achieving And Maintaining An Ideal Body
	Diabetes Care
	Diabetes Complications
	Complications Symptoms Of Diabetes
	Ayurvedic Treatments To Diabetes
	MedlinePlus: Diabetes Type 1
	Cinnamon Diabetes Treatment

Figure 3.13 - The First 21 hits of 'Diabetes' Query -MIR Model

	doctitle
	American Association Diabetes Educators, A Core Curriculum For Diabetes Educa
	Beginning Signs Of Diabetes - Diabetes Articles - ArticleDoctor.com
	Diabetes Symptom
	Diabetes Related Problems - Diabetes Articles - ArticleDoctor.com
	Canine Diabetes Is A Common Diagnosis In Dogs. - Diabetes Articles - ArticleDoct
	American Association Diabetes Educators, A Core Curriculum For Diabetes
	Exercises To Control Diabetes - Diabetes Articles - ArticleDoctor.com
	Diabetes Types Are Type2, Type 1 And Gestational Diabetes. - Diabetes Articles -
	What Causes Diabetes? - Diabetes Articles - ArticleDoctor.com
	Diabetes Education - Diabetes Articles - ArticleDoctor.com
	Diabetes Medication
	Information Regarding Diabetes - Diabetes Articles - ArticleDoctor.com
	Causes Of Diabetes Mellitus
	Controlling Type II Diabetes Is Achieving And Maintaining An Ideal Body
	Diabetes Care
	Diabetes Complications
	Complications Symptoms Of Diabetes
	Ayurvedic Treatments To Diabetes
	MedlinePlus: Diabetes Type 1
	Cinnamon Diabetes Treatment
	Diabetes In Children - Diabetes Articles - ArticleDoctor.com

Figure 3.14 - The First 21 Hits of 'Diabetes' Query -Vector Model

	doctitle
	American Association Diabetes Educators, A Core Curriculum For Diabetes Educatior
	Diabetes Symptom
	Beginning Signs Of Diabetes - Diabetes Articles - ArticleDoctor.com
	Diabetes Types Are Type2, Type 1 And Gestational Diabetes. - Diabetes Articles - Art
	Diabetes Education - Diabetes Articles - ArticleDoctor.com
	Exercises To Control Diabetes - Diabetes Articles - ArticleDoctor.com
	Diabetes Medication
	Information Regarding Diabetes - Diabetes Articles - ArticleDoctor.com
	What Causes Diabetes? - Diabetes Articles - ArticleDoctor.com
	Diabetes Related Problems - Diabetes Articles - ArticleDoctor.com
	Canine Diabetes Is A Common Diagnosis In Dogs. - Diabetes Articles - ArticleDoctor.c
	American Association Diabetes Educators, A Core Curriculum For Diabetes
	Causes Of Diabetes Mellitus
	Diabetes In Children - Diabetes Articles - ArticleDoctor.com

Figure 3.15 - The First 14 Hits of 'Diabetes' Query –MeSH Co-Occurrence

Figures 3.13, 3.14 and 3.15 show that the rank of retrieved documents distinct among the models under discussion; except in some cases. For example, the document that received a rank 2 using MIR model received a rank 12 in the vector model and 8 in MeSH co-occurrence model. Table 3.12 summarizes this information for all queries. Notice that, the documents that have been selected in this test are the ones which have been selected by the questionnaire's participants. For example, in the questionnaire, users have selected {1, 2, 3, 4, 5, 6, and 9} as the best choice for the query 'Diabetes'. For every model, we put the rank of the corresponding documents that have been retrieved by vector and MeSH co-occurrence models. For example, the hit list {1,2,3,4,5} from proposed model correspond to the set {1,2,7,6,3} from MeSH co-occurrence model.

Table 3.12 - The list of users' choices

Query	*MIR Model*	*Vector Model*	*MeSH co-occurrence Model*
Insomnia	{1,2 ,3, 4, 5}	{1, 6, 8, 2, 11}	{1, 2, 7, 6, 3}
Diabetes	{1, 2, 3, 4, 5, 6, 9}	{1, 12, 9, 10,7,3,21}	{1, 8, 9, 5, 6, 2,14}
Flu	{1,2 ,3 ,4 , 5, 6, 7, 8}	{4,6,1,2,7,11,9,16}	{2,5,1,8,9,10,6,15}
Pregnancy	{1,2,3,4,5,6}	{1,4,5,6,9,19}	{1,2,6,7,9,12}
HIV	{1,2,3,4,5,6,7,8,9,10}	{5,6,1,7,8,10,11,14,13,18}	{1,2,4,6,5,9,10,11,13,14}

Table 3.12 shows the ranking differences among the models under study. In few cases, some documents receive the same rank in all models such as document 1 in queries 'Insomnia', 'Diabetes', and 'Pregnancy'. While in many cases, the rank seems to be different among the models. Thus, these data is sufficient to tell us that

every model provide different ranking for a collection of documents that is relevant to a specific query.

Next, we used to measure the difference among these ranks using reference-count measure for the top 10 documents retrieved from every model. After applying the method, we find the following reference counts for each model.

Table 3.13 - Reference-Count corresponding to the top 10-hits for each query

Query	*RC(MIR)*	*RC(Vector)*	*RC(MeSH co-occurrence model)*
Insomnia	40	27	36
Diabetes	47	14	32
Flu	52	32	32
Pregnancy	45	22	29
HIV	55	17	35

Notice that, the higher the value of reference-count is, the better the ranking is in the first *N*-hits. From Table 3.13, we can conclude that MIR model provides better ranking at the top 10 hits while MeSH co-occurrence model receives the second position and vector model obtains the third one.

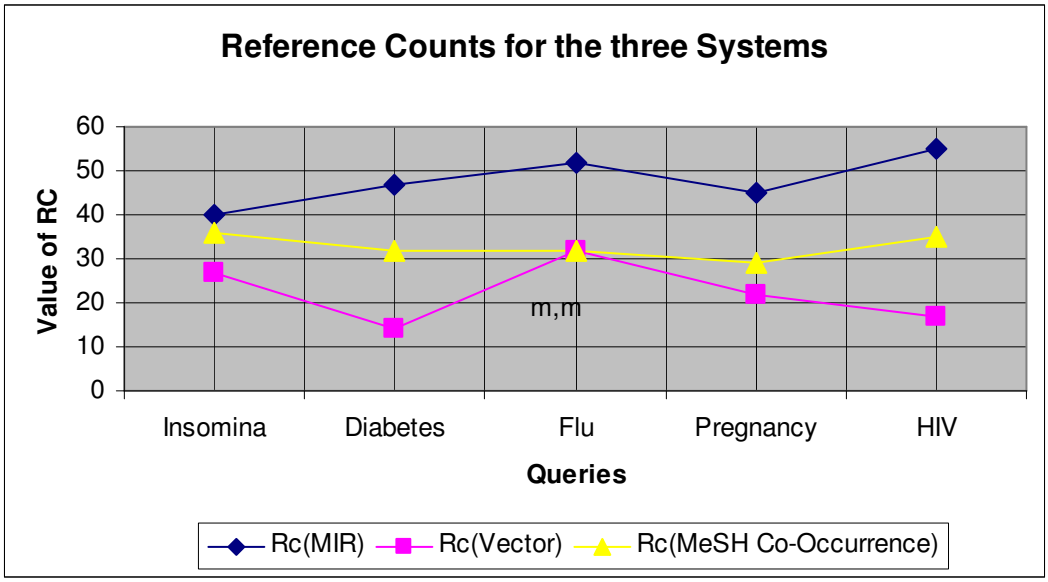


Figure 3.16 - Reference-Counts (Rc) for the Models Understudy

Figure 3.16 shows the correlation among the three models with respect to five queries. The line with diamond markers represents the reference-count of MIR model, the line with triangular markers depicts the reference-count of MeSH co-occurrence model, and the line with square markers represents the reference-count of traditional vector model.

3.5 Related Work

Modelling the similarity among text entities is a potential area that can affect the overall effectiveness, in terms of performance, of information retrieval systems. Rather than focusing on algorithm performance, recent research concentrates on representing information using different features to improve the retrieval process [59, 60].

In the literature, the common approach is to represent documents as a collection of all individual terms, often referred to as bag-of-words representation. Many studies [61, 62] show that using sophisticated feature representation does not improve the effectiveness of retrieval processes for general-purpose retrieval systems but rather provides for the significant enhancement of domain-specific text.

In traditional models such as the VSM, terms are assumed to be the basic entity in statistical methods for feature analysis and discrimination. To accomplish high-precision document retrieval in a domain-specific environment, the development of more informative features has become a significant area of research in the information retrieval literature. Methods such as: Bi-grams, trigrams, and n-grams have been widely used in NLP research areas [63, 64] to construct advanced features.

In the medical domain, experiments in [65, 66] showed that using medical terms and medical phrases resulted in better information retrieval performance in comparison with the classical bag-of-words approach. In this study, we apply multiple features to analyze the similarity among medical and healthcare documents.

Although medical websites and portals such as SNOMED [67], OMNI [68], and MEDHUNT [69] offer a useful search engine for medical information, these tools do not provide consistent responses for medical topics. For instance, OMNI distinguishes between the queries “Breast cancer” and “Carcinoma of Breast,” while these queries are synonymous in a medical context. In contrast, our proposed model

relies on semantic-enrichment to overcome this problem. Semantic enrichment enhances the retrieval and ranking tasks [70-71] to extract relations that reduce searching biases.

Similarly, the models used in PubMed [2], WEBMD [3], MEDLINEPLUS [72], RELEMED [73], ESSIE [74], and WRAPIN [75] rely on structured collection of documents and fixed data sources. Other semantic search engines such as GOWEB [76] have been designed for searching heterogeneous clinical databases based on question answering and ontology enrichment techniques. The semantic searching technique in GOWEB is targeted to experts in medical domain, while our proposed model targeted to non-professional users. We applied query expansion using UMLS (Unified Modelling Language System) [48] to handle this requirement.

Document networks have been implemented to increase the effectiveness of searching tools. Such networks improve the quality of generated hit-lists, as closely associated documents seem to be relevant to the same users' requests [77]. A study of PubMed query logs shows that users choose article titles with noticeable frequency, and in fact, they frequently choose the same titles [78]. The information gathered indicates that around 20% of all specialist sessions comprise one click or more on a relevant document. Another observation from this study demonstrates that the most repeated behaviour following selecting a relevant article is another click on different related article. This behaviour has been noticed on about 40% of the observations. The study suggests that specifications of document networks provide an evidence for ranking document according to their topics and increase the

effectiveness of the searching process. This study motivates us to investigate the significance of document networks in ranking medical documents.

In TREC-9 [54], participating groups have suggested many approaches to process the official collection of medical documents; OHSUMED collections [55]. These approaches can be classified into four categories, namely, query expansion, threshold adaptation, document profiling, and local context feature selection. Query expansion is the process of adding relevant concepts to user queries. For example, the system expands the query "Breast Cancer" with relevant phrases such as "Carcinoma of Breast." Threshold adaptation is the process of omitting retrieved documents with similarity that is less than a specific threshold (or percentage). The document profiling approach keeps information about a document's structure, such as title, abstract, and keywords sections. Finally, context feature selection defines a vector of features relevant to each topic. In other words, a set of terms is defined as a representative vector for every topic in the collection.

Okapi and KUN [79] rely on query expansion as well as complex threshold adaptation to retrieve relevant documents. Unlike the Okapi statistical method of query expansion, our proposed model expands query terms based on UMLS [48], which is a specialized source of medical knowledge that generates related concepts based on MeSH sub-headings.

CMUDIR [80] relies on extracting relevant documents by separating documents according to their domain using a KNN-like algorithm. A similar method is used in

IRIT research to separate irrelevant documents using document-profile. Similar to document profiling, our model extracts information such as titles and keywords as one feature to model similarity among documents. Unlike CMUDIR and IRIT, our model is constructed to address different types of documents, including structured and unstructured documents. In addition, the features used to model the similarity function in this study are directed at maximizing the retrieval precision as well as the ranking of relevant documents.

ICDC [81] implements a local context feature selection algorithm to improve precision. This approach achieves the highest precision and the highest ranking scores in TREC-9. Our approach is similar to ICDC in terms of using representative terms. Our model represents documents using medical-only terms as a representative vector in addition to other features. Unlike the ICDC approach, our technique does not rely on representing information for every topic, because we assume that topics are not known in advance.

Finally, KELSI (Knowledge-Enhanced Latent Semantic Indexing) [56] implements latent semantic analysis to enhance query matching performance. KELSI used to construct term-document vectors to apply singular value decomposition. In addition, the method attaches relevant MeSH concepts as vector to the augmented matrix for the purpose of analyzing the semantic relations among terms in different documents. Our method, in this dissertation, is similar to KELSI in terms of indexed-concept and query expansions.

3.6 Discussion

Our experimental results showed that the MIR model provides effective performance in terms of recall, precision, and topical ranking. The results demonstrate that the MIR model achieved high average precision and high precision at different recall levels. The following table shows the improvement achieved by MIR model over different runs in TREC-9.

Table 3.14 - Improvement Achieved by MIR over top TREC-runs

Run ID	*Average Precision*	*Improvement*	*R-Precision*	*Improvement*
CMUDIR	0.2016	**+ 37.56%**	**0.2544**	**+ 33.30%**
Mer9	0.2131	**+ 36.41%**	**0.2887**	**+ 29.87%**
Ok9	0.3538	**+ 22.34%**	**0.2712**	**+ 31.62%**
KUN	0.3640	**+ 21.32%**	**0.2228**	**+ 36.46%**
S2RN	0.4629	**+ 11.43%**	**0.3477**	**+ 23.97%**

In addition, the questionnaire-based test showed that the topical-ranking task in the defined network performs quite well, because most of the participants indicate that the relationship among successive retrieved documents is “Strong.”

The enhancement achieved by implementing MIR model over other runs submitted to TREC filtering-track resulted from the expansion of concepts through UMLS metathesaurus. We used to expand the medical concepts in order to build the

document semantic network and expand user query. Furthermore, the direct expansion of concepts shows better performance as compared with latent semantic analysis; KELSI.

3.7 Conclusion

In this Chapter, we presented a model for retrieving medical and health information to be used in a search engine by both medical and non-medical users. To assess its effectiveness, we have implemented and evaluated the model in terms of recall, precision, and ranking metrics. For this purpose, we performed two experiments to measure the performance of the proposed model. The first experiment measured the retrieval performance in terms of recall/precision. The second experiment concentrated on measuring the ranking task, according to user judgment, based on semantic document networking to rank documents according to their topics.

We can summarize the results on average performance as follows.

1. In our experiment on the TREC collection, the model achieved a higher average precision and R-Precision as compared with the top five runs in TREC-9.
2. MIR achieved higher interpolated average precision as compared with KELSI on 58 OHSUMED queries out of 63.
3. The first 10 results satisfied the requests of professional. Furthermore, 92% of professional users assess the relationship among successive retrieved documents as 'Strong'.

4. The Average precision reported by professional users for the first 10 retrieved documents is 97.48%, for the first 20 retrieved documents is 94.88%, and the average precision of all retrieved documents is 91.54%
5. The value of average recall metric at threshold value of zero is 100%. This value reflects the fact that the model is able to retrieve all relevant documents in the collection. The value of average recall metric at threshold value of 0.1 is 88%.

Indeed, the proposed model is based on the vector model, as it represents documents using vectors. However, it includes more semantic features directed to the medical domain. These features are evaluated using medical domain semantic relations.

