ABSTRACT

CONCEPT GRAPHS: APPLICATIONS TO BIOMEDICAL TEXT CATEGORIZATION AND CONCEPT EXTRACTION

by
Said Bleik

As science advances, the underlying literature grows rapidly providing valuable knowledge mines for researchers and practitioners. The text content that makes up these knowledge collections is often unstructured and, thus, extracting relevant or novel information could be nontrivial and costly. In addition, human knowledge and expertise are being transformed into structured digital information in the form of vocabulary databases and ontologies. These knowledge bases hold substantial hierarchical and semantic relationships of common domain concepts. Consequently, automating learning tasks could be reinforced with those knowledge bases through constructing human-like representations of knowledge. This allows developing algorithms that simulate the human reasoning tasks of content perception, concept identification, and classification.

This study explores the representation of text documents using concept graphs that are constructed with the help of a domain ontology. In particular, the target data sets are collections of biomedical text documents, and the domain ontology is a collection of predefined biomedical concepts and relationships among them. The proposed representation preserves those relationships and allows using the structural features of graphs in text mining and learning algorithms. Those features emphasize the significance of the underlying relationship information that exists in the text content behind the interrelated topics and concepts of a text document. The experiments presented in this study include text categorization and concept extraction applied on biomedical data sets.

The experimental results demonstrate how the relationships extracted from text and captured in graph structures can be used to improve the performance of the aforementioned applications. The discussed techniques can be used in creating and maintaining digital libraries through enhancing indexing, retrieval, and management of documents as well as in a broad range of domain-specific applications such as drug discovery, hypothesis generation, and the analysis of molecular structures in chemoinformatics.

CONCEPT GRAPHS: APPLICATIONS TO BIOMEDICAL TEXT CATEGORIZATION AND CONCEPT EXTRACTION

by

Said Bleik

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Information Systems

Department of Information Systems

May 2013

APPROVAL PAGE

CONCEPT GRAPHS: APPLICATIONS TO BIOMEDICAL TEXT CATEGORIZATION AND CONCEPT EXTRACTION

Said Bleik

Dr. Min Song, Dissertation Advisor Date
Associate Professor of Information Systems, NJIT

Dr. Fadi Deek, Committee Member Date
Professor of Information Systems, NJIT

Dr. James Geller, Committee Member Date
Professor of Computer Science, NJIT

Dr. Vincent Oria, Committee Member Date
Associate Professor of Computer Science, NJIT

Dr. Jun Huan, Committee Member Date
Associate Professor of Electrical Engineering and Computer Science,
University of Kansas

BIOGRAPHICAL SKETCH

Author:	Said Bleik
Degree:	Doctor of Philosophy
Date:	May 2013

Undergraduate and Graduate Education:

- Doctor of Philosophy in Information Systems,
  New Jersey Institute of Technology, Newark, NJ, 2013

- Master of Science in Computer Science,
  Lebanese American University, Beirut, Lebanon, 2005

- Bachelor of Science in Computer Science,
  Lebanese American University, Beirut, Lebanon, 2002

Major: Information Systems

Presentations and Publications:

Bleik, S., Mishra, M., Huan, J., Song, M., "Text Categorization of Biomedical Data Sets using Graph Kernels and a Controlled Vocabulary", Computational Biology and Bioinformatics, IEEE/ACM Transactions on, 2013.

Mishra, M., Huan, J., Bleik, S., Song, M., "Biomedical Text Categorization with Concept Graph Representations Using a Controlled Vocabulary", International Workshop on Data Mining in Bioinformatics, BIOKDD 2012, 2012.

Bleik, S., Song, M., and Xiong, W. "Enhancing Biomedical Concept Extraction using Semantic Relationship Weights", International Journal of Data Mining and Bioinformatics, 2012.

Song, M., Bleik, S., Yu, H. and Han, W.S. "Extracting Biomedical Concepts from Fulltext by Relative Importance in a Graph Model", International Workshop on Biomedical and Health Informatics in conjunction with BIBM 11, 2011.

Bleik, S., Xiong, W., Wang, Y. and Song, M. “Biomedical Concept Extraction using Concept Graphs and Ontology-based Mapping”, Bioinformatics and Biomedicine, 2010. BIBM '10. IEEE International Conference on, 2010.

Bleik, S., Smalter, A., Song, M., Huan, J. and Lushington, G. “CGM: A Biomedical Text Categorization Approach Using Concept Graph Mining”, Workshop of Applications of Machine Learning in Bioinformatics in conjunction with BIBM 09, 2009.

To All

ACKNOWLEDGMENT

I would like to thank my advisor, Dr. Min Song, for his continuous motivation and support throughout the past years. I was fortunate to have had the opportunity to work with him on applications of text mining, which I have been increasingly intrigued by to this day. I would also like to express my sincere gratitude to Dr. Fadi Deek for his support, encouragement, and help since the early days at NJIT. I have always appreciated our chats and his words of motivation. I would also like to thank Dr. James Geller and Dr. Vincent Oria for serving on my committee and for their feedback and kind words. I greatly appreciate their friendly nature and positive attitude, especially during critical times of my work. I am also very grateful to Dr. Jun Huan for his valuable remarks and collaboration efforts on graph mining which turned out to be a major component of my research work and revived an old interest of mine in graph theory and applications. My sincere appreciation goes to Dr. Yi-fang Brook Wu as well for her guidance and valuable comments on my research work and for the time she dedicated for our meetings and extensive discussions.

Finally, I would like to thank my family, friends, colleagues, faculty and staff for their invaluable assistance, advice, and support.

TABLE OF CONTENTS

Chapter **Page**

TABLE OF CONTENTS
(Continued)

Chapter **Page**

TABLE OF CONTENTS
(Continued)

Chapter **Page**

TABLE OF CONTENTS
(Continued)

Chapter **Page**

TABLE OF CONTENTS
(Continued)