These results indicate that the proposed model is effective and a good alternative to classical models to retrieve and rank medical and health information.

Chapter 4

ROLEX-SP: RULE-BASED CATEGORIZATION OF MEDICAL DOCUMENTS

Due to the rapid growth of free text documents available in digital form, efficient techniques of automatic categorization are of great importance. In this Chapter, we present an efficient rule-based method for categorizing free text documents. The contributions of this research are the formation of lexical syntactic patterns as basic classification features, a categorization framework that address the problem of classifying free text with minimal label description, and an efficient learning algorithm in terms of time complexity and F-measure. The framework of ROLEX-SP concentrates on capturing the correct classes of text as well as reducing classification errors.

We performed experiments in order to evaluate the ROLES-SP and assess our work as compared to state-of-the-art categorization techniques. The results indicate that ROLEX-SP outperforms other methods in terms of micro averaged F-measure.

Furthermore, the improvement of ROLEX-SP over other methods is statistically significant. Thus, the observed differences between ROLEX-SP and other methods reflect a real difference; not due to chance

4.1 Rules of Lexical Syntactic Patterns

The learning algorithm of ROLEX-SP generates rules such that: given a category $c_i \in C$, a positive pattern $p_{ci}^{+} \in P_{ci}^{+}$ associates with category ci, and a set of negative patterns $P_i^{-} (P^{-} \cap P^{+} = \phi)$, where P^{-} is the set of negative patterns associated to a specific category and P^{+} is the set of all positive patterns, the classifier H_{ci} of category ci is identified as a collection of rules. We used the rule's representation in [82] as follows:

$$c_i \leftarrow p_{ci}^{+} \in d, \quad \neg(p_{i1}^{-} \in d) \wedge \neg(p_{i2}^{-} \in d) \wedge ... \wedge \neg(p_{im}^{-} \in d) \quad \textbf{(EQUATION 4.1)}$$

If a positive example p_{ci}^{+} occurs in document d and none of the following negative patterns appear in d, the classifier assigns document d under category ci. Unlike the semantic of the rules in [82], the restriction $(P^{-} \cap P^{+} = \phi)$ imposed on the set of negative patterns to guarantee that a document might be categorized under more than one category; negative patterns are prevented from undoing the effect of other categories' positive ones.

4.2 ROLEX-SP Framework

In this section, we explain the theoretical basis of constructing a lexical-based classifier associated with a specific category. The goal is to show the algorithms to automate the inference process of lexical syntactic patterns and the conversion of these patterns into classifier's rules.

The proposed framework relies on dividing the corpus of documents into 3-sets: training set (TS) occupies 50% of the corpus; validation set (VS) occupies 25% of the corpus; and testing set represents 25% of the corpus.

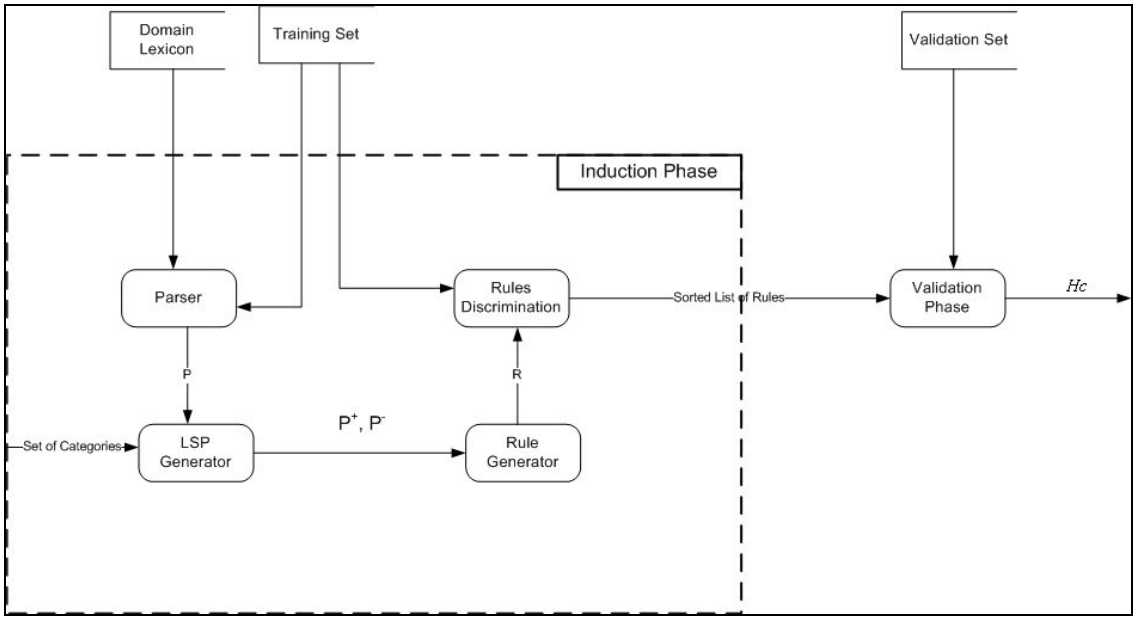


Figure 4.1- The framework of ROLEX-SP

A lexicon, in this context, is a set of formalized entries and their correspondent related concepts; such as synonyms, antonyms, or co-existing concepts. The lexicon's entries are lexical concepts that represent key or distinguished concepts to a specific category of information. Specifically, the definition of ROLEX-SP framework provides no restriction on the size or the way to construct a lexicon. One might define the lexicon for a specific domain; such as shopping, or a more comprehensive lexicon to include concepts of medical domain; such as MeSH lexicon.

Categories description is another important technique to construct a lexicon such as MeSH description of medical categories [48]. In many cases, such description might not be available. For example, in medical domain, one might tend to classify documents using categories like: symptoms, diagnosis, treatments, and medications. ROLEX-SP is able to construct classifiers to these categories using little information such as: category name, synonyms, and co-existing concepts.

The automatic inference framework of ROLEX-SP consists of two phases: induction and validation. The induction phase is responsible to learn both positive and negative patterns. Furthermore, it delivers a list of rules sorted according to their accuracy. While the function of validation phase is to validate the resulted features using the validation set of documents.

4.3 Induction Algorithms

In this section, we describe the induction algorithm to extract rules from the set of training documents. The description includes the algorithm's instructions and the mathematical basis of converting patterns into rules.

4.3.1 LSP Generator

LSP generator is responsible to extract positive lexical syntactic patterns from a labeled set of documents (training set). In order to define its functionality, first, we define the semantic structure of a lexical pattern as follows:

$$
\begin{aligned}
&Pattern \rightarrow \{NP\}^m, \quad m \geq 1 \\
&\{NP\} \rightarrow \{pre-Modifier\}\{Head-Noun\}\{post-Modifier\} \\
&\{Head-Noun\} \rightarrow \{POS\}^n\{Lexicon\ Entry\}^n\{POS\}^n, \quad n \geq 0 \\
&\{pre-Modifier\} \rightarrow \{POS\}^n\{Lexicon\ Entry\}^n\{POS\}^n \\
&\{post-Modifier\} \rightarrow \{POS\}^n\{Lexicon\ Entry\}^n\{POS\}^n
\end{aligned}
$$

Figure 4.2 - Semantic of Lexical Syntactic Patterns

where *NP* is a noun phrase and *POS* is a part of speech tag. Given a set of domain concepts, LSP is the output of parsing a given chunk of text and locating domain concepts. When the parser locate a noun-phrase contains a domain concepts, the program, then, store this phrase as a set of part-of-speech tags and domain-concepts. For example, "A progressive degenerative disease of the **brain** that causes **Loss** of **Cognitive activities**". Assume that the concepts {brain, loss, cognitive, and

activities} are lexicon entries. The LSP module converts this statement into the following pattern:

$$Head - Noun \rightarrow NN = "Brain"$$
$$Post - Modifier \rightarrow \{NN = "loss"\}\{JJ = "Cognitive"\}\{NN = "Activity"\}$$

Figure 4.3 - Example of a Lexical Syntactic Patterrn

where *NN* tag refers to a noun concept and *JJ* tag refers to an adjective one.

The output of this module is a set of patterns $P^+ = \{P_{c1}^+, P_{c2}^+, ..., P_{ck}^+\}$, where P_{ci}^+ is a set of positive patterns extracted from documents relevant to category ci. Notice that, LSP can detect the relation $d_j \in c_i$ from the labeled training set (ideal classification). Moreover, the module is responsible to deliver the set $P^- = \{P_1^-, P_2^-, ..., P_k^-\}$ in which $(P^- \cap P^+ = \phi)$ holds. The following algorithm shows the instructions to implement LSP module.

LSP_Generator

- **Goal**: to extract positive and negative lexical syntactic patterns from the set TS
- **Input**: Lexicon, TS, C.
  - **Lexicon**: Domain lexicon that contains domain concepts
  - **TS**: the set of labeled training set of documents
  - **C**: the set $C = \{c_1, c_2, ...c_k\}$ of categories where $k > 1$.
- **Output**: P^+, P^-
  - $P^+ = \{P_{c1}^+, P_{c2}^+, \cdots, P_{ck}^+\}$ A set of positive patterns relevant to all categories.
  - $P^- = \{P_1^-, P_2^-, \cdots, P_k^-\}$ A set of negative patterns relevant to all categories.
- **Method**: Apply the following instructions

Begin

1 $P^+ = \{\}, P^- = \{\}$

2 For each $ci \in C$

3 For each $d \in TSc$

4 $P = parse(d, Lexicon(ci))$

5 For each $p \in P$

6 $$accuracy(p, ci) = \begin{Bmatrix} n_{correct}(p, ci) / n_{covers}(p), & n_{covers}(p) > 0 \\ 0 \quad , & n_{covers}(p) = 0 \end{Bmatrix}$$

7 $if\ accuracy(p, ci) > 0\ then$

$P_{ci}^+ = P_{ci}^+ \wedge p$

$Elseif\ n_{correct} = 0 \wedge n_{covers} > 0\ then$

$P_i^- = P_i^- \wedge p$

8 Next p

9 Next d

10 Next ci

11 return(P^+, P^-)

End

Patterns are extracted according to the existence of category lexicons. The function *parse* is responsible to extract the set P of noun-phrases, from a given document, that contains lexicon concepts relevant to a specific category (Line 4).

To select a set of patterns among large number of generated patterns, we chose the accuracy metric as a goodness function of extracted patterns. Given a pattern p and a documents collection $D_{ci} \in C \times D$; a set of documents associated to a specific category, let $n_{covers}(p)$ be the number of documents that can be identified by (covered by) pattern p; and $n_{correct}(p, ci)$ be the number of documents that correctly classified by pattern p under category ci:

$$accuracy(p, ci) = n_{correct}(p, ci) / n_{covers}(p) \qquad \textbf{(EQUATION 4.2)}$$

The accuracy of every pattern in the set P is computed with respect to a given category ci (Line 6). In this context, accuracy reflects the capability of a given pattern to retrieve documents relevant to a specific category.

Line 7, if-statement, is intended to discriminate useful patterns; positive patterns with accuracy exceed *Zero*. In addition, if a pattern does not exist in relevant documents to category ci (*i.e.* $n_{correct} = 0$) but occurs in some documents of other categories (*i.e.* $n_{covers} > 0$), then the pattern is considered as negative one. In other words, a negative pattern is a pattern that occurs in other categories but not capable to recognize category documents.

4.3.2 Rule Generator

Rule Generator module receives the sets P^+, P^-; sets of positive and negative patterns, filter-out common patterns in P^+ and P^- (such that $P^- \cap P^+ = \phi$ holds), and generate a set of rules for each category ci such that:

$$R_{ci} = \{R_1, R_2, \cdots, R_W\} \qquad \textbf{(EQUATION 4.3)}$$

where $w = \sum_{i=1}^{k} | P_{ci}^+ |$ is the number of all generated positive patterns. The algorithm generates rules of the form

$$R : c_i \leftarrow p_{ci}^+ \in d, \quad \neg(p_{i1}^- \in d) \wedge \neg(p_{i2}^- \in d) \wedge \ldots \wedge \neg(p_{iM}^- \in d) \quad \textbf{(EQUATION 4.4)}$$

This formula is expressed as follows: if the positive pattern p_{ci}^+ occurs in document d and none of the negative patterns, which have defined with respect to category ci, the classifier will assign document d under category ci.

The filtering task is required to allow assigning a document to more than one category. For instance, the negative patterns of a specific category will not affect the impact of other categories' positive-patterns.

Notice that, in each rule only one single-positive pattern holds. And, every rule under a specific category shares the same set of negative patterns. Negative patterns are constraints on the category type, thus, negative patterns must not occur in any of categories documents in order to be classified under that category.

4.4 Validation Phase

The main goal of this phase is to validate the rules induced through induction-phase and produce a classifier H_{ci}. A text classifier, in this context, is the "best" set of rules that represent a specific category.

Definition 1 (**Representative Set RS**): given a set of rules sorted according to their accuracy, RS is the set of rules of the form

$$c \leftarrow p_{ci}^{+} \in d, \quad \neg(p_{i1}^{-} \in d) \wedge \neg(p_{i2}^{-} \in d) \wedge \cdots \wedge \neg(p_{iM}^{-} \in d), \quad \textbf{(EQUATION 4.5)}$$

where M is the number of negative patterns in category C and RS have the highest coverage metric.

Given a rule ***R*** and a set of documents $D_{ci} \in C \times D$; a set of documents belong to a specific category, let $n_{covers}(R, ci)$ be the number of documents covered by ***R*** under category ci, and $| D_{ci} |$ be the number of documents in D_{ci}:

$$coverage(R, ci) = n_{covers}(R, ci) / | D_{ci} | \qquad \textbf{(EQUATION 4.6)}$$

We define the validation phase as an optimization problem: given $R = \{R_{c1}, R_{c2}, \cdots R_{ck}\}$ where $R_{ci} = \{R_1, R_2, \cdots, R_W\}$ and $w = \sum_{i=1}^{k} | P_{ci}^{+} |$, the algorithm is responsible to produce the set $RS_{ci} = \{R_1, R_2, \cdots R_x\}, where\ x \le w\ and\ RS_{ci} \subseteq R_{ci}$, of rules such that: $Coverage(RS_{ci})$ is the maximum.

Definition 2 (Redundant Rule): a rule R_j is a redundant rule if one of the following conditions holds:

1. $(\forall i)(\exists j) : R_i = R_j \wedge i \neq j$
2. $(\forall i)(\exists j) : Coverage(R_j) \subseteq Coverage(R_i) \wedge i \neq j$

Validating rules requires pruning (removing) redundant rules; preventing rule-over-fitting. Putting aside redundant rules purify the resulted rules with general and effective ones.

Learning Process

- **Goal**: learn the best classifier to represent categories in C
- **Input**: $Lexicon, TS, C$
- **Output**: H_c
- **Method**: Apply the following instructions

Begin

- **-------Induction Phase-------**
  - $LSP\_Genrator(Lexicon, TS, C, P^+, P^-)$
  - $R_c = Rule\_Generator(P^+, P^-)$
- **-------Validation----------**
  - $H_c = \{\}$
  - For each $ci \in C$
    - $RS_{ci} = \{\}$
    - For each $R \in R_{ci}$

      $Coverage(R, ci) = n_{covers}(R \setminus R_{ci}) / |D_{ci}|$

      If $Coverage(R, ci) \geq Threshold$ Then

      $RS_{ci} = RS_{ci} \wedge R$

      Else skip //Remove Redundant Rules//
    - Next R_i
    - $H_{ci} = RS_{ci}$
    - $H_c = H_c \wedge H_{ci}$
  - Next ci

End

4.5 Time Complexity

The time required to construct a classifier for a specific category of documents is computed as the sum of the time complexity required for induction and validation phases.

Proposition 1. Algorithm LSP generator learns positive and negative patterns in time $O(nq^2)$ where n is the average number of noun-phrases with lexicon terms in the training set and $q = |TS|$ is the size the training set (number of documents).

Proof. We observed that $|TSc|$ is bounded by $q = |TS|$. $|C|$, the number of categories, is bounded to a small constant value c. The function *Parse* (Line 4) requires time $O(n)$, since it needs to scan every noun-phrase in the training set. In addition, the formula *accuracy* (Line 6) requires time $O(q)$ to find the correct and the cover sets. The For-Loop (Lines 5-8) requires time $O(nq)$ since P is bounded to n. The For-Loop (Lines 3-9) requires time $O(nq + nq^2)$ which is bounded to $O(nq^2)$. Finally, the outer For-Loop (Lines 2-10) requires time $O(cnq^2)$ which is bounded to $O(nq^2)$ because c is a small constant.

Proposition 2. The Rule-Generator algorithm generates rules associated with all categories in time $O(n)$ where n is the average number of patterns in the training set.

Proof. We observed that the sets $P^+ and\ P^-$ are bounded to n. The algorithm needs to construct a rule for every positive pattern $O(n)$ and filter-out common pattern

in $P^+ and P^-$ $O(n)$. The time required by Rule-Generator algorithm is $O(2n)$ which is bounded to $O(n)$.

Proposition 3. The Induction phase induces classification rules in time $O(nq^2)$

Proof. The time required to extract patterns and generate rules is $O(n + nq^2)$ which is bounded to $O(nq^2)$.

Proposition 4. The Validation phase validates classification rules in time $O(n^2)$.

Proof. The number of rules $|R_{ci}|$ is bounded to *n* since every rule represents a positive pattern in P^+. The formula *Coverage* requires time *q* to scan for the number of documents covered by a specific rule. Thus, time complexity for the inner-loop is $O(nq)$.

Proposition 5. The Learning process, which consists of Induction and Validation, requires polynomial time $O(nq^2)$.

Proof. The time required to run the learning process is the sum of the time required by induction and validation phases $O(nq^2) + O(nq)$ which is bounded to $O(nq^2)$; since $nq^2 >> nq$ holds.

4.6 Running Example

In this section, we provide an example of how ROLEX-SP induces patterns from a training set of documents. To simplify the idea, assume that the training set consists

of two categories: Alzheimer and Dementia. Also, assume that every category consists of two documents. Furthermore, assume that the following table represents the medical domain lexicon created to serve the needs of explaining this example.

Table 4.1 Lexicon of Medical Terms

Term	*UMLS ID*	*Category ID*
Cause	C0678227	C01
Diseases	C0012634	C01
Impairment	C2598156	C01
Cognitive	C1516691	C01
Alzheimer	C0002395	C01
Problem	C0033213	C01
Memory	C0025260	C01
Mild	C1270972	C01
Damage	C1883709	C02
Brain	C0006104	C02
Diseases	C0012634	C02
Dementia	C0497327	C02
Memory Loss	C0751295	C02
Symptom	C1457887	C02

The terms in Table 4.1 are medical terms and the UMLS concept identifier is used to retrieve synonyms and related terms. Moreover, the last column represents to which category a concept is related. According to the pattern definition, a pattern is a noun-phrase with lexical terms. The tool will scan the text documents and extract noun-phrases. Every noun phrase that contains a lexicon's term will be considered and

evaluated as a pattern, either positive or negative. In this example, we will show the positive and negative patterns extracted from the following documents.

Document 1 (C01)

Some people with **memory problems** have a condition called **mild cognitive impairment** (MCI). People with this condition have more **memory problems** than normal for people their age.

Document 2 (C01)

People with **mild cognitive impairment**, compared with those without MCI, go on to develop **memory problems**.

Document 3 (C02)

Memory loss is a common **symptom** of **dementia**. Many different **diseases** can **cause dementia**, including **Alzheimer's disease.**

Document 4 (C02)

People with **dementia** have serious **problems** with two or more brain functions, such as **memory** and language**.**

The terms in bold format indicate domain terms; medical terms. The goal behind this example is to extract positive and negative patterns for category C01 only. In other words, we want to construct patterns that identify the category “Alzheimer” using training set of two categories: Alzheimer and Dementia.

The learning algorithm starts by scanning documents in the first category "C01" for the purpose of extracting positive patterns. When a noun-phrase that contains lexical terms of category "Alzheimer" is detected, the algorithm replaces noun-phrase tags with equivalent domain concepts. In this example, the following patterns will be extracted from the first document.

P1: Memory/NN problems/NNS mild/JJ cognitive/JJ impairment/NN

P2: Memory/NN problems/NNS

where *NN* indicates noun concept, *NNS* indicates plural noun, and *JJ* indicates adjective term.

Since both of these patterns are able to identify other documents under category (C01) and they cover no other document in other categories, the tool will consider these patterns as positive patterns.

Then, the algorithm will precede finding patterns that hold lexical terms of C01. After scanning the document in C01, the algorithm will detect the following pattern which contains the term "Alzheimer" (a key concept of C01).

P3: diseases/NNS cause/VB dementia/NN Alzheimer/NNP disease/NN]

where *NN* indicates noun concept, *VB* indicates verb, and *NNP* indicates proper noun.

Pattern P3 contains key concept of category C01 but appears in a different category. Thus, P3 is a Negative pattern of category C01. Note that, pattern P3 cover no other document in C02 which implies that it is not a candidate positive pattern of C02.

4.7 Experiments and Results Analysis

In this section, we describe the experiments of applying ROLEX-SP. We provide a description of the benchmark to be used in the experiment, the preprocessing tasks, and the performance metric to be used in order to measure the effectiveness of the proposed framework.

4.7.1 Benchmark Corpus

The OHSUMED test collection [55] (See appendix B) represents a portion of the MEDLINE medical database. In this experiment, we used the categorization corpus that consists of 20,000 documents from the OHSUMED collection released on 1991. The collection consists of the 23 Medical Subject Headings (MeSH).

Table 4.2 - Top 5-Frequent OHSUMED categories

Category Name	*Category ID*	*Size (#doc.)*
Pathological Conditions	C 23	3952
Neoplasms	C 04	2630
Cardiovascular Diseases	C 14	2550
Nervous System Diseases	C 10	1562
Disorders of Environmental Origin	C 21	1263

4.7.2 Pre and Post processing

In order to create the domain lexicon, we used the MeSH description of medical categories [48]. MeSH provides detailed description of each category in addition to sub-categories of medical information. Furthermore, it provides a list of key-concepts relevant to medical classes of knowledge.

Then, we followed the learning algorithm to extract positive and negative rules. During the experiments, the learning algorithm extracts 7823 rules from the training set (i.e. average of 340 per category). Figure 4.4 and 4.5 show examples of positive and negative patterns extracted during experiments.

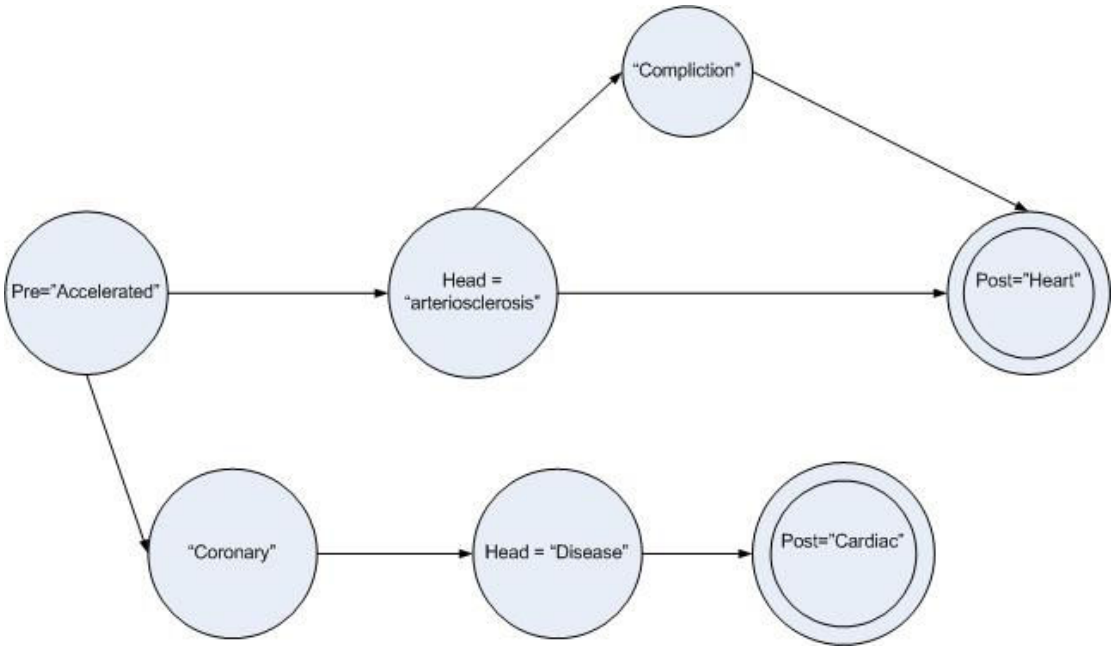


Figure 4.4 - Positive Pattern of Category C14 "Cardiovascular Diseases"

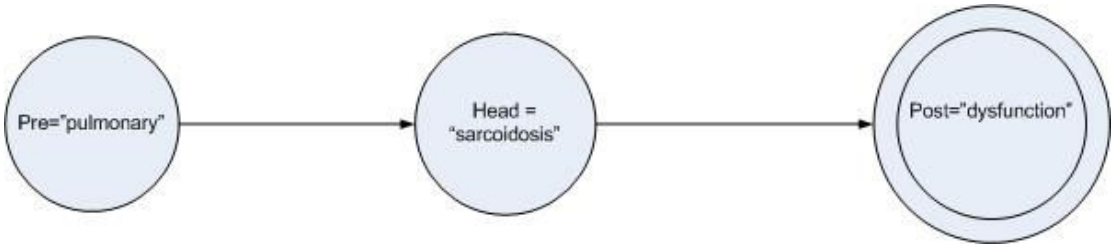


Figure 4.5 - Negative Pattern of Category C14 "Cardiovascular Diseases"

4.7.3 Performance Metric

The effectiveness of the proposed categorization method was measured in terms of recall, precision and F-measure. The standard micro-averaged precision and recall is used in order to obtain the performance over a set of categories.

$$\mu \Pr = \sum_{c \in C} | TP_c | / \sum_{c \in C} (| TP_c | + | FP_c |, \qquad \textbf{(EQUATION 4.7)}$$

$$\mu \mathrm{Re} = \sum_{c \in C} | TP_c | / \sum_{c \in C} | TP_c | + | FN_c | \qquad \textbf{(EQUATION 4.8)}$$

where $| TP_c |$ is the number of correctly categorized documents in the testing set under category c, $| FP_c |$ is the number of incorrectly categorized documents under category c, and $| FN_c |$ is the number of documents in the testing set which were not

classified under category c but should have been. The micro-averaged F-measure is defined as follows:

$$F_\alpha = \mu\,\mathrm{Pr} \times \mu\,\mathrm{Re}/(1-\alpha)\mu\,\mathrm{Pr} + \alpha\mu\,\mathrm{Re} \qquad \textbf{(EQUATION 4.9)}$$

where $\alpha \in [0,1]$.

The parameter α enumerates the relative degree of significance given to precision and recall. We choose $\alpha = 0.5$ to give equal importance for both recall and precision.

4.7.4 Experimental Results

In this section, we provide the results from implementing ROLEX-SP on OHSUMED collection [55]. We conducted experiments to measure the performance of the proposed method in terms of F-measure. Furthermore, another experiment has been conducted to measure the effect of the number of rules and the number of documents per category on the resulted performance.

4.7.4.1 Performance

In order to evaluate the performance of ROLEX-SP, we start with analyzing our data set for the purpose of choosing the best threshold on the coverage metric. Apparently, the question arises here is: how to choose the threshold value in the validation phase to produce a good performance in terms of recall and precision.

Our observations indicate that the distribution of terms frequencies in OHSUMED collection varies in non-normal form. In other words, some terms appear frequently in a category while many terms receive low frequency. Figure 4.6 shows the distribution of term-frequencies within category 14 (The distributions of all categories are available in Appendix B)

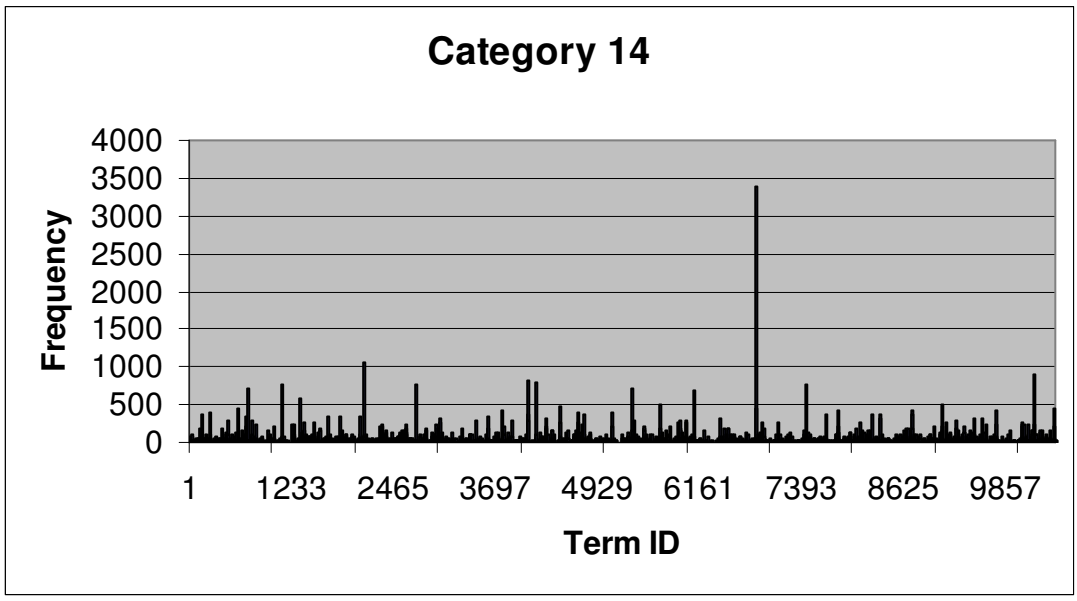


Figure 4.6 -The Distribution of Term-Frequency in Category 14

Figure 4.7 shows the maximum term frequency for each category in the collection. Notice that, the minimum term frequency is 1 for all categories. The histogram shows the variation of the maximum term frequencies among different categories.