Chapter **Page**

LIST OF TABLES

LIST OF FIGURES

Figure **Page**

CHAPTER 1

INTRODUCTION

1.1 Introduction

In a document-driven environment, such as digital library systems, the basic units of information vary in size, significance, and location. Text documents can usually be decomposed into smaller units like phrases, terms, or any domain specific knowledge that can be extracted from the content or from a data collection as a whole. Whether it is a full-text document, a webpage, or a smaller chunk of data, the text content embeds a lot of interrelated topics or concepts buried in a corpus. This embedded information, non-trivial by nature, is sometimes hard to discover or extract from the text and requires nontraditional techniques [1]–[4]. Enhancing the representation of documents is therefore essential for understanding the content and implementing better information extraction components in text mining applications and digital libraries. Text mining is an emerging discipline at the intersection of artificial intelligence, natural language processing, linguistics, statistical learning, and data mining. It involves extracting useful knowledge and hidden patterns in textual data collections, encompassing a set of automation techniques needed in managing the growing text repositories and digital libraries as well as techniques of knowledge discovery from unstructured text documents [5]. In an attempt to bridge the gap between conventional data mining techniques and unstructured text, the following study explores graph-based representations of text documents through several applied experiments. Graphs representing linked entities capture additional relational information that might be present in the text content and thus provide a basis for applying graph mining techniques that utilize the structural features embedded in the representation. In this study a model of representing text documents using graphs is presented. A graph is comprised of nodes and edges that describe relationships among them. The nodes represent concept terms identified in the text and the edges represent semantic-based relationships among these concepts within a certain domain. The relationships are defined by human experts in a domain-specific knowledge base. This external knowledge can be incorporated into the text representation forming richer connected graph structures that preserve additional information often ignored in common text representation methods, such as the widely used vector space model [6].

This leads to the following research question:

Can graph representations of text, in which relationships among concepts are preserved, improve the performance of text mining applications, when compared to baseline methods? The concept relationships provide additional information to the text representation when compared to standard Bag-of-words representations, in which such relationships are disregarded, as the text is typically treated as a collection of words or phrases extracted from the content. The relationships that are considered in this study are based on human-defined semantic relationships that exist among concepts in a certain domain. These can be incorporated into a text document's representation in the form of links that connect related concepts of the text or as external related concepts not present in the text. The former can be considered as implicit semantic information existing inherently in the text content. The latter can be regarded as external domain knowledge accumulated through experience, or in other words, human expertise available in the form of an ontology that can be incorporated into the available representation. In both cases, adding such information allows mining the structure of the text when represented in graph form.

To answer this question, this study attempts to evaluate the performance of two common text mining applications: Text Categorization (TC) and Concept Extraction (CE), applied on biomedical datasets. The research question can thus be broken down into the following. **RQ1**: Do concept relationships and external related concepts, captured in a graph form, provide a better representation for classifiers to discriminate text content and to make more accurate classification decisions using supervised learning methods? **RQ2**: Can the structural properties of a graph provide additional useful attributes for a text document's feature set to improve the ranking of key concepts present in that document? The precision of a concept extraction application is investigated and the significance of using the graph properties is studied.

The experiments corresponding to those research questions are presented in Chapters 3, 4, 5 and 6, including the methodology, results, and evaluation.

The methods used in this study involve transforming biomedical text documents into graphical representations through mapping text entities into predefined ontology concepts and use the graphs and their features in the aforementioned applications. The study investigates whether and how graph representations and their features improve the accuracy of the learning algorithms and how they capture hidden information that might be ignored in baseline methods. These representations offer a practical and natural conceptualization of the text and thus could be applied more effectively than traditional representations such as the common bag-of-words representation, or term co-occurrence relations that might not be as explicit or specific as semantic relations defined within an ontology.

The process of building the graphs and applying them to classification and information extraction algorithms is explored in detail in the following chapters through different experiments and the analysis of evaluation results.

Graph representations have been gaining a lot of attention due to their structural nature and the way they capture links or relations between entities [7], [8]. Graph modeling borrows similarities from cognitive modeling and how humans perceive objects and relations between them. To illustrate this consider a human expert, with adequate knowledge in biomedical sciences, reading an excerpt of an article selected from a collection of documents about *renal failure*. The expert, without prior knowledge of what the article is about, encounters the concept terms *kidney*, *diabetes*, and *hypertension* in the text. The expert intuitively recognizes that *diabetes* and *hypertension* are common causes of *chronic kidney disease* and predicts that the article's topic is most likely related to *renal failure*, rather than *diabetes* or *hypertension*. Figure 1.1 shows a graph representing a possible mental model of the expert's perception of the topic discussed in the article about *renal failure*. This illustration shows how graphs can be used to represent the text content. The relationships can be extracted from a domain ontology of biomedical knowledge, as described in the following chapters. The nature of the relationships are not explicitly used in this study, resulting in the graphs being undirected. However, the proposed methods emphasize the structural properties of the constructed graphs, and how they can be quantified and used in learning algorithms that can make predictions in classification and information extraction tasks.

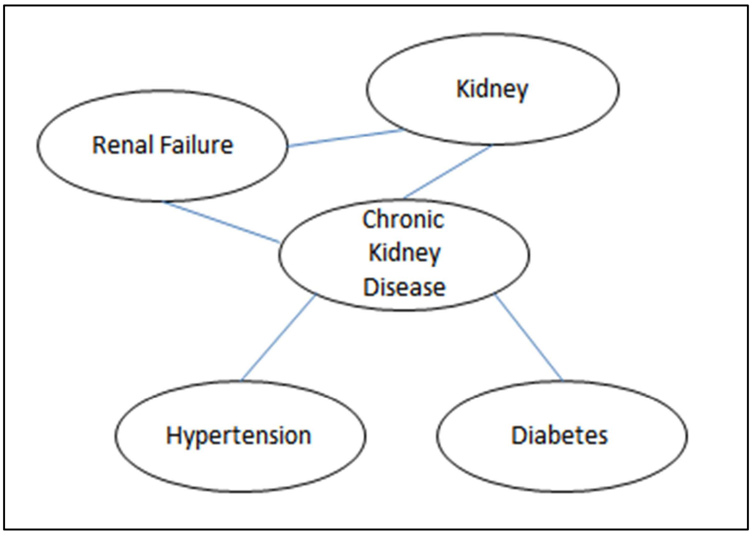


Figure 1.1 Mental image of *renal failure* made by a human reader.

The structural relations hold additional information essential for visualizing and categorizing objects and hence are useful in understanding, learning, and decision making in domain specific tasks. Graphs also have a solid theoretical background in mathematics and computer science where graph analysis and manipulation algorithms have been studied extensively [9]. For these reasons, utilizing graphs is promising and could enhance existing text mining techniques. On the other hand, the wide availability of comprehensive ontologies, specifically biomedical knowledge bases, makes it possible to construct such complex graph structures and explore how their features contribute to the performance of information extraction and other text mining applications.