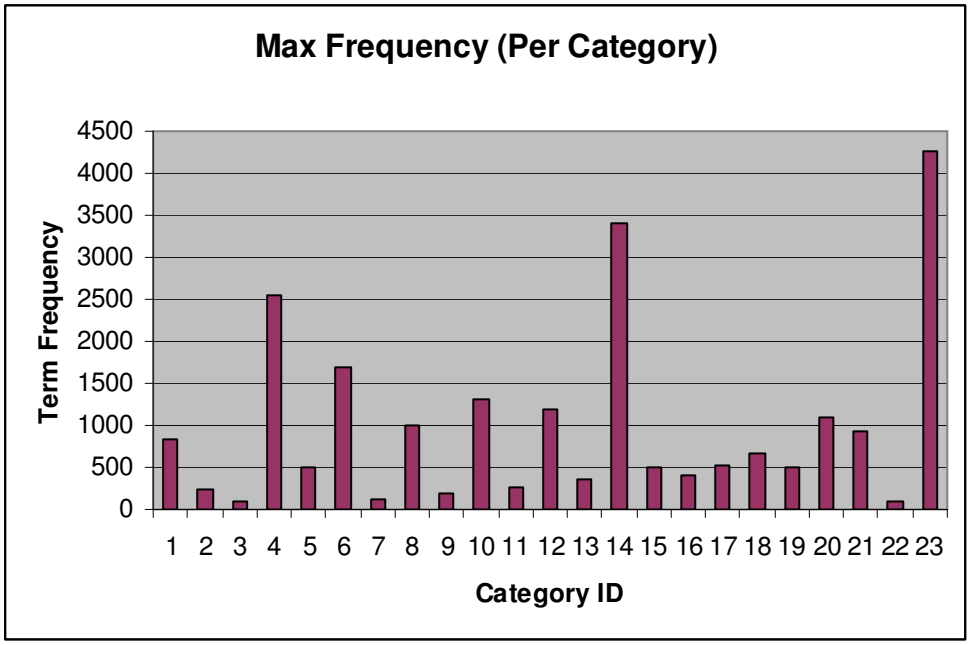


Figure 4.7 Maximum Term Frequency Per Category

A threshold value is used to evaluate the coverage metric for the purpose of selecting a representative set of rules; set of rules that maximize recall and precision. For instance, category 14 consists of 2550 documents. A rule that correctly classifies 5 documents in this category receives 0.0002 coverage value. Therefore, because of the distributions of term-frequencies within categories and among different categories vary in non-normal form, we chose low threshold values of 0.0, 0.0005, and 0.001.

The micro-averaged F-measure related to every fold is reported in Table 4.3 with respect to different threshold values. In cross-validation, the dataset is partitioned into k mutually exclusive subsets; in this case k=5. For each iteration, one fold is considered as a testing set and the remaining ones serve as training sets.

Table 4.3 - The Results of The 5-Fold Cross Validation

	μ F-Measure Threshold=0.0	*μ F-Measure Threshold=0.0005*	*μ F-Measure Threshold=0.001*
Fold 1	73.38	70.08	51.72
Fold 2	73.31	69.10	53.64
Fold 3	72.43	69.14	51.05
Fold 4	73.05	68.87	53.52
Fold 5	72.63	69.81	53.77
Average F-Measure	***72.96***	***69.40***	***52.74***
Average Recall	***78.43***	***67.36***	***47.45***
Average Precision	***68.2***	***71.56***	***59.37***

Table 4.3 shows that the performance measures reported for each fold are close to each other. This result is normal when the dataset consists of homogenous data (documents) and the size of folds (sub-datasets) is exactly equal. The average micro-averaged F-measure reported in this experiment at threshold value of 0.0 is 72.96%, threshold value of 0.0005 is 69.4, and threshold value of 0.001 is 52.74. Furthermore, our findings show that at threshold value of 0.0 the average F-measure and average recall measure is higher than those reported for other threshold values. Moreover, the best precision achieved at threshold value of 0.0005.

The results in Table 4.3 indicate that the threshold correlate negatively with the average recall; the higher the threshold value the lower the average recall is. This situation resulted from ignoring positive rules that achieved low coverage as compared with the experimental threshold. Thus, the number of correctly classified documents decreases.

On the other hand, we noticed that the average precision value at threshold value of 0.0 is less than the average recall, while the average precision becomes higher than the average recall when the threshold value is greater than 0.0. This result indicates that using a threshold value greater than 0.0 decreases the number of misclassified documents as compared to the total number of documents covered by rules.

4.7.4.2 Sensitivity Analysis

In this section, we analyze the performance of ROLEX-SP to measure how the number of rules affects the performance in terms of F-score. The purpose is to explain how the learning and categorization methods are affected by these parameters.

Table 4.4 shows the performance of ROLEX-SP as a function of the number of rules (refer to fold 1). The Table shows the performance on 10 N% intervals; where N is the percentage of rules.

Table 4.4 – The Effect of the Number of Rules on F-measure

	10%	*20%*	*30%*	*40%*	*50%*	*60%*	*70%*	*80%*	*90%*	*100%*
F-measure	12.1	17.51	23	27.12	34.92	41.98	58.76	67.15	72.61	73.38

The results indicate that the higher is the number of rules, the higher is the performance is in terms of *F*-measure. Notice that, during the learning process, every redundant rule is removed. Furthermore, the effect of the rules depends heavily on the lexicon entries since concepts in the lexicon must be descriptive to the domain. The experiment conducted in this research relies on specialized lexicon of medical concepts (MeSH).

In order to analyze the effect of category size on the overall performance of the categorization method, we compute the F-measure for each category (macro F-measure) in separate and report the performance of the 5 most frequent categories in

Table 4.5. The results show that the size of the category does not affect the performance of ROLEX-SP.

Table 4.5 - Average F-Measure for Five Selected MeSH Categories at threshold=0.0 (macro F-measure)

Category	*#doc*	*F-Measure*	*#rules*	*#negative*
C23	3952	61.81	510	73
C14	2630	86.17	463	57
C04	2550	74.94	441	67
C10	1562	67.35	397	62
C06	1263	70.09	391	43

4.7.4.3 Performance Comparison

In this section, the performance of 5-algorithms is reported in Table 4.6. The goal is to compare our proposed method with state-of-the-art induction algorithms: Naïve Bayes (NB), C4.5, Ripper, and Poly-SVM, in addition to OLEX. The micro-averaged F-measure scores have been computed on OHSUMED by each algorithm on every cross-validation fold. Notice that, we compared our method with other algorithm at threshold value of 0.0.

Table 4.6 - F-measure on Each Fold [82]

	NB	*C4.5*	*Ripper*	*Poly-SVM*	*Olex*	*ROLEX-SP*
Fold1	62.00	58.84	60.79	66.19	66.46	***73.38***
Fold2	62.37	58.67	59.84	65.24	65.97	***73.31***
Fold3	62.97	59.15	59.78	66.59	66.34	***72.43***
Fold4	62.79	58.68	60.15	66.21	65.30	***73.05***
Fold5	62.40	59.01	60.51	65.93	66.35	***72.63***
Avg μ-F	62.51	58.87	60.21	66.03	66.08	***72.96***

The results indicate that ROLEX-SP outperforms other methods. The application of lexical syntactic based rules reduces misclassified documents. Traditional term and phrase-based features perform well in enhancing the recall measure by increasing the number of correctly classified documents. LSP, on the other hand, increases recall metric by producing relaxed rules by filtering out redundant ones. Also, it reduces classification errors resulted from the presence of category's terms or phrases in irrelevant documents.

Most important, the strong definition of our lexicon (MeSH) plays a significant role in producing the patterns. MeSH facilitates capturing key concepts and allows ROLEX-SP to generate category-specific patterns.

4.8 Related Work

In this Chapter, we defined the learning algorithm of rule-based text categorization according to the description of the inductive logic programming (ILP) [83]. Similar to ILP, we aimed to implement two types of features to identify classes; positive and negative features. Positive features represent patterns of knowledge that accurately identify classes. While negative features represent patterns that appear in the text but refer to some other classes.

We defined the text categorization problem, according to the learning description of the inductive logic programming, as follows: given

1. A finite set C of independent categories of the form $\{c_1, c_2, ..., c_k\}$ where $k > 1$, meaning that there is more than one defined category and the classification results of a category do not affect the classification results of other categories.
2. A set $D = \{d_1, d_2, ..., d_N\}$ of text documents such that $\forall(j)\exists(\ S \subseteq C \wedge |\ S\ | = v) : d_j \in S$ where $1 \le v \le k$ and $1 \le j \le N$, meaning that a document might belong to more than one category; S is a subset of one or more categories.
3. A set P_{ci}^{+} of positive patterns consisting of ground logical facts of the form $p_{ci}^{+} \in D_{ci}$ such that $(p_{ci}^{+} \in d \wedge d \in D_{ci}) \Rightarrow d \in ci$; a positive pattern

under category ci that occurs in the subset D_{ci}, which represent a set of documents that belong to category ci.

4. A set P_i^- of negative facts; patterns that appear in a document but does not refer to category ci.

construct a classifier H_{ci} that is consistent with all positive and negative patterns. In other words, the classifier is a set of rules to predict a category or set of categories of a given documents based on the presence or absence of some patterns in that documents.

Given background knowledge, a set of positive examples, and a set of negative examples, a classifier is required to assign a text-document to a set of categories if the positive rules occur in a document but not negative ones. To address the multi-class classification of documents, we extend the definition of positive and negative examples in ILP by restricting the definition of negative patterns to be independent from positive ones of other categories. Our proposed learning process satisfies the learning properties in [84] of ILP. Furthermore, our learning algorithm is PAC-learnable [85]; a category C is efficiently PAC-learnable if there is a learning algorithm runs in time polynomial.

In [82], the learning algorithm generates rules such that: given a category $c_i \in C$, a positive pattern $p_{ci}^+ \in P_{ci}^+$ associate with category ci, and a set of negative patterns $P_i^- (P^- \cap P^+ = \phi)$, where P^- is the set of all negative patterns and P^+ is the

set of positive patterns, the classifier H_{ci} of category ci is defined as a set of rules. We used the rule's representation in [82] as follows:

$$c_i \leftarrow p_{ci}^{+} \in d, \quad \neg(p_{i1}^{-} \in d) \wedge \neg(p_{i2}^{-} \in d) \wedge ... \wedge \neg(p_{im}^{-} \in d) \quad \textbf{(EQUATION 4.10)}$$

If a positive example p_{ci}^{+} occurs in document d and none of the negative patterns occur in d, the classifier assigns document d under category ci. Unlike the semantic of the rules in [82], the restriction $(P^{-} \cap P^{+} = \phi)$ imposed on the set of negative patterns to guarantee that a document might be categorized under more than one category; negative patterns are prevented from undoing the effect of other categories' positive ones.

In the following subsections, we provide a description of similar method and techniques in the literature. The description include similarities between these techniques and ROLEX-SP

4.8.1 OLEX

OLEX [82] is a rule-based learning method based on d-terms as a feature to distinguish text categories. In OLEX, the classification features have selected as discriminative terms; positive and negative. For instance, negative terms, in OLEX, are defined as constraints on a category but these terms might be positive to another category. Therefore, the description of OLEX does not provide clear evidence about

how OLEX deals with this situation. ROLEX-SP, on the other hand, addresses this problem by learning an independent set of negative patterns; independent from positive ones. In addition, ROLEX-SP learning process is more efficient than OLEX in terms of time complexity; $O(nq^2)$ as compared to $O(nq^3)$.

4.8.2 SWAP

SWAP-1 [14] is a rule-based induction method to construct classifiers using a dictionary of related terms or phrases. The basic idea of SWAP-1 is to find a set of attributes to represent one class of knowledge. Defining a dictionary of terms and phrases makes SWAP-1 modifiable technique; adapting different terminologies relevant to different domains. Unlike SWAP-1, ROLEX-SP is intended to solve multi-class categorization problem by learning the intersection area among different categories rather than extracting attributes to increase the distance between classes.

4.8.3 Sequential Covering Algorithms

RIPPER [15], CN2 and AQ [24] are sequential covering algorithms that learn rules directly from the training set without having to create a decision tree for post-induction. Classification features are generated by discriminating association rules using coverage metric. Our proposed method is similar to rule-based sequential algorithms in that it learns rules through an induction learning algorithm based on coverage and accuracy metrics.

In the sequential induction of rules, each time a rule is learned and the tuples covered by the rule are removed, and the process repeats on the remaining tuples. Our proposed method groups the rules according to their relation to categories, then, removes tuples learned by a rule for a subset of the training data Dc. The definition of the vocabulary (lexicon) allows ROLEX-SP to generate more accurate rules in comparison with RIPPER, CN2, and AQ.

4.8.4 Decision-Tree Induction Algorithms

ID3 decision tree [86], ID6NB [87], and C4.5 [88] are algorithms that construct a decision tree from the training set in order to induce rules. Each node in the tree represents a test on an attribute. DT induction algorithms are widely used to induce relations among datasets. In [89] and [90], results showed the problems of using keyword-based features to classify text documents. In addition, the experiments in [90] indicated that the more information provided about the context the more accuracy achieved in classification. The goal of constructing lexicon, in ROLEX-SP, is to provide background knowledge to the learner, which reduces misclassified documents. The application of rule-based lexical syntactic patterns require minimal description of class labels; making the lexicon more dynamic to adapt different aspects of user and application requirements.

Furthermore, our method extracts rules without any post-induction phase. For example, C4.5 algorithm is used to build a decision-tree in order to induce the rules.

Our direct induction method provides more relaxed set of rules in terms of coverage criterion resulted in better performance in categorizing free text-documents.

4.8.5 ASPECT-BASED Classification

In [91], the aspect-based multi-class categorization technique requires learning a set of intermediate aspect variables that encode properties of the labels. The classification problem is defined as a structured learning problem with constraints on assignments to aspect variables. Unlike this technique, ROLEX-SP induced rules without intermediate phase of optimization (post-optimization). Moreover, our technique relies on negative text patterns rather than penalty variables to reduce misclassified documents.

4.9 Discussion

Our experimental results showed that ROLEX-SP framework is an effective method to categorize free-text documents. In addition, we showed that ROLEX-SP constructed classifiers that are efficient in terms of standard F-measure and time complexity. The results demonstrate that ROLEX-SP achieved higher average F-measure in compare with state-of-the-art methods. The following table shows the improvement achieved by ROLEX-SP over other methods.

Table 4.7 - Improvement Achieved by ROLEX-SP

Method	*Average F-measure*	*Improvement*
NB	62.51%	+ 10.45%
C4.5	58.87%	+ 14.09%
Ripper	60.21%	+ 12.75%
Poly SVM	66.03%	+ 6.93%
OLEX	66.08%	+ 6.88%

ROLEX-SP is the first method to apply lexical-syntactic patterns as a feature to represent free text. The application of LSP as basis for constructing rules resulted in better performance in compare with other methods. Furthermore, the use of lexical syntactic patterns reduces classification errors resulted from the existence of a domain concept (term-based frequency). Negative patterns have been constructed to address this issue.

ROLEX-SP learning algorithm is efficient in terms of learning general rules and time complexity. The rules generated by ROLEX-SP are filtered (see definition 2) to include highly-coverage rules; handle redundancy issue. Also, we showed in section 3.5 that the time complexity of ROLEX-SP is $O(nq^2)$, which is better than many rule-based learning methods such as OLEX and RIPPER.

The improvement of ROLEX-SP over other methods is statistically significant. Thus, the observed differences between ROLEX-SP and other methods reflect a real difference; not due to chance. Table 4.8 summarizes the confidence intervals resulted from applying paired t-test to compare ROLEX-SP with other methods.

Table 4.8 - Statistical Student's Paired t-Test (95% Confidence Intervals and 4-degree of freedom)

	Confidence Interval
NB	[9.54 -11.37]
C4.5	[13.34 -14.84]
Ripper	[12.13 -13.35]
Poly-SVM	[5.92 -7.93]
OLEX	[6 -7.75]

Finally, although ROLEX-SP achieved higher performance, we believe that the strong definition of MeSH (lexicon) categories plays a significant role in these enhancements. Thus, the definition of the lexicon, which is an integral part of ROLEX-SP framework, is a major limitation; making ROLEX-SP not applicable on domains where categories were not defined by subject headings.

4.10 Conclusion

In this Chapter, we presented a framework for categorizing free text documents. The contributions of this research are the formation of lexical syntactic patterns from domain lexicon to solve multi-class classification problem, a categorization framework that addresses the problem of classifying free text with minimal label description, and an efficient learning algorithm in terms of time complexity and F-measure to induce categorization rules.

We performed experiments for the purpose of evaluating the proposed framework and compare it with well known algorithms in the literature. The results indicated that ROLEX-SP outperform other methods in terms of micro averaged F-measure. Also, the improvement achieved by ROLEX-SP is statistically significant. The use of lexical syntactic patterns, both positive and negative, contributes on increasing the accuracy of ROLEX-SP over other methods.

In addition, we also provided a sensitivity analysis to the performance of ROLEX-SP to measure how the number of rules affects the performance of our method. The results indicated that ROLEX-SP affected by the number of rules positively. On the other hand, our observations during experiments indicated that the number of documents in the training set is not correlated to the overall performance.

The results indicated that ROLEX-SP is a good alternative as compared with other text categorization methods if there would exists a vocabulary of concepts that defined to describe categories in a domain specific environment.

CHAPTER 5

CONCLUSION

This dissertation presents a model to construct document semantic network for the purpose of enhancing the retrieval and ranking of medical documents. Furthermore, it presents a text categorization framework based on lexical syntactic patterns. In this Chapter, theoretical and technical contributions of this study are presented. Next, we present the limitations of the study in terms of design and implementation. The Chapter, then, concludes and recommends further research.

5.1 Summary of Work

In this dissertation, we have explored the effect of applying multiple semantic features in a mathematical similarity model to retrieve and rank domain specific knowledge, specifically medical domain. Moreover, we studied the impact of applying lexical syntactic patterns to categorize medical documents with minimal label description.

In Chapter 3, we described our methodology to formulate the similarity model that combines three features. Also, we provided detailed design description and implementation issues to realize the proposed model. In addition, we explained in detail two experiments to evaluate the proposed model.

In the first experiment, we applied our model to OHSUMED collection. Then, we compared the ranking and retrieval performance of MIR model with the results reported in TREC-9 and the results of KELSI method. In the second experiment, we used to evaluate the proposed model using a collection of full text documents. We distributed a questionnaire to two groups of users in order to measure ranking, recall, and precision metrics.

In Chapter 4, we explained the theoretical and mathematical basis of a rule-based categorization method to classify medical documents. Furthermore, we proposed the application of lexical syntactic patterns as a classification feature to categorize medical documents with minimal labels. Also, we explained the experiment to evaluate our method and compare the results with well known and similar methods in the field of rule-based categorization.

5.2 Research Contribution

The contributions of this research in the field of ranking and categorizing medical documents can be summarized as follows:

- A similarity model that combines multiple semantic features to model the relationships among documents containing medical and healthcare information.

The purpose is to overcome the frequency anomaly of traditional methods and retrieve more accurate results by shrinking the hit-list via reduction of the maximum number of relevant documents, which results in high precision.

- A system that facilitates medical and non-medical searching by expanding user queries with related concepts through the use of a specialized medical lexicon and a metathesaurus. The system then attaches user queries to a network of documents and computes similarity based on a set of predefined semantic features.
- A ranking method that sorts highly relevant documents toward the top of the hit-list. The ranking task is implemented on top of a semantic document network created to rank documents according to their topics.
- A technique to automatically formulate lexical syntactic patterns as basic classification features for medical documents
- A categorization framework that addresses the problem of categorizing free text with minimal label description with efficient learning algorithm in terms of time complexity and F-measure to induce categorization rules

5.3 Comments on Results

Our experimental results demonstrate the effectiveness of MIR model in terms of recall, precision and topical ranking. In the first experiment, we compared our results with top-five systems reported in TREC-9. The comparison highlights that MIR model outperforms other systems in terms of precision and R-precision. Also, we performed experiment to measure the interpolated precision per query. The goal

behind this experiment is to compare our proposed method with KELSI (Knowledge-Enhanced Latent Semantic Indexing). The experimental findings demonstrate the effectiveness of MIR by achieving higher interpolated precision on 58 queries (out of 63).

In the second experiment, we analyzed the questionnaire's data and reported the results in terms of recall, precision, and ranking assessment. The results indicate that the proposed model is effective and a good alternative to classical models to retrieve and rank medical and health information.

We achieved a significant improvement over other retrieval methods because MIR model relies on concept and query expansion using MeSH concepts. Furthermore, concept expansion with related MeSH terms plays an important part in constructing the semantic network of documents to enhance the ranking of retrieved documents

In the second phase of this dissertation, we performed an experiment to evaluate the performance of ROLEX-SP as a categorization framework for domain specific knowledge. The experiment concentrate on categorizing medical documents based on lexical syntactic patterns features. The results indicate that ROLEX-SP performs well in comparison with existing methods on short-text documents with minimal label description.

ROLEX-SP received significant performance as compared with state-of-the-art and related methods because of the strong definition of domain lexicon; MeSH lexicon. Equally important, the application of lexical syntactic patterns, as classification

features, reduces the number of misclassified text documents and, therefore, enhances the overall performance of ROLEX-SP

5.4 Limitation and Future Work

There are some restrictions and limitations that need to be taking into account in this context.

1. The proposed model is not normalized, meaning that it is not restricted to a maximum bound. This limitation may affect the understandability of the similarity values.
2. The system produces different rank for queries that belong to the same topic but contain different non-medical terms. The reason behind this anomaly is that MIR model considers only medical terms as the basic representative feature to medical text documents.
3. The semantic parameters in the model are restricted to include the title and address. We believe that including more medical-specific parameters such as medical grammars will increase the performance and the effectiveness of our technique.
4. During experiments, we have observed that the proposed model performs better with long-text in terms of precision and ranking. Although the MIR model shows good results with short-text collections (i.e., OHSUMED), we believe that the model can demonstrate better precision metrics, much like the one in the questionnaire-based experiment.

5. The effectiveness of ROLEX-SP is dependent on the noun phrase extractor algorithm since the process of constructing lexical syntactic patterns is totally depend on noun phrases.
6. The performance of ROLEX-SP is positively correlated to the predefined vocabulary (lexicon); ROLEX-SP is intended to categorize domain specific knowledge in which classes of information are described by a set of key concepts.

Similar to [5], in the future we plan to measure the impact of cognitive biases on the searching task and relevance rankings. DEBIASING strategies, such as question-answering user interface, might be applied to reduce such biases that, in turn, enhance the overall performance of the proposed system.

Another future direction is the application of lexical syntactic patterns on the categorization of multi-lingual text and other domains of knowledge. In this direction, we would like to extend ROLEX-SP vocabulary to categorize medical documents written in different languages.

REFERENCES

1. Betin Can A., Baykal N. (2007). MedicoPort: A medical search engine for all. *Computer methods and programs in biomedicine 86*, 73–86.

2. PubMed, a service of the US national library of medicine and the national institutes of health. Available from: http://www.ncbi.nlm.nih.gov/pubmed/. Accessed April 23, 2009.

3. WebMD. Available from: http://www.webmd.com. Accessed April 23, 2009.

4. Enrico W. Coiera, Victor Vickland (2008). Is Relevance Relevant? User Relevance Ratings May Not Predict the Impact of Internet Search on Decision Outcomes. *J Am Med Inform Assoc. 15*, 542–545. DOI 10.1197/jamia.M2663.

5. Annie Y.S. Lau, Enrico W. Coiera, Mbbs (2009). Can Cognitive Biases during Consumer Health Information Searches Be Reduced to Improve Decision Making?. *J Am Med Inform Assoc. 16*, 54–65. DOI 10.1197/jamia.M2557.

6. Yildiz M. and Pratt W. (2005). The Effect of Feature Representation on MEDLINE Document Classification. *AMIA 2005 Annual Symposium proceedings / AMIA Symposium*. AMIA Symposium 849-53.

7. Lewis, D. and Hayes, P., (1994). Introduction to the Special Issue on Text Categorization. *ACM Trans. Information Systems*, *12(3),* 231.

8. Han, J. and Kamber, M., (2006). *Data Mining: Concepts and Techniques*. Second Edition. San Francisco, CA: Morgan Kaufmann.

9. Sebastiani, F., (2002). Machine Learning in Automated Text Categorization. *ACM Computing Surveys*, *34(1),* 1-47.

10. Aiolli, F. and Sperduti, A, (2005). Multiclass Classification with Multi-Prototype Support Vector Machines. *Journal of Machine Learning Research*. *6*, 817-850

11. Stefan A., Athitsos, V., Yuan, Q., Sclaroff, S., (2009). Reducing JointBoost-Based Multiclass Classification to Proximity Search. *In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

12. Suna, A., Limb, E., and Liu, Y., (2009). On strategies for imbalanced text classification using SVM: A comparative study. *Decision Support Systems 48(1),* 191-201

13. Zhuang, L. Dai, H. Hang, X., (2005). A Novel Field Learning Algorithm for Dual Imbalance Text Classification. *Springer-Verlag*, *3614*, 39-48.

14. Apte´, C., Damerau, F., and Weiss, S., (1994). Automated Learning of Decision Rules for Text Categorization. *ACM Trans. Information Systems*, *12(3)* 233-251.

15. Cohen, W. and Singer, Y., (1999). Context-Sensitive Learning Methods for Text Categorization. *ACM Trans. Information Systems*, *17(2)*, 141-173.

16. Johnson, D., Oles, F., Zhang, T., and Goetz, T., (2002). A Decision-Tree-Based Symbolic Rule Induction System for Text Categorization. *IBM Systems*, *41(3)*, 428-437.

17. Li, W., Han, J., and Pei, J., (2001). Cmar: Accurate and Efficient Classification Based on Multiple-Class Association Rule. *Proc. First IEEE Int'l Conf. Data Mining (ICDM).*

18. Quinlan, J.R., (1987). Generating Production Rules from Decision Trees. *Proc. 10th Int'l Joint Conf. Artificial Intelligence*, 304-307.

19. Ricardo Baeza-Yates and Berthier Riberio-Neto (1999). *Modern Information Retrieval*. New York: Addison-Wesley.

20. Bordogna G. and Pasi G. (1993). A Fuzzy Linguistic Approach Generalizing Boolean Information Retrieval: A Model and Its Evaluation. *Journal of the American society for information science 44(2),* 70-82.

21. S. E. Robertson and K. Sparck Jones (1976). Relevance weighting of search terms. *Journal of the American Society for information Sciences, 27(3),* 129-146.

22. G. Salton and M. E. Lesk (1968). Computer evaluation of indexing and text processing. *Journal of the ACM, 15(1),* 8-36.

23. Tan, Pang-Ning, Steinbach, Michael, and Kumar, Vipin (2006). *Introduction to Data Mining*. Addison-Wesley/Prentice Hall.

24. Han, J. and Kamber, M., (2006). *Data Mining: Concepts and Techniques*. Second Edition. San Francisco, CA: Morgan Kaufmann.

25. Witten, Ian H. and Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. San Francisco, Morgan Kaufmann Publishers.

26. Hearst, Marti A., (1992). Automatic acquisition of hyponyms from large text corpora. *In Proceedings of the 14th International Conference on Computational Linguistics*.

27. Lau Y., Enrico W., Coiera, M., (2009). Can Cognitive Biases during Consumer Health Information Searches Be Reduced to Improve Decision Making?. *J Am Med Inform Assoc, 16*, 54-65. DOI 10.1197/jamia.M2557.

28. Tombros A., Villa R. and Van Rijsbergen C. J. (2002). The effectiveness of query-specific hierarchic clustering in information retrieval. *Information Processing and Management, 38,* 559-582.

29. Yang, L., Ji, D., Zhou, G., Nie Y., Iao, G. (2006). Document re-ranking using cluster validation and label propagation. *Proceedings of the 15th ACM international conference on Information and knowledge management CIKM*, 690 – 697.

30. Kurland, O. and Lee, L. (2005). PageRank without hyperlinks: structural re re-ranking using links induced by language models. *In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, 306-313.

31. Kurland, O. and Lee, L. (2006). Respect my authority!: HITS without hyperlinks, utilizing cluster-based language models. *In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, 83-90.

32. Zhang, B., Li, H., Liu, Y., Ji, L., Xi, W., Fan W.W., Chen Z., and Wei-Ying Ma, W. (2005). Improving web search results using affinity graph. *In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, 504-511.

33. Fernando Diaz (2005). Regularizing ad hoc retrieval scores. *In Proceedings of the 14th ACM intern international conference on Information and knowledge management, Bremen, Germany, ACM*, 672-679.

34. Deng, H., Lyu M., and King I. (2009). Effective latent space graph graph-based re-ranking model with global consistency. *In Proceedings of the Second ACM International Conference on Web Search and Data Mining, Barcelona, Spain, ACM,* 212-221.

35. Mitra M., Singhal A. and Buckley C. (1998). Improving Automatic Query Expansion. *In Proc. ACM SIGIR'98.*

36. Landauer, T., Foltz, P. and Laham, D. (1998). An Introduction to Latent Semantic Analysis. *Discourse Processes 25,* 259-284.

37. David M. Blei, Andrew Y. Ng and Michael I. Jordan (2003). Latent dirichlet allocation. *J. Mach. Learn. Res*. 3, 993-1022.

38. Phan, X., Nguyen, L., and Horiguchi S. (2008). Learning to classify short and sparse text & web with hidden topics from large-scale data collections. *In Proceeding of the 17th international conference on World Wide Web, Beijing, China, ACM,* 91-100.

39. Mei, O., Ling, X., Wondra, M., Su, H. and Zhai, C. (2007). Topic sentiment mixture: modeling facets and opinions in weblogs. *In Proceedings of the 16th international conference on e World Wide Web, Banff, Alberta, Canada, ACM*, 171-180.

40. Titov, I. and McDonald R. (2008). Modeling online reviews with multi multi-grain topic models. *In Proceeding of the 17th international conference on World Wide Web, Beijing, China, ACM*, 111-120

41. Wei, X. and Croft, W. (2006). LDA LDA-based document models for ad ad-hoc retrieval. *In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in info information retrieval, Seattle, Washington, USA, ACM,* 178-185.

42. H. Lieberman. (1995). Letizia: an agent that assists web browsing. *In Proceedings of the International Joint Conference on Artificial Intelligence,* 924–929.

43. Billsus, D. and Michael J. Pazzani. (1999). A hybrid user model for news story classification. *In Proceedings of the Seventh International Conference on User Modeling. Banff Canada,* 99–108.

44. Pazzani, M. and Billsus, D. (1997). Learning and revising user profiles: The identification of interesting web sites. *Machine Learning, 27,* 313–331.