1.2 Overview and Motivation

Documents have been commonly represented by vectors of words, key phrases, or sentences. Recently, the document structure and entity links within the text have been successfully incorporated to enrich the representation [10]. In particular, graph representations have been gaining a lot of attention lately due to their structural nature and the way they capture links or relationships between entities [11], [12]. Those relationships often hold interesting information that can be mined from the text. In the biomedical domain for example, a gene interaction network can be used to infer certain functions of that gene. Similarly, a semantic network in a certain biomedical text would help finding significant terms or a set of concepts in the text by examining how they are related to other entities. The relationships can be used as similarity measures in structural pattern matching or can be quantified and used as additional feature weights for machine learning algorithms. Although graph mining is being studied and applied widely in different domains, there are numerous areas for theoretical development and empirical studies, especially in machine learning applications. Applications that target biomedical data collections, for example, still cannot match human knowledge and judgment as it requires extensive and specific domain expertise. Improving the performance of those applications is thus challenging as much as it is desirable when applied in the real world as it would thrust further research efforts and application development in text mining, bioinformatics, and other fields of computing sciences such as network security, grid computing, and social networks where graphs can naturally be used in modeling.

Whether applied to molecular structures, social networks, geographic maps, sensor networks, or text documents, graphs offer an intuitive and effective representation model. A graph, in its generic form, is a set of vertices and edges that connect them. A vertex, also called a node, represents a domain specific entity of interest within an application. It can refer to a certain term or concept in a text document, a person within a social network, or a location on a geographical map. Edges are connections or links between the nodes. They represent relations or ties between entity nodes. In a social network, for example, edges could be used to represent a friendship relation between two persons. In the World Wide Web, graph edges could be hyperlinks that connect web pages. In a text document, they can represent semantic relations between the terms in the text.

In particular, graph representations of biomedical text documents, mainly published articles in scientific journals, are constructed and used in the experiments. Biomedical literature is growing rapidly as medical sciences, molecular biology, and genomics evolve. Vast amounts of publications and structural data that have been released are awaiting analysis and further study in hope for breakthrough discoveries. Biomedical concepts in a text document are often contextually and semantically related. Identifying those relationships provides additional knowledge that is useful in understanding the text content, recognizing patterns and interactions among concepts, and making predictions in automation tasks such as classification, summarization, or knowledge discovery. The relationship between the concepts *Kidney* and *Creatinine*, for example, imply that the topic *Renal Failure* is most likely relevant to a certain text's content. Such relations are sometimes explicit and can be identified with the help of predefined ontologies created by experts. In other cases, the relations are implicit or unknown and require more sophisticated tools such as learning algorithms to recognize them. The study starts with describing how text documents can be transformed to graph structures and then investigates how graph-based models affect the performance of text categorization and concept extraction tasks applied on biomedical data.

The main contributions of this work are: 1) providing a graph construction method using ontology mapping. 2) Improving text categorization through the use of, knowledge-based features, graph edge features, and graph kernels. 3) Improving concept extraction using graph features for ranking top key concepts in text documents. Those methods are evaluated and compared to ones that do not use graph structures in addition to popular baseline methods. Essentially, the study investigates how the graph structures capture additional hidden information that is often ignored in baseline methods.

1.3 Organization

This dissertation is organized as follows: Chapter 2 is a literature review of the main concepts and techniques discussed in the study. Chapter 3 focuses on building a knowledge-based graph representation of a text document and applying it in a biomedical text categorization task using a Naïve Bayes classifier. Chapter 4 extends the previous study with an additional experiment, where graph edges are weighted and used as the documents' features. Chapter 5 studies the use of graph kernels in text categorization. Two different kernels are defined to compute similarities between the graphs and used with k-Nearest Neighbor and Support Vector Machine classifiers. Chapter 6 discusses the use of concept graphs and their effect on concept extraction from biomedical text documents. Chapter 7 concludes the studies with an overall summary and highlights of the contributions, limitations, and future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In the following subsections some of the related work that has been done in graph mining and graph representation of text documents is introduced. The techniques discussed range from textual information extraction, knowledge bases, representation of structured data, and applications in text mining. The challenges lie in learning the complex structure of data and in extracting hidden information that is often critical in improving text mining algorithms. The proposed attempts highlight the popularity graph mining has gained in recent years and the broad spectrum of applications where graph mining techniques can be used. The discussion begins with named entity recognition techniques that are used in some experiments in the early stages of concept identification, and then continues to explain how knowledge bases, mainly biomedical sources, can be used in constructing graphs by providing external knowledge in the form of ontologies and controlled vocabularies. Finally, graph representations and similar linked structures and their applications in data mining are reviewed. Other related work on text categorization and concept extraction is reviewed later in the corresponding chapter.

2.2 Named Entity Recognition

Named Entity Recognition (NER) techniques have been used as basic tools for the identification of entities of interest in various domains. NER techniques are based on conditional random fields (CRF) which use a probabilistic model to segment and identify sequence data, including text streams [13]. CRFs can be thought of a graphical representation of sequential data with statistical properties that can be analyzed and used to extract significant entities such as text elements from natural language [14], [15] or more domain-specific entities as those found in biomedical data collections such as gene names and proteins [16], [17]. Two popular biomedical NER tools that are available for public use are ABNER [18] and LingPipe [19]. ABNER is based on conditional random fields and uses regular expressions and neighboring tokens and extracts orthographic and contextual features rather than semantic and syntactic features. ABNER is trained and evaluated on the NLPBA corpus, a modified version of GENIA [20] and the BioCreAtIvE corpus [21]. LingPipe is another software package that also offers a customizable and trainable NER toolkit for general and biomedical entity identification. Named entity recognition can be done simply using dictionary matching and regular expressions or through supervised training of a statistical model. The LingPipe module used in the experiments is trained on the GENIA corpus and can recognize most of the biomedical concept mentions in articles. Named entity recognition can be coupled with concept mapping to a predefined biomedical vocabulary. This ensures a unified representation and usage of biomedical terms that appear in different formats in text documents and across different datasets. In the next section some background on dictionary-based systems and ontologies in text mining is provided.

2.3 Knowledge Bases and Ontologies

2.3.1 Introduction

In biomedical text mining and bioinformatics in general, specific knowledge bases proved to be essential in building large-scale information systems and in improving various information extraction tasks. These include controlled vocabulary databases, thesauri, and ontologies. The term *ontology* will be used throughout this study to refer to a knowledge source used in the experiments since ontologies include a vocabulary database in addition to thesaural or semantic relationships between the predefined concept entities. An ontology is thus a richer knowledge source since it includes additional predefined information, such as hierarchies, categories, and semantic relations. Moreover, the term ontology is commonly used in the biomedical domain, which is where the methods presented are applied.