45. Mooney R. J. and Roy L. (2000). Content-based book recommending using learning for text categorization. *In Proceedings of the 5th ACM Conference on Digital Libraries,* 195–204

46. Asnicar, F. and Tasso, C. (1997). ifweb: a prototype of user model-based intelligent agent for documentation filtering and navigation in the word wide web. *In Proceedings of 1st Int. Workshop on adaptive systems and user modeling on the World Wide Web,* 3–12.

47. Stefani, A. and Strapparava, C. (1998). Personalizing access to web sites: the siteif project. *In Proc. of 2nd Workshop on Adaptive Hypertext and Hypermedia*.

48. National Library of Medicine. Unified Medical Language System Fact Sheet. Available at: http://www.nlm.nih.gov/pubs/factsheets/umls.html. Accessed June 12, 2008.

49. Isaac Newton, editor. (1687). *Philosophiæ Naturalis Principia Mathematica*.

50. Brin S, Page L, editors. (1998). The anatomy of a large-scale hypertextual Web search engine. *In Proceedings of the 7th International World Wide Web Conference. Brisbane, Australia*.

51. Craswell, N., Hawking, D., Robertson, S., editors. (2001). Effective site finding using link anchor information. *Annual ACM Conference on Research and Development in Information Retrieval archive Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval.*

52. Ahmad, K., Gillam, L., Tostevin, L. (1999). University of surrey participation in trec8: Weirdness indexing for logical document extrapolation and retrieval (wilder). *In The Eighth Text Retrieval Conference (TREC-8).*

53. English Stopword List. Available at: ftp://ftp.cs.cornell.edu/pub/smart/english.stop. Retrieved June 20, 2008.

54. Robertson S., Hull DA (2000). The TREC-9 filtering track final report. In: Voorhees EM and Harman DK, Eds., *Proceedings of the Ninth Text REtrieval Conference (TREC-9). Department of Commerce, National Institute of Standards and Technology,* 25-40.

55. Hersh, W., Buckley, C., Leone, T., and Hickman, D., (1994). OHSUMED: An Interactive Retrieval Evaluation and New Large Text Collection for Research. *Proceedings of 17th ACM Int'l Conf. Research and Development in Information Retrieval (SIGIR '94), W.B. Croft and C.J. van Rijsbergen, eds.,* 192-201.

56. GUO, David, Berry, Michael W., Thompson, Bryan B., and Bailin, Sidney (2003). Knowledge-Enhanced Latent Semantic Indexing. *Information Retrieval, 6,* 225-25.

57. Trajkova, J. and Gauch, S. (2004). Improving ontology-based user profiles. *In Proceedings of the Recherche d'Information Assiste par Ordinateur, RIAO 2004,* 380–389. University of Avignon (Vaucluse), France.

58. Eirinaki, M., Vazirgiannis, M. (2003). Web Mining for Web Personalization. *ACM Transactions on Internet Technology, 3(1)*, 1-27

59. Geng X., Liu T., Qin T., Li H., (2007). Feature selection for ranking. *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. 407-441.

60. Chagoyen M., Carmona-Saez P., Shatkay H., Caraz J. M., Pascual-Montano A., (2006). Discovering semantic features in the literature: a foundation for building functional associations. *BMC Bioinformatics, 7(41)* doi:10.1186/1471-2105-7-41.

61. Dumais, S. T., Platt, J., Heckerman, D., and Sahami, M (1998). Inductive learning algorithms and representations for text categorization. *In Proceedings of CIKM. Bethesda, MD*.

62. David D. Lewis (1992). An evaluation of phrasal and clustered representations on a text categorization task. *Proceedings of SIGIR-92, 15th ACM International Conference on Research and Development in Information Retrieval*.

63. E. Charniak (1993). *Statistical Language Learning*. USA: Massachusetts Institute of Technology MIT Press.

64. F. Jelinek (1997), *Statistical Methods for Speech Recognition*. USA: Massachusetts Institute of Technology Press.

65. Mao W., Chu W. (2002). Free-text Medical Document Retrieval via Phrase-based Vector Space Model. *AMIA Annual Symposium*.

66. Boyack, Kevin W., Mane, Ketan and Börner, Katy, editors (2004). Mapping Medline Papers, Genes, and Proteins Related to Melanoma Research. *IV2004 Conference, London, UK* 965-971.

67. Rector A. and Brandt S. (2008). Why Do It the Hard Way? The Case for an Expressive Description Logic for SNOMED. *J Am Med Inform Assoc. 15*, 744–751. DOI 10.1197/jamia.M2797.

68. OMNI Medical Search. Available from: www.omnimedicalsearch.com. Accessed April 23, 2009.

69. Health On the Net foundation. Available from: http://www.hon.ch/MedHunt/. Accessed April 23, 2009.

70. Lin Y., Li W., Chen K., Liu Y. (2007). A Document Clustering and Ranking System for Exploring MEDLINE Citations. *J Am Med Inform Assoc. 14*, 651–661. DOI 10.1197/jamia.M2215.

71. Lu Z., Kim W., Wilbu W. J. (2009). Evaluating Relevance Ranking Strategies for MEDLINE Retrieval. *J Am Med Inform Assoc. 16*, 32–36. DOI 10.1197/jamia.M2935.

72. MedlinePlus, a service of the US national library of medicine and the national institutes of health. Available from: http://medlineplus.gov. Retrieved June 12, 2008.

73. Siadaty M., Shu J., Knaus W. (2007). Relemed: sentence-level search engine with relevance score for the MEDLINE database of biomedical articles. *BMC Medical Informatics and Decision Making, 7(1)* doi:10.1186/1472-6947-7-1.

74. Nicholas C. Ide, Russell F. Loane, Dina Demner-Fushman (2007). Essie: A Concept-based Search Engine for Structured Biomedical Text. *J Am Med Inform Assoc. 14*, 253–263. DOI 10.1197/jamia.M2233.

75. Gaudinat, P. Ruch, M. Joubert, P. Uziel, A. Strauss, M. Thonnet, R. Baud, S. Spahni, P. Weber, J. Bonal (2006). Health search engine with e-document analysis for reliable search results. *International Journal of Medical Informatics 75(1),* 73-85.

76. Dietze H., Schroeder M., (2009). GoWeb: a semantic search engine for the life science web. *BMC Bioinformatics 10(1),* doi: 10.1186/1471-2105-10-S10-S7.

77. Lin J. (2008). PageRank without Hyperlinks: Re-ranking with Related Document Networks. *Technical Report LAMP-TR-146/HCIL-2008-01, University of Maryland, College Park.*

78. Lin J., DiCuccio M., Grigoryan V., and Wilbur W. (2007). Exploring the effectiveness of related article search in PubMed. *Technical Report CS-TR-4877/UMIACS-TR-2007-36/HCIL-2007-10, University of Maryland, College Park, Maryland.*

79. Robertson S.E., Walker S. (2001). Microsoft Cambridge at TREC-9: Filtering track. *In Proceeding of Ninth Text REtrieval Conference (TREC-9). National Institute of Standards and Technology, Special Publication.*

80. Ault T., Yang Y. (2002). Information Filtering in TREC-9 and TDT-3: A Comparative Analysis. *Information Retrieval*, *5,* 159-187. DOI 10.1023/A:101574591176.

81. Stricker M., Vichot F., Dreyfus G., Wolinski F. (2000). Training context-sensitive neural networks with few relevant examples for the trec-9 routing. *The Ninth Text REtrieval Conference (TREC9). National Institute of Standards and Technology, special publication* 500-249.

82. Rullo, P., Policicchio, V., Cumbo, C., and Iiritano, S., (2009). Olex: Effective Rule Learning for Text Categorization. *IEEE Transactions on Knowledge and Data Engineering*, *21(8)*, 1118-1132.

83. Lavrac, N. and Dzeroski, S., (1994). *Inductive Logic Programming: Techniques and Applications*. New York: Ellis Horwood.

84. Valiant, L.G., (1984). A Theory of the Learnable. *Proceedings of 16th Ann. ACM Symposium. Theory of Computing (STOC '84),* 436-445.

85. Anthony, M. and Biggs, N, (1992). *Computational Learning Theory*. Cambridge University Press.

86. Wang, Y. And Wang, Z., (2005). Text Categorization Rule Extraction Based On Fuzzy Decision Tree. *Proceedings of the Fourth International Conference on Machine Learning and Cybernetics, Guangzhou,* 18-21.

87. Appavu, S., Rajaram, R., (2009). Knowledge-based system for text classification using ID6NB algorithm. *Knowledge-Based Systems, 22(1)*, 1-7.

88. Quinlan, J.R., (1993). *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers.

89. Kapalavayi, N., Murthy, S.N., and Hu, G., (2009). Document Classification Efficiency of Phrase-Based Techniques. *IEEE/ACS International Conference on Computer Systems and Applications.*

90. Chim, H. and Deng, X., (2008). Efficient Phrase-Based Document Similarity for Clustering. *IEEE Transactions On Knowledge And Data Engineering, 20(9).*

91. Roth, D. and Tu, Y., (2009). Aspect Guided Text Categorization with Unobserved Lables. *CDMIEEE Computer Society,* 962-967.

92. A. Kayısoglu, *LOKMAN: A Medical Ontology Based Topical Web Crawler*. Middle East Technical University, Ankara, Turkey, 2005.

93. Wu, S. and Crestani, F. (2003). Methods for ranking information retrieval systems without relevance judgments. *In Proceedings of the ACM symposium on applied computing conference* 811–816.

APPENDICES

APPENDIX A: SAMPLE OF OHSUMED COLLECTION

Infectious arthritis

Any patient who presents with an acute monarticular arthritis, especially a new asymmetric effusion with underlying joint disease, should be suspected of having a bacterial process. Because synovial fluid findings (leukocyte counts and glucose) may not be predictive of infection, bacteriologic analysis by smear and culture is necessary in the evaluation of any new synovial effusion.

A chronic monarticular process is highly likely to be infectious also, but mycobacterial or fungal etiologies frequently require appropriate culture of synovial tissue in addition to processing fluid. Acute polyarticular syndromes are seen as manifestations of disseminated gonococcal infections (DGI) and certain viral infections in adults.

Diagnostic clues include historic and physical findings (exposure history and type of rash). The major pathogen in adults remains Staphylococcus aureus, so initial therapy is directed at this organism unless urinary tract infection is present also. Proper recommended therapy for DGI is ceftriaxone because penicillin-resistant strains are present in many urban canters. Early recognition and treatment of bacterial arthritis may prevent poor outcome, particularly in elderly patients or those with underlying joint diseases.

For chronic mycobacterial or fungal infections, surgery may need to be combined with medical management.

An unusual manifestation of Paget's disease of bone: spinal epidural hematoma presenting as acute cauda equina syndrome.

Neurologic sequelae of Paget's disease of bone include involvement of the spinal cord or cauda equina due to mechanical compression by enlarged vertebrae, ischemia caused by a spinal artery, steal syndrome or neoplasm.

We describe a patient with Paget's disease of bone who presented with acute cauda equina syndrome due to a spinal epidural hematoma.

Clinicians need to recognize this entity since surgical intervention may result in a favorable outcome.

Prospective payment system and impairment at discharge.The 'quicker-and-sicker' story revisited

Since the introduction of the prospective payment system (PPS), anecdotal evidence has accumulated that patients are leaving the hospital "quicker and sicker." We developed valid measures of discharge impairment and measured these levels in a nationally representative sample of patients with one of five conditions prior to and following the PPS implementation.

Instability at discharge (important clinical problems usually first occurring prior to discharge) predicted the likelihood of postdischarge deaths.

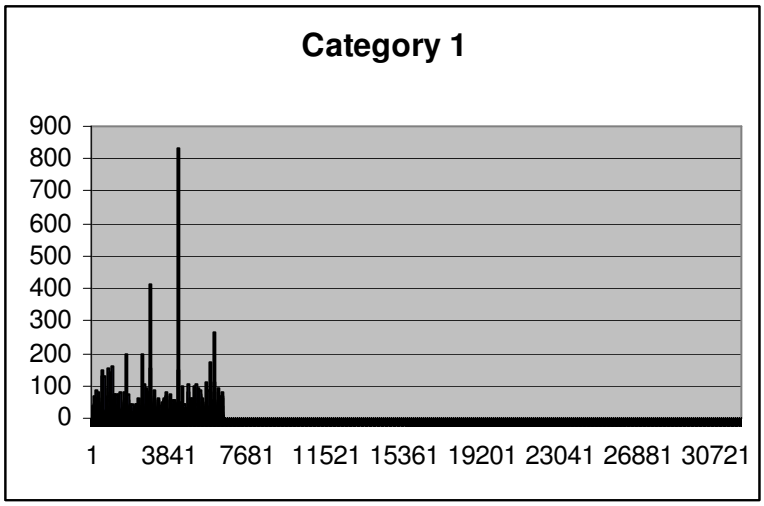
At 90 days postdischarge, 16% of patients discharged unstable were dead vs 10% of patients discharged stable.

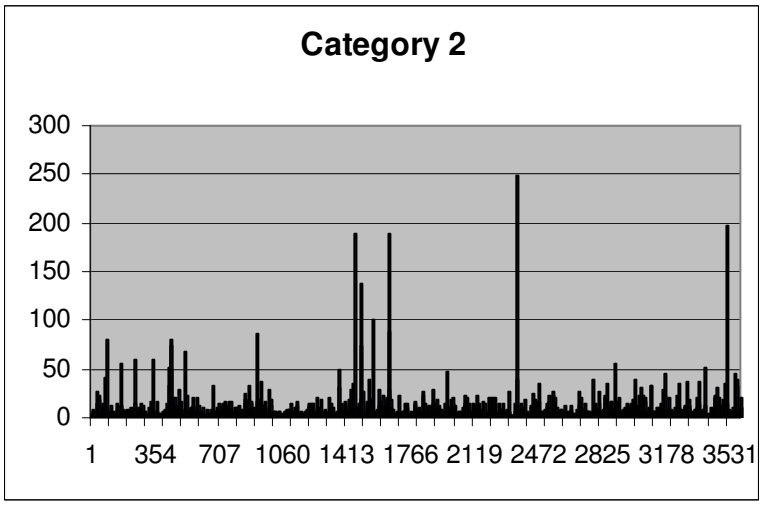
After the PPS introduction, instability increased primarily among patients discharged home.

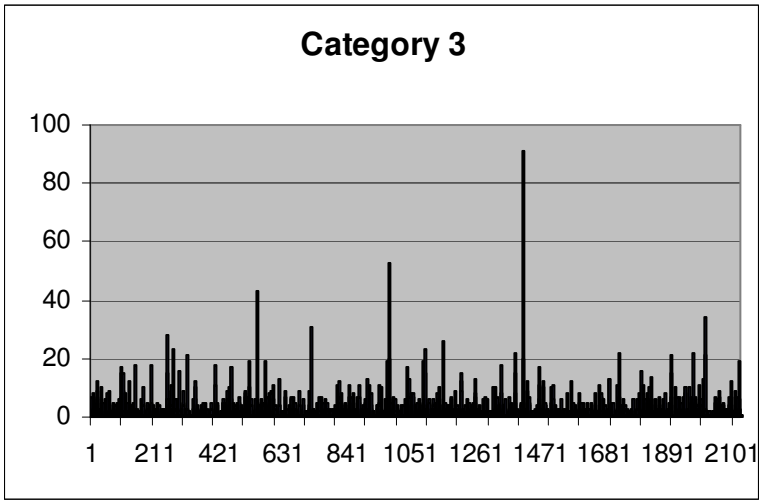
Prior to the PPS, 10% of patients discharged home were unstable; after the PPS was implemented, 15% were discharged unstable, a 43% relative change.