An ontology can be defined as a formal specification of a shared conceptualization which provides a common understanding of a certain phenomenon or domain [22]. It is used to model knowledge in a certain domain using a representation of common concepts and relations or interactions between them. Ontologies help in different aspects of building information systems. They allow conceptualizing and better understanding of the data at hand as well as incorporating external knowledge into applications. Consequently, they can be used with different methods of data analysis and mining. In biomedical research, the vast number of concepts and technical names used in the literature requires some sort of standardization and integration [23]. As a result, biomedical ontologies have been used extensively in different text mining applications and techniques that target different topics. Some of the prominent works that involve biomedical ontologies include: The Gene Ontology (GO) which contains detailed information on gene and protein roles in cells behavior, molecular functions, and other biological processes [24]; The Molecular Biology Database Collection provides a public repository and a number of online services that allow accessing various molecular biology resources [25]; [26] proposed an integration method to combine GO vocabularies with other external vocabularies to handle the problem of species-specific terms and to provide a better representation of concepts; [27] also described how integrating epitope data into other biomedical knowledge resources would help in organizing the increasingly growing data on immune epitopes; The semantic metadatabase project (SEMEDA) is a semantic integration and federated databases querying system [28]. It is a multi-tiered web application that allows querying integrated databases and provides an ontology-based semantic metadatabase as well as an ontology-based querying interface. The authors describe the integration process and its requirements and evaluate existing relevant ontologies; [29] developed an ontology-driven system for capturing and managing protein family data addressing maintenance and sustainability issues; The Unified Medical Language System (UMLS) [30], will be described in detail in the following section since it is used in the experiments as the main knowledge source of biomedical concepts.

Other ontology-based approaches related to text mining and the proposed methods will be referred to later in the subsequent sections.

2.3.2 UMLS

The Unified Medical Language System (UMLS) is one of the most widely used knowledge sources in the biomedical domain [30], [31]. Made available by the National Library of Medicine, UMLS provides databases, tools, and services for researches and practitioners in health sciences, medical sciences and bioinformatics. The backend of UMLS comprises comprehensive vocabulary databases of biomedical and health-related concepts such as diseases, drug names, anatomical structures, biological functions, and others. The concepts unify the usage of common terms of different formats defined in different sources through unique identifiers. In other words, they can be considered as a higher level representation of the meaning behind the terms, even if they appear in different forms in the literature. In addition, the UMLS database includes a set of predefined relationships among the concepts, allowing the construction of ontologies, concept mapping tools, and graph representations of a set of related biomedical concepts.

The distribution of the UMLS relationship types available in the database version used in this study is shown in Table 2.1. The frequency values change with time as UMLS is updated and new relationships are added. The relationships are of hierarchical and semantic nature and include synonyms, similar, siblings, parent-child, broad, narrow, allowable qualifiers, qualified-by, and unlabeled relationships. The relationships used in the experiments are dependent on the concepts identified in the text and on the specific experiment, as some relationships are excluded to limit the size of graphs. For example, in one experiment, only parent-child relationships and synonyms are considered. In another experiment the relationships used are restricted to those defined in a specific vocabulary source in UMLS. To illustrate the nature of the relationships in UMLS, consider the concept *tomography* as an example. *Tomography* is an imaging technique that produces images of specific sections of the body. Therefore, a parent-child relationship is defined in UMLS between the concepts *imaging* and *tomography*. Similarly, *optical coherence tomography,* a special tomographic technique, is defined as a child concept of *tomography*. It is also worth mentioning that the relationships are neither comprehensive nor accurate in all cases, as they are collected from different sources of vocabularies, and require constant updating. Nevertheless, for the purpose of the experiments, they provide a good source of structural information that is embedded in the text and can be represented by graphs.

In addition, UMLS also includes a table of statistical relations, determined by co-occurrence frequency information. However, those are not used in the experiments. Instead, the available semantic relationships are used for constructing the graphs as they provide a more explicit structural representation of text documents whereby semantics are preserved. The semantic relationships are defined by human experts and thus are a more natural interpretation of the domain knowledge, which includes characteristics, interactions, and classification of the concepts.

Table 2.1 Distribution of UMLS Semantic Relationships

Type of Relationship		**Relative Frequency**
Allowed Qualifier	AQ	1.14
Child	CHD	7.74
Parent	PAR	7.73
Qualified-by	QB	1.14
Broad Concept	RB	2.83
Similar Concept	RL	0.12
Narrow Concept	RN	2.84
Unlabeled	RO	19.32
Possible Synonym	RQ	3.37
Sibling	SIB	43.85
Synonym	SY	9.93

On top of these databases, UMLS also provides higher level tools and services to leverage application development and research in text mining and bioinformatics. Among these are a semantic network of all defined concepts and a set of lexical tools used in natural language processing (NLP) of biomedical text. However, this study relies only on the backend databases to build customized modules that allow accessing the concepts and relationships and constructing graph representations of text as described in detail ahead in the following sections.

The UMLS sources have been extensively used in various research projects and applications in text mining. To start with, some efforts have been made to map biomedical concepts and vocabularies to and from the UMLS [32]–[34], in an attempt to either build ontologies, integrate different knowledge sources, or improve term identification and indexing. UMLS also proved effective when coupled with natural language processing techniques. Several works have been presented addressing the identification and representation of biomedical knowledge in text datasets using NLP methods [35]–[38]. UMLS has also been used in automatic text summarization where terms are mapped to predefined concepts and containing sentences are weighted and extracted to generate summaries [39]–[41]. In information retrieval UMLS has been used in query expansion [42]–[44], translation and cross language retrieval [45], and in search results organization [46]. Other applications include knowledge discovery in medical sciences [47]–[51], question answering [52], topic identification [53], [54], text categorization [55], [56], and keyphrase extraction [36], [57].

2.4 Graph Representations

2.4.1 Introduction

Complex structures and networks can often be represented using graphs where links, relationships, or associations could be used in the mining process. These connections possess additional information that is, in many cases, critical when representing and analyzing the data and later in making predictions or decisions in various mining applications. Graph mining exploits large amounts of structural data that holds implicit and explicit entity relationships or links by looking for interesting patterns or knowledge within the structure [58]. As most of the traditional data mining techniques that target unstructured data are not suitable for graphs, graph mining has emerged as a new research direction within data mining, at the intersection of algorithmic graph theory, link analysis, statistical learning, pattern recognition, information extraction, and other related fields in data mining.

2.4.2 Representation

Different approaches have been used to represent text documents as graphs using different text components and features. The components and features are selected and extracted to capture relevant task-specific information. Text components such as noun phrases, keywords, or sentences often possess inherent structural relations in the form of statistical, syntactic, semantic, or ontological information. The level of explicitness of these relations varies where those based on statistical information are considered the least explicit since they are typically extracted using a collection of documents and not straight from the text whereas predefined ontological relations are considered the most explicit, being defined by domain experts in external knowledge bases.

Within a document collection, term co-occurrence is often used to define associations or relations between terms. In [59], the co-occurrence frequency of any pair of terms in the text is used as the edge weight that connects the respective term nodes. The node weights are calculated using the common term frequency and inverse frequency. The resulting graphs, each comprised of two vectors, a node weights vector and an edge weight vector, are then used to find the similarity between two text documents. The edges can also be derived from co-occurrence within the same sentence as proposed in [60] where the term associations are independent from a certain corpus and thus are domain independent as well. Edge weights can also be described as co-occurrence conditional probabilities that two terms appear sufficiently close to each other. A sliding window can be used to measure the proximity threshold for the terms where edges falling out of the proximity window can be dropped from the graph [61], [62]. Another method that has been used is finding the co-occurrence of symmetric relations in the text using graph edges. A part-of-speech tagger is used and adjacent noun phrases, that appear in a list or are separated by conjunctions, are located and a graph edge is defined to represent the symmetric relation [63].

2.4.3 Graphs and Knowledge Sources

Semantic relations, on the other hand, can be identified with the help of external knowledge sources. Wikipedia has been successfully used to incorporate linked web content as relationship information into graphs. Again, co-occurrence of those links can be used as edge weights. A relation is thus defined between two Wikipedia concepts when there is an internal hyperlink between the concepts from one concept page to the other [64]–[67]. This approach can also be extended to include hierarchical relations by mapping the concepts to Wikipedia categories and used for document classification and categorization tasks [68]–[70]. Similarly, other studies used WordNet [71], an English lexical database, to build graphs where terms are mapped to sets of predefined synonyms, referred to as synsets [72], [73]. The edges represent semantic relations including synonyms, antonyms, hyponyms, meronyms, and troponyms.

In the biomedical domain, Gene Ontology (GO) [24] subgraphs were used to represent documents where directed edges represent hierarchical *is_a* relations (where a concept is a type of or form of another higher level concept) between predefined GO terms. This representation was compared to a flat non-graph representation in a text classification task [74]. Similar GO subgraphs were described in [75] to help in interactive visualization of relations within biological processes. The Systematized Nomenclature of Medicine--Clinical Terms (SNOMED) collection [76] had been used as well to create graphs representing clinical information. The SNOMED collection is hierarchical and has clinical terms grouped into conceptual categories with a linked structure. The relations can be interpreted as *is_a*, *part_of*, *made_of*, and others as described in [77] where SNOMED graphs were used to build a formal conceptualization framework that can be used in relational data modeling or concept mapping into other formal systems.

The UMLS sources were also used extensively in biomedical data representation and mining as discussed in section 2.3.2. One of the early attempts to use UMLS resources to build a graphical ontological structure of medical concepts was described in [35]. The semantic types and relations in UMLS were used in addition to other more general time, space, and value relations to build a custom hierarchical structure to be used in a more specific medical domain. The resulting structure is a concept type lattice that can be used in concept graph formalism and operations on knowledge representation in medical knowledge-based systems. In medical information retrieval systems, natural language queries were transformed to concept graphs using the UMLS semantic network [78]. The graphs are then used to search collections of medical literature. In a similar effort, the thesaural relations and semantic network of UMLS were used to model an end-user's navigation of biomedical concepts into concept graphs that can convey the user's specific interest in a query [79]. In [80], a conceptual model of three levels is proposed. UMLS concepts are linked through an intermediary level of views that represent specific contexts in the medical domain that are identified using a higher level semantic network graph. On top of the resulting concept graph data structure, an object oriented computational model that access existing development tools in bioinformatics is described. This model allows users to translate sentences into graphs that can be used in information retrieval tasks.

2.4.4 Background

An early attempt to formalize the use of graphs by information systems in different domains was introduced by Sowa in 1976 [81], [82]. The formal notation proposed was intended to be used as an intermediary between users and the data. The motivation behind it was to allow the translation of natural language queries or questions asked by humans into graph structures that can be interpreted by computer algorithms, thus providing a flexible database access interface.

Following Sowa's work, concept graphs were described in detail in [83] where a rigorous mathematical description was presented with reference to ontologies, graph properties, and operations which could be applied to knowledge representation in different domains. In addition, [84] tried to formally define the notions of concept graphs and presented a study of different graph operations in terms of logical operations and algorithmic complexity. As for the applied aspects of graph mining, a considerable amount of work has been done in graph matching [85], [86] and finding graph patterns, mainly frequent subgraphs [87], [88]. Patterns of interest or frequent subgraphs in collections of structured data are often desired for different applications in indexing and retrieval [89], [90], web mining [91], bioinformatics [92] and prediction of behavior or interaction in various domains [93], [94].

Link Mining is another closely related topic that was studied extensively [95]. Link Mining explores structural data and linked entities and has emerged from the traditional link analysis research area [96] and has been applied to graph-like structures in different domains. The applications of link mining are numerous as these can be applied to any set of data of interlinked objects. The following are some of these applications. In web information retrieval and link-based ranking, the algorithms PageRank [97] and HITS [98] were proposed. These algorithms rank web search results by importance measures, also referred to as relevance, authority, or connectedness, and are derived from how webpages are linked to each other. In social networking, the centrality of individuals is an important property of individual nodes and is calculated based on the position of those individuals and their links to other individuals [99]. Other proximity measures derived from graph properties are also used to predict links between individuals in social networks [100]. In citation analysis, link prediction could be used to detect possible citation links, predict the nature of those citations, and recommend citations for scientific publications. [101] applied structured logistic regression models to the problem of link prediction in citation graphs.

2.5 Summary

In this chapter an overview of related work in ontologies, graph representations and linked structures and their applications in data mining is presented. The graph mining approaches show how the rich structure of data can be exploited to improve several information extraction techniques. In addition, some of the earlier studies attempt to formalize the use of graphs within a theoretical basis, allowing scalability in different domains and applications.

CHAPTER 3

APPLICATION: NAÏVE BAYES TEXT CATEGORIZATION USING KNOWLEDGE-BASED FEATURES

Graph representations of text offer an intuitive transformation of the text content of a document into a rich set of concepts and semantic relationships that are useful in capturing the underlying topics of that document. In this chapter a method for constructing graph representations of text documents is proposed. Using minimal information extracted from text or from documents' meta-data, the graph representations are constructed and applied in automatic classification of biomedical articles. The method makes use of external domain knowledge and graph features instead of commonly used textual features and attributes. Experimental results of a Naïve Bayes classifier using two graph configurations are reported. In the first configuration, only the graph nodes are used, while in the second, the graph edges are included as well. The method is also compared to a standard baseline classifier that uses a vector space model and occurrence frequency weights. The method could be useful in practical applications where the full content of articles is not available or when access to it is limited.