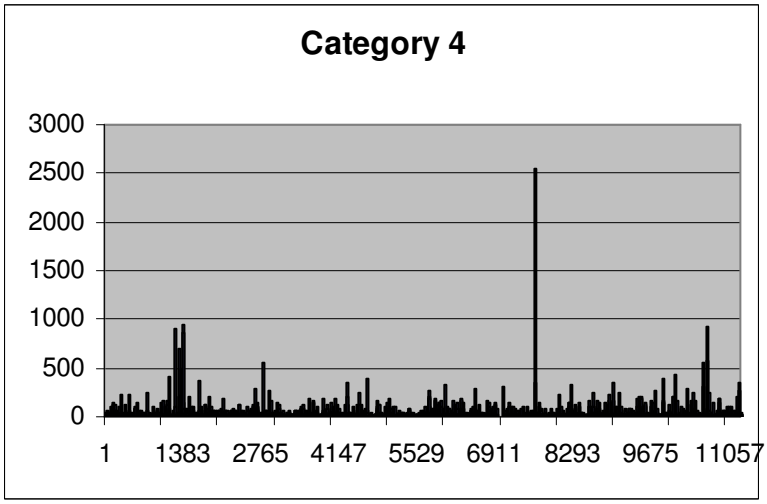
Efforts to monitor the effect of this increase in discharge instability on health should be implemented.

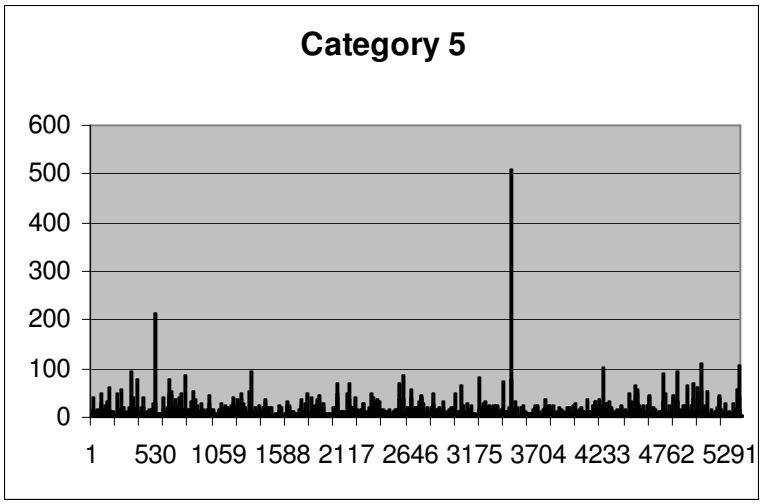
APPENDIX B: THE DISTRIBUTION OF TERM-FREQUENCIES IN OHSUMED COLLECTION

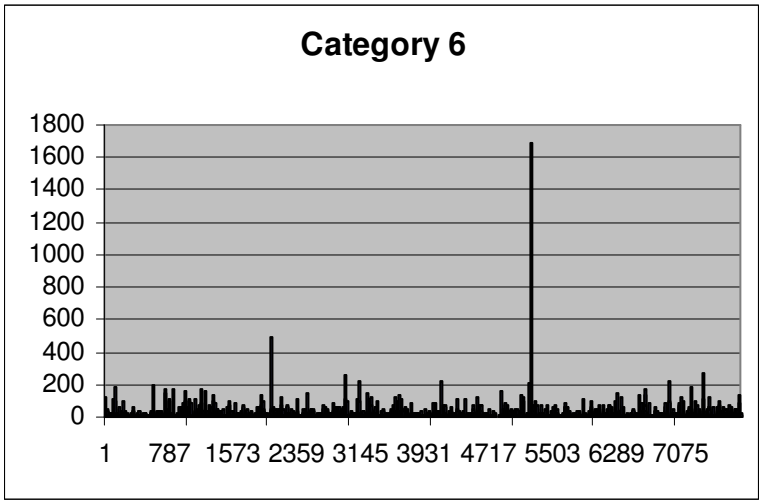


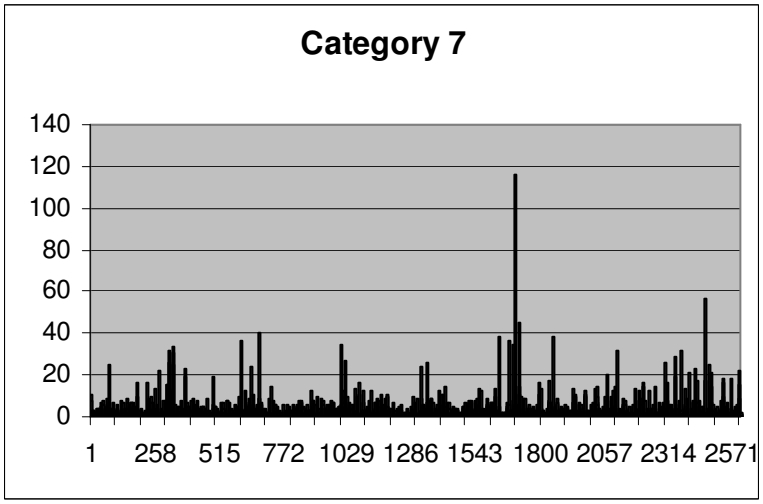


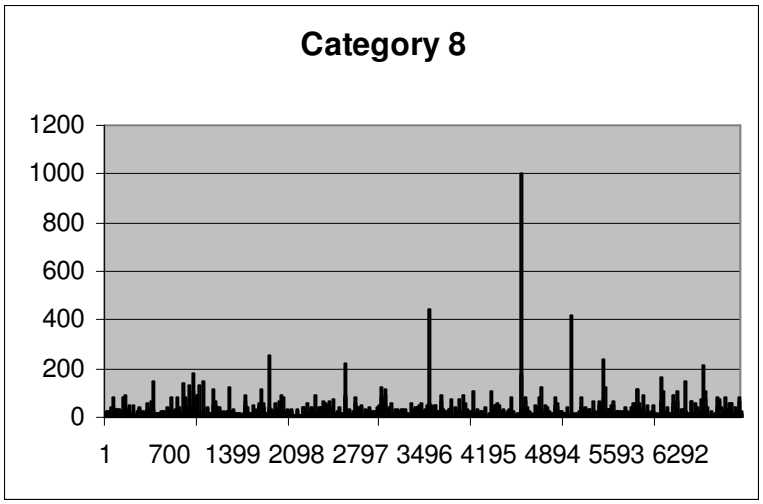

Category 3
100
80
60
40
20
0
1
211
421
631
841
1051
1261
1471
1681
1891
2101

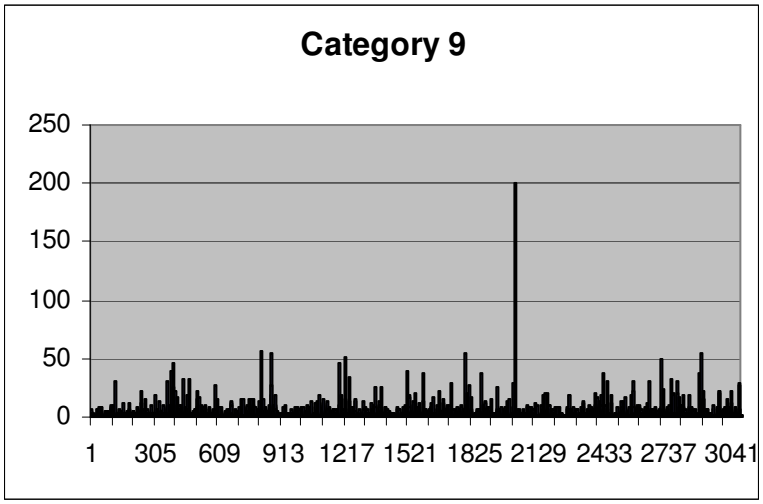

Category 4
3000
2500
2000
1500
1000
500
0
1
1383
2765
4147
5529
6911
8293
9675
11057

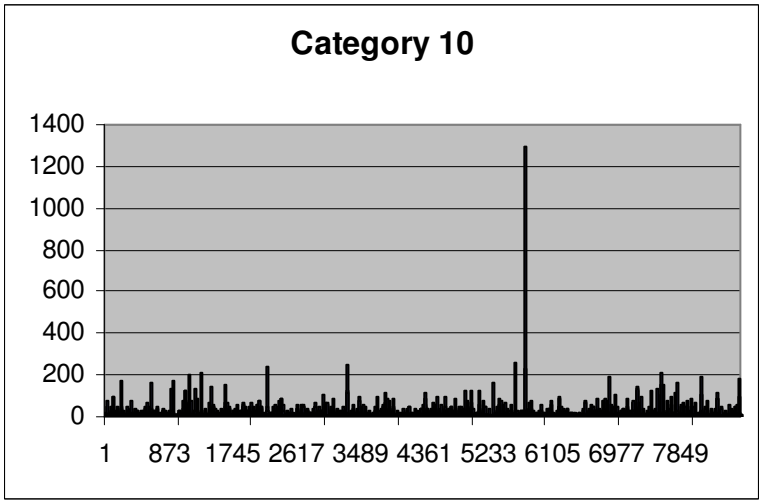

Category 5
600
500
400
300
200
100
0
1
530
1059
1588
2117
2646
3175
3704
4233
4762
5291

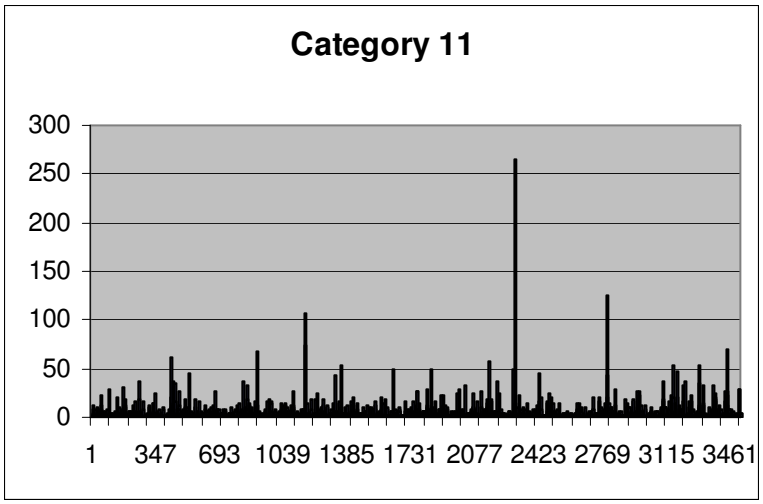

Category 6
1800
1600
1400
1200
1000
800
600
400
200
0
1 787 1573 2359 3145 3931 4717 5503 6289 7075

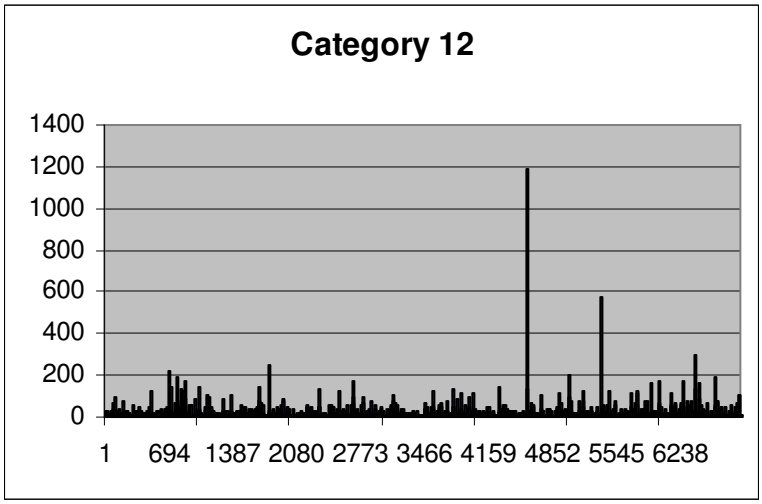

Category 7
140
120
100
80
60
40
20
0
1 258 515 772 1029 1286 1543 1800 2057 2314 2571

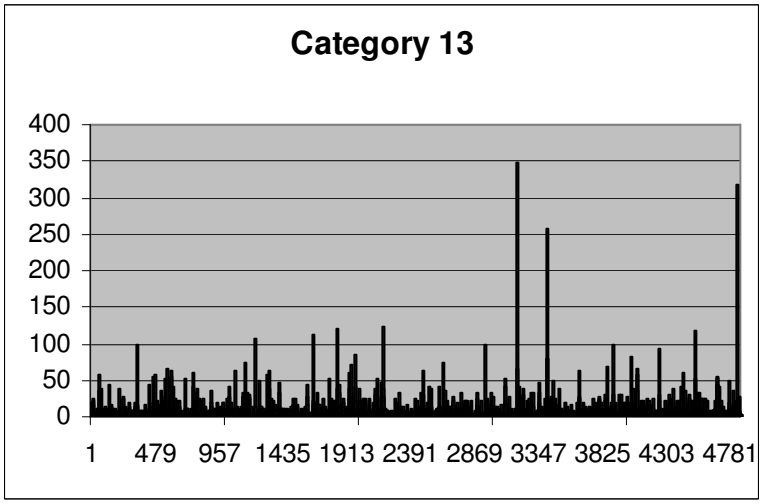

Category 8
1200
1000
800
600
400
200
0
1 700 1399 2098 2797 3496 4195 4894 5593 6292

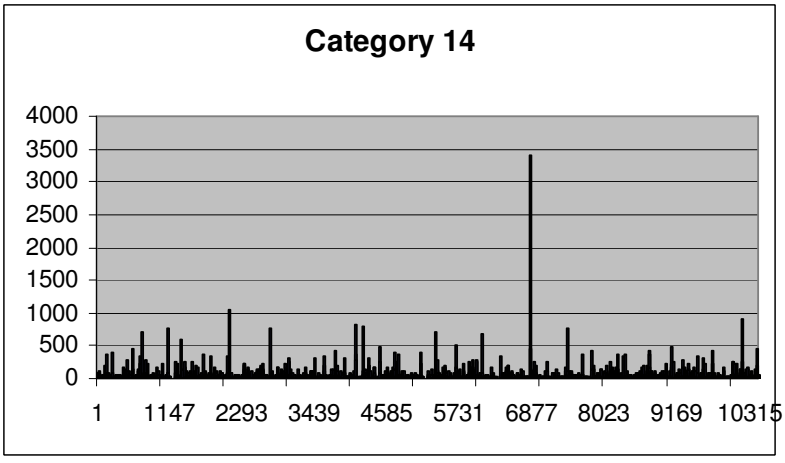

Category 9
250
200
150
100
50
0
1
305
609
913
1217
1521
1825
2129
2433
2737
3041

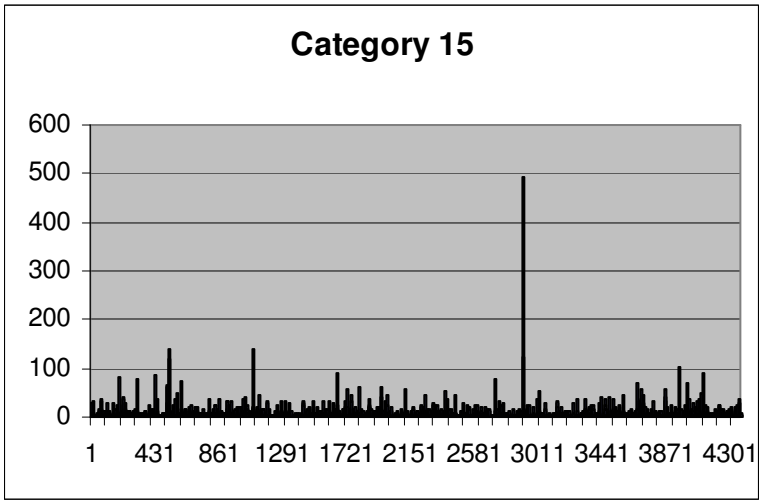

Category 10
1400
1200
1000
800
600
400
200
0
1
873
1745
2617
3489
4361
5233
6105
6977
7849