3.1 Introduction

With the progress of biomedical-related fields, experimental reports and articles are being published extensively and stored in digital repositories. Archives of old scientific articles are also being digitized, indexed, and made available in digital libraries. The problem that arises with the rapid growth of such documents is management and search within large databases. Automatic classification of documents could help alleviate the overhead of maintaining and searching through such large collections.

In this chapter a key-concept graph construction technique is presented. The technique can be used in categorizing biomedical text documents using minimal features extracted from the documents or from their meta-data. The target documents are scientific articles published in different journals of medicine and related biomedical fields. Graphs, representing such articles, are constructed from a small set of key concepts that represent the text content or the topics of the documents. The graphs are generated using minimal information that is either extracted from the text or made available from other sources such as authors. The author-provided keywords, used to label the articles, and the articles' titles are chosen to construct the initial set of key-concepts. Each representation is used separately in a different experiment. Alternatively, one could use other sources such as article abstracts or a small set of keyphrases extracted using a keyphrase extraction tool [102], [103]. The graphs can then be expanded into higher degrees through mapping external domain knowledge. The motivation behind using a small set of concepts is two-fold. The first is reducing the dimensionality of the feature set used in classification that is often very high especially when full-text documents are considered. The second is allowing accurate classification of documents in situations where the content is incomplete or not available. The method is thus independent of the document length, structure, and occurrence frequencies of terms. The key-concept lists, however, are too small to be good representatives of the documents in a classification task and thus need to be expanded into a more 'meaningful' representation. For that reason the initial sets of key concepts are expanded into concept graphs with the help of ontology concepts. On one hand, this representation is an enhanced structure that contains more information, when compared to the initial set of concepts, with a consistent domain knowledge incorporated in the graph, including semantic relationships. On the other hand, the noise, including less relevant terms and concepts that are often present in the text, is eliminated as is it is not included in the representation prior to graph expansion. This technique demonstrates how external knowledge features can replace commonly used text feature attributes such as occurrence frequencies while still achieving a relatively high classification accuracy.

The features that are considered in this study are predefined biomedical concepts available in the form of a controlled vocabulary as well as relationships that might exist among them. The descriptors used in the vocabulary represent specific and general concepts in medicine, biology, and related fields such as: diseases, anatomical structures, pharmacologic substances, biologic functions, and others. The relationships are also predefined and are of semantic nature and include synonyms, parent-child relations, sibling relations, and other narrow or broad relations defined in the ontology. In particular, the initial key-concept list, representing a document, is mapped to concepts defined in the Unified Medical Language System (UMLS) [30], [31]. After the initial set of concepts is mapped, a set of related concepts are retrieved from UMLS to build a more meaningful representation. The resulting document representation is therefore a graph of concept nodes where the edges that connect them represent semantic relations that exist among the concepts. The process is similar to how humans read and perceive the text content through mapping and relating to accumulated knowledge from past experience. With enough domain expertise, it is usually possible that a topic or a higher level category be identified without further reading into the full-text. The details of constructing the graph will be further discussed in section 3.2. Figure 3.1 illustrates the graph construction part.

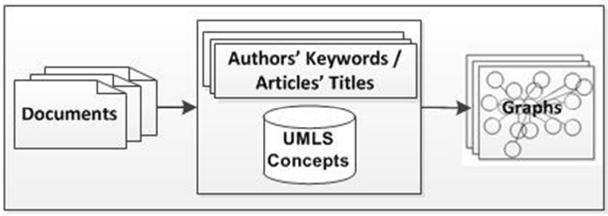


Figure 3.1 From documents into graphs.

The documents used in the experiments are scientific articles published in six medical journals spanning different topics. A set of articles is collected from each journal where the journal category is used as a class label in the categorization process.

As for the classification task, the graphs (both nodes and edges) generated from a training set of documents using different graph setups are indexed to estimate the prior probabilities of the classes and the conditional probabilities of concepts occurring in the target journals. A Naïve Bayes classifier is then used to predict a target class, which in this case, is the journal that an article is most likely selected from.

In the experiments section the performance results of two different graph representations are reported. In the first configuration, the graph nodes from the constructed graphs are used as the feature set of the classifier. In the second configuration both nodes and edges are used, in an attempt to consider the semantic information embedded in the text in the classification process. The proposed method is also compared to a standard Naïve Bayes classifier that uses a vector space model to represent documents and *TF-IDF* weighting of the terms in the text.

The proposed technique could be useful in practical text categorization and indexing applications where minimal information about the dataset is known or available. It could also be useful when the text contains a lot of noise and the target categories are of higher abstraction level, since the graph representation would be a filtered projected view of the text into a common and more-specific domain representation.

3.2 Document Representation

In this section, common document representation techniques used in text mining are discussed and the process of constructing the proposed graph representations of text documents is described. The graph construction process starts with an initial small set of concepts that is expanded into a rich graph with additional semantic information. The discussion also explains the motivation behind using such representations for classification tasks.

3.2.1 Background

In text categorization and information retrieval tasks, documents are commonly represented as vectors of term or keyphrase weights, which is referred to as the vector space model [6]. The weights are considered indicators of how strong the terms or keyphrases represent the document. The most common weighting scheme is *TF-IDF* [104] which is based on the term frequencies – that is how many times a term occurs in a document - and the inverse document frequencies – that is the number of documents in which a certain term appears throughout the whole dataset. This representation is also referred to as the bag-of-words model since each document is transformed into a collection of terms or words, without taking the order in which they appear in the text or the existence of semantic or other relationships between the words into consideration.

Other similar approaches extend this representation and use n-grams features to represent combinations of characters [105] or words [106] of a text's content and apply it in classification techniques. The vector space model weighing scheme was also used to represent sentences in a document, as described in [107], where documents are decomposed into sentences and each sentence was represented as a weighted vector of term frequencies and applied in a text summarization application.

Other efforts have also been made to utilize the structure and semantics of the text and incorporate them into the representation to enhance the used techniques. For example, [108] incorporated the semantic structure at both sentence and document levels. Their models combined statistical features and a conceptual ontological graph representation that represents the sentence structure while maintaining the sentence semantics in the original document. [109] transformed documents into a space of conceptual feature structures using an ontology and lexical resources for a higher level representation and applied it in content-based search. [110] designed a lexical chain that holds a set of semantically related words of a document and used it to represent the semantic content of a portion of the document. [111] presented a keywords extraction algorithm that treats each document as a semantic network that holds both syntactic and statistical information. A semantic network model was developed in which each term is represented by a node and a relation between two terms by an edge. Additional in-depth description of the use of the vector space model and semantics in capturing meaning of the text as well as their applications can be found in [112].

Graph structures have also been used to represent documents as they preserve the structure embedded in the content and allow using graph techniques that have a strong algorithmic and mathematical foundation in discrete math and computer science. For instance, [113] propose a graph representation for document summarization tasks. They use a thesaurus and association rules to connect key phrases in the text. [114] also use graphs to represent documents for summarization. They use graphs to capture word-word, word-sentence, and sentence-sentence relationships in the text. They then compute word and sentence saliency scores to rank their results. Similarly, ontology-based mapping of text into concept graphs have been used in text categorization [115] and concept extraction [102] applied on biomedical datasets where the graph features are incorporated into the representation.

Term or keyphrase statistics, such as occurrence frequencies extracted from the text, are usually essential for learning and classification and have been successfully used in text categorization and other text mining applications. However, in this experiment, the problem where such information is not available is addressed. This could perhaps be due to the absence or limited availability of the full-text content, or when the documents are very large and using an alternative reduced representation would be desired. The method also highlights how domain knowledge can be incorporated into the representation and applied in text categorization. In the following section the method of representing a text document, starting with a few available key concepts that characterize the document, is described. Later in the experiments section this method is compared to a baseline representation that uses the standard *TF-IDF* weight vector of document terms.

3.2.2 Key-Concept Graphs

In the following, key-concept graphs, which are sets of nodes and edges representing the text documents, are described. The representation is initialized using a small set of concept nodes extracted from a document's meta-data. External concept nodes with the corresponding relationships (edges) are then added to enrich the representation.

3.2.2.1 Alternative Representation. The proposed representation is constructed using a small set of document features and expanded into a richer representation using domain concepts and semantic relations. In this representation statistical information obtained from the text is not considered. This makes the proposed method less dependent on a document's content. In addition, using external domain knowledge, the representation is projected into a more domain-specific feature space. Starting with a small set of keywords representing a document and mapping those into predefined concepts and relations, each document is represented by a graph, where nodes represent concepts that might or might not appear in the text, and edges represent semantic relations that exist among the concepts in a certain domain.

In a real world scenario, a human reader with sufficient domain expertise is capable of identifying a high level category of an article by reading the title or a small number of keywords (labels) assigned to that document. Based on this intuition, the process of transforming a text document using such information into a higher level representation, appropriate for processing and classification, is described in the following sections.

3.2.2.2 Initial Setup. The dataset used in the experiments is a collection of articles collected from medical journals. In addition, UMLS [30], [31] is used as an external knowledge base of biomedical concepts. UMLS provides a comprehensive vocabulary database and ontology of biomedical concepts and relationships among them.

For each article in the dataset, the author-provided keywords are extracted. Those are typically the labels that authors assign to their articles upon publication. In addition, the titles of the articles are extracted, and used in a different experiment.

The author-provided keyword list and the noun phrases in the title serve as the initial representation of each document. Those are then mapped into predefined UMLS concepts and referred to as key concepts. In the mapping process, both a first-best (*fb*) match and an n-gram (*ng*) match are attempted to map a keyphrase into UMLS concepts. For instance, if the phrase ‘Atypical antipsychotic drugs’ is found in the author-provided keyword list (or extracted from the title), it would be mapped to the concept ‘Antipsychotic Drugs’ using first-best matching since ‘Antipsychotic Drugs’ is the first successful match with a maximum length (number of terms), even though the whole initial phrase containing that concept term does not exist in UMLS. Using n-gram mapping, it would be mapped to all combinations of concepts that correspond to the terms in the phrase and exist in UMLS, in this case: ‘Antipsychotic Drugs’, ‘Antipsychotic’, and ‘Drugs’.

Combinations of the concept mapping modes and the usage of author keywords vs. titles are used to generate different instances of the graphs and are evaluated separately in different experiments.

3.2.2.3. Concept Relationships. After the author keywords or titles are mapped into unique UMLS concepts, the obtained list is used as the base nodes list of a key-concept graph. The graph is then expanded by adding related concepts queried from UMLS.

Relationships are available as pairs of related concepts and semantic relationships between them. Examples of related concepts in UMLS are: *'Anxiety – Mental Disorders'* and '*Pathologic Process – Psychological Stress'*. The semantic relationships are typically *synonym, parent-child*, *sibling, broad*, and *narrow* relationships. The related concepts are added to the graph as new nodes, where the relationships are represented by edges. Upon adding new nodes, if a concept is related to an existing concept in the graph, an edge is also added to link them together. This process is also parameterized, as the number of levels of related concepts to be added to the graph, is also variable. In the experiments graphs with up to two levels of related concepts are constructed. When two levels are considered, concepts related to the related concepts are also included in the representation. This is meant to increase the degree of the graph representation by adding more domain knowledge that could be more discriminative with respect to a document's class. Adding more levels of related nodes however, would increase the degrees of graphs exponentially and could add some noisy and irrelevant concepts to the representation. Figure 3.2 shows an example of concept nodes and the relationship edges that connect them. Figure 3.3 shows an example of the resulting graphs representing a document taken from a journal of *psychology*.

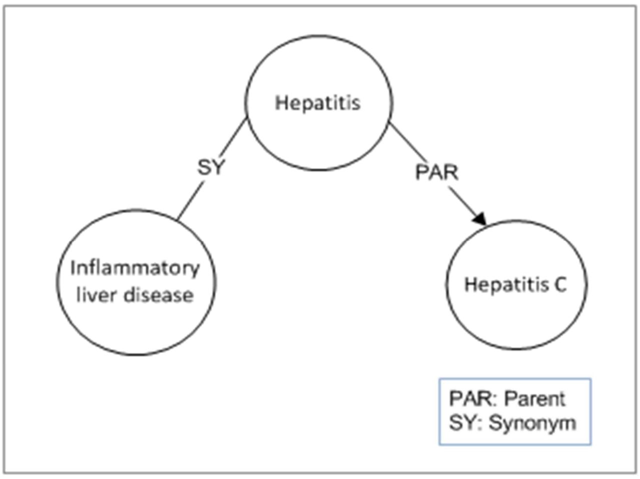


Figure 3.2 Concept nodes and relationships.

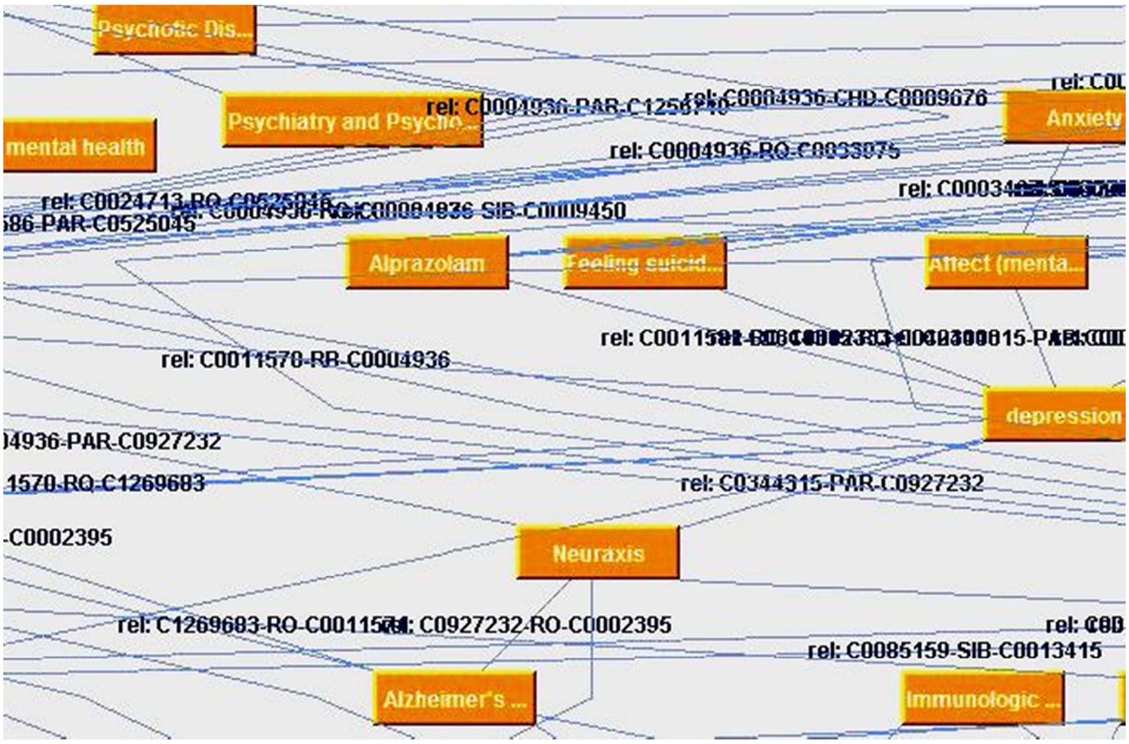

Psychotic Dis...
Psychiatry and Psycho...
mental health
Anxiety
rel: C0004936-PAR-C1256740
rel: C0004936-CHD-C0009676
rel: C0004936-RO-C0033975
rel: C0024713-RO-C0525045
586-PAR-C0525045
rel: C0004936-SIB-C0009450
Alprazolam
Feeling suicid...
Affect (menta...
rel: C0011570-RB-C0004936
depression
)4936-PAR-C0927232
1570-RO-C1269683
rel: C0344315-PAR-C0927232
-C0002395
Neuraxis
rel: C1269683-RO-C0011570
rel: C0927232-RO-C0002395
rel: C0085159-SIB-C0013415
Alzheimer's ...
Immunologic ...

Figure 3.3 A sample graph representation.

3.3 Text Categorization

In this section the Naïve Bayes classifier, on which the proposed method is based, is described. Although the classification procedure presented here is applied on biomedical articles, the techniques, can be easily applied to other domains, including general domains where an ontology can be built from available information sources such as Wikipedia and WordNet [116]–[118]. Furthermore, other classifiers can also be used instead of the Naïve Bayes as long as they can be adapted to the graph representation. For example, a k-NN classifier could be applied on the same representation but would require defining a graph similarity measure, perhaps by using graph kernel functions as described in chapter 5.

3.3.1 Background

Text categorization is the automated process of sorting documents into classes or groups based on their content. Text categorization has attracted significant research interest in information science and machine learning [119]. The applications of text categorization include indexing and classifying of scientific publications, email filtering, literature based discovery, and finding relationships among biomedical entities. The success of a text categorization application is based on the efficiency and accuracy of the underlying information retrieval and machine learning techniques used.

Several text categorization techniques have been proposed to automate the manual process of organizing and searching documents. The following techniques have been successfully used to classify documents based on content similarity. The Naïve Bayesian probabilistic approach was suggested for automatic indexing of documents and is shown to be straightforward but surprisingly efficient in terms of classification [120]. It is assumed that the extracted feature words are independent and therefore Bayes' theorem can be used in the classification algorithm. The k-Nearest-Neighbor (k-NN) technique has also been used in text categorization [121] and is popular due to its simplicity, nonlinearity, and ability to handle multi-class objects. Support Vector Machines (SVM) are shown to be very suitable for categorizing documents and perform very well even with large feature spaces [122]. Decision trees and decision rules offer an intuitive symbolic way to model the classification procedure which is usually based on logical decisions and predictions and perform fast compared to other learning methods [123]. As for biomedical literature and digital libraries, text categorization has been widely used to sort and manage medical and health records. [124] designed a classification system based on inductive decision trees that can handle different types of medical records. [125] showed that using phrases instead of words significantly improves the accuracy of medical text retrieval. [126] showed that using additional knowledge sources improves the classifier performance by adding useful information to the feature vector.

Using textual features for categorization is not the only approach to classifying documents. Complex structures such as documents can be represented as graphs where nodes represent textual or other document features, and edges represent relationships between those features. The addition of relationship edges to describe documents can create a much higher-dimensional feature space, thus allowing for more nuanced and potentially useful embedded knowledge of the documents. Graph matching techniques have been commonly used to categorize graph-represented documents. [10] proposed a web document classification technique based on k-NN. In [108], conceptual ontological graphs were used to represent documents based on sentence structure and on a concept statistical analyzer. The graphs are then used to construct normalized feature vectors for text categorization. Graph classification is a major application in machine learning where graphs, representing objects, need to be categorized based on the entities they represent and the relationships between them. Supervised learning is usually applied on graphs where the similarity between graphs is calculated using kernel functions. [70] used a semantic kernel that incorporates Wikipedia background knowledge to enrich the document representation. They achieved improved accuracy in document classification when compared to traditional bag-of-words representation. In [127], three different datasets were used for classification experiments each having its own representation of relationships between node objects in a graph. Co-authors were used to link scientific publications, actors to link movies, and page hyperlinks to link Wikipedia documents. Weighted frequent subgraphs were used in [128] to construct effective feature vectors for classification and to overcome the computation overhead that is associated with graph structures. [129] uses exact and inexact graph matching as well as substructure pruning and ranking to optimize classification and compare their result to a Naïve Bayesian classifier. [