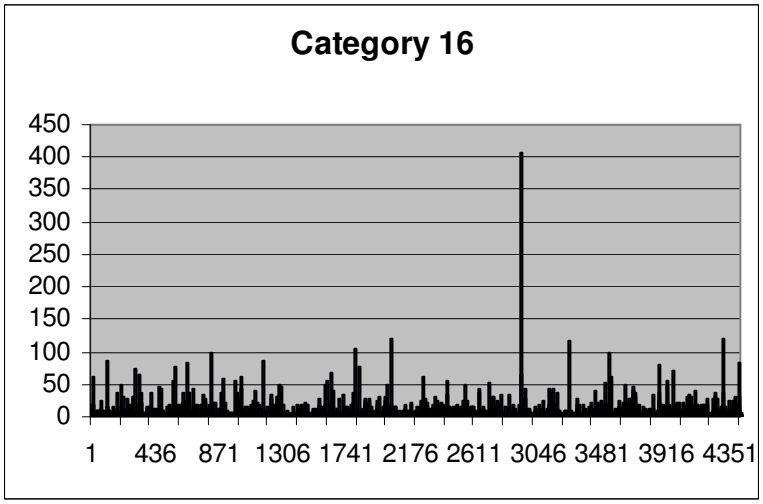

Category 11
300
250
200
150
100
50
0
1
347
693
1039
1385
1731
2077
2423
2769
3115
3461

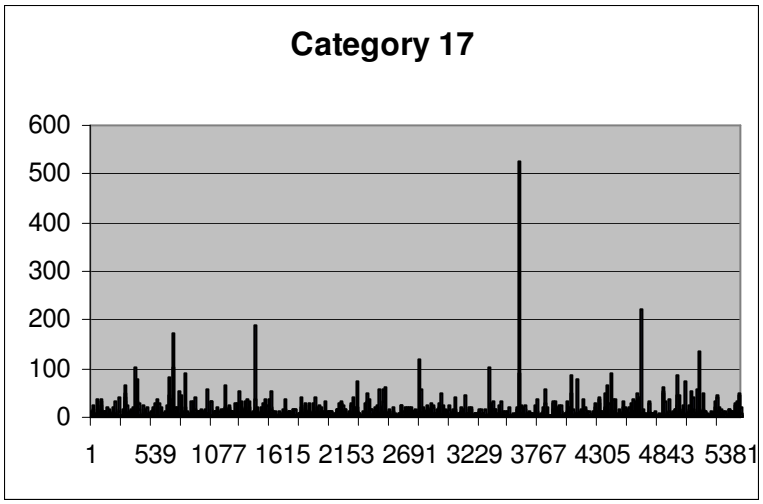

Category 12
1400
1200
1000
800
600
400
200
0
1
694
1387
2080
2773
3466
4159
4852
5545
6238

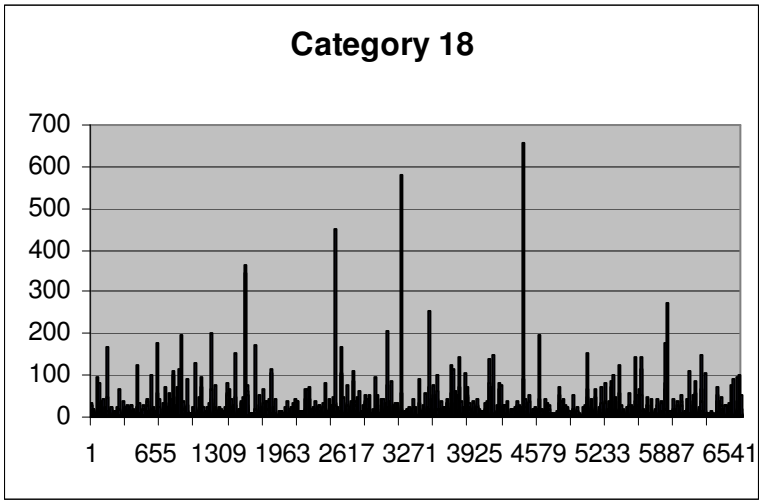

Category 13
400
350
300
250
200
150
100
50
0
1
479
957
1435
1913
2391
2869
3347
3825
4303
4781

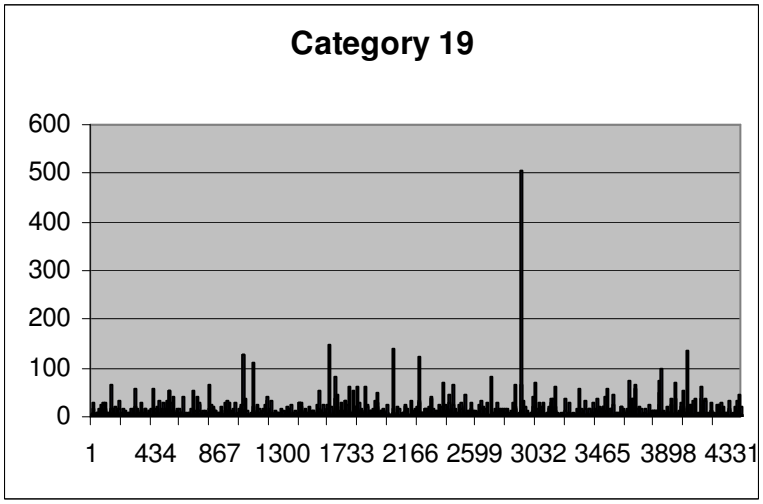

Category 14
4000
3500
3000
2500
2000
1500
1000
500
0
1
1147
2293
3439
4585
5731
6877
8023
9169
10315

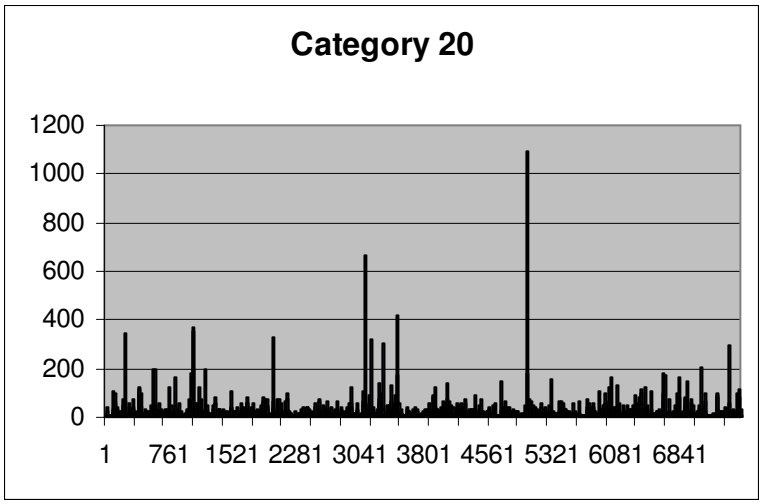

Category 15
600
500
400
300
200
100
0
1
431
861
1291
1721
2151
2581
3011
3441
3871
4301

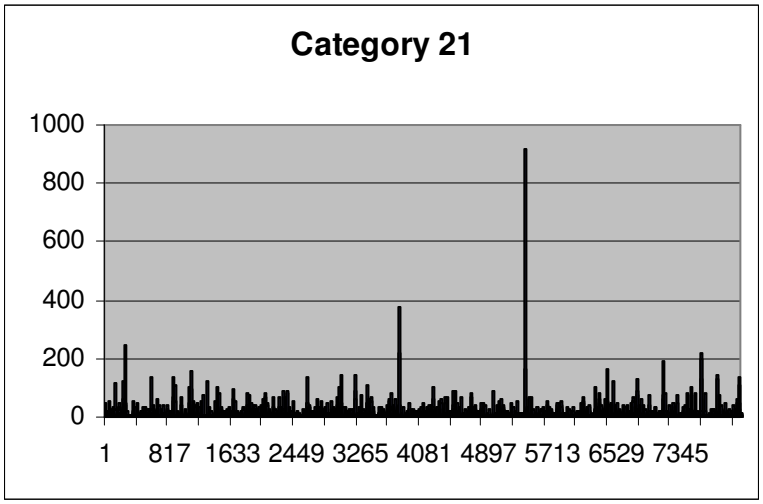

Category 16
450
400
350
300
250
200
150
100
50
0
1
436
871
1306
1741
2176
2611
3046
3481
3916
4351

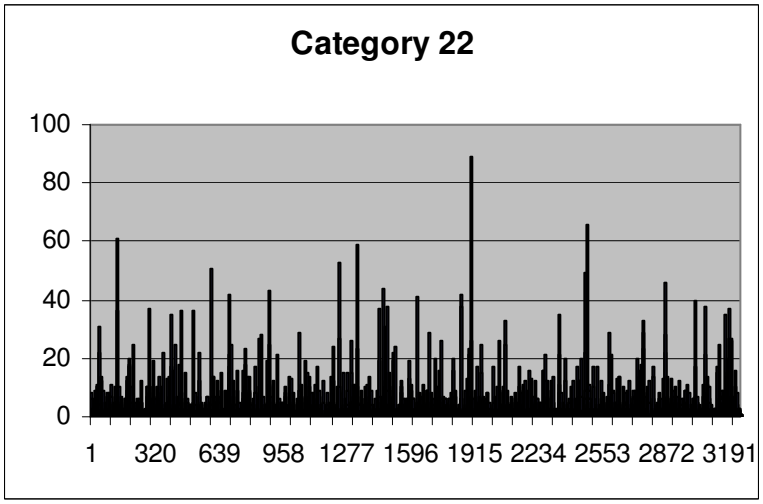

Category 17
600
500
400
300
200
100
0
1
539
1077
1615
2153
2691
3229
3767
4305
4843
5381

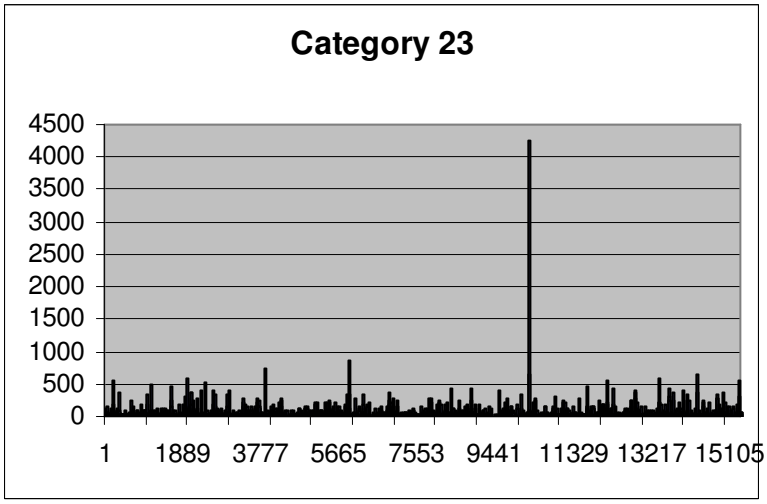

Category 18
700
600
500
400
300
200
100
0
1
655
1309
1963
2617
3271
3925
4579
5233
5887
6541


Category 19
600
500
400
300
200
100
0
1
434
867
1300
1733
2166
2599
3032
3465
3898
4331


Category 20
1200
1000
800
600
400
200
0
1
761
1521
2281
3041
3801
4561
5321
6081
6841


Category 21
1000
800
600
400
200
0
1
817
1633
2449
3265
4081
4897
5713
6529
7345


Category 22
100
80
60
40
20
0
1
320
639
958
1277
1596
1915
2234
2553
2872
3191


Category 23
4500
4000
3500
3000
2500
2000
1500
1000
500
0
1
1889
3777
5665
7553
9441
11329
13217
15105

APPENDIX C: MESH CATEGORIES OF MEDICAL KNOWLEDGE

Category ID	Category Description
C01	Bacterial Infections and Mycoses
C02	Virus Diseases
C03	Parasitic Diseases
C04	Neoplasms
C05	Musculoskeletal Diseases
C06	Digestive System Diseases
C07	Stomatognathic Diseases
C08	Respiratory Tract Diseases
C09	Otorhinolaryngologic Diseases
C10	Nervous System Diseases
C11	Eye Diseases
C12	Male Urogenital Diseases

C13	Female Urogenital Diseases and Pregnancy Complications
C14	Cardiovascular Diseases
C15	Hemic and Lymphatic Diseases
C16	Congenital, Hereditary, and Neonatal Diseases and Abnormalities
C17	Skin and Connective Tissue Diseases
C18	Nutritional and Metabolic Diseases
C19	Endocrine System Diseases
C20	Immune System Diseases
C21	Disorders of Environmental Origin
C22	Animal Diseases
C23	Pathological Conditions, Signs and Symptoms

CURRICULUM VITAE

Personal Information

Name	: AL Zamil, Mohammed
Nationality	: Jordanian
Date and Place of Birth	: AL Kuwait Dec 28, 1978
Marital Status	: Single
Phone	: +962 777 260 802
Email	: Mohammedz@yu.edu.jo

EDUCATION

Degree	Institution	Grad. Year
B.A	Yarmouk University (Jordan) Computer Science	2001
M.Sc.	Yarmouk University (Jordan) Computer and Information Sys.	2003
Ph.D.	Middle East Technical University Information Systems	2010

AREAS OF RESEARCH INTERESTS

Information Retrieval, Data Mining, Robotics Planning, Software Testing, Model Checking, Software Design Patterns, Algorithms Design, Software Adaptation, and Numerical Analysis.

Work Experience

– Feb – Sep 2001	System Analyst, Technology Lights, Kingdom of Saudi Arabia, AL-Riyadh
– Sep 2003- Feb 2006	Lecturer, YARMOUK University, Kingdom of Jordan, Irbed
– Jun-Sep 2004	Project Manager, Technology Lights, Kingdom of Saudi Arabia, AL-Riyadh
– Jan-Dec 2005	IT Consultant, CAREER XCHANGE (South Florida-USA) Jordan Branch, Amman.

Foreign Languages

Arabic and English

Publication

- Mohammed GH. AL Zamil, "Toward effective information retrieval in medical domain". Data mining in Healthcare Informatics, 23rd European Conference on Operational Research. BONN, 2009.

- Mohammed GH. AL Zamil and Aysu Betin-Can. "Toward Effective Medical Search Engines", Accepted by 5th Int. Symposium on Health Informatics and Bioinformatics (HIBIT 2010)

- Mohammed GH. AL Zamil and Aysu Betin-Can. "A Model Based on Multi-Features to Enhance Healthcare and Medical Document Retrieval". Journal of Informatics for Health and Social Care.

- Mohammed GH. AL Zamil and Aysu Betin-Can. "ROLEX-SP: Rules of Lexical Syntactic Patterns for free text Categorization". Journal of Knowledge-Based Systems (**Submitted**).

References

Available upon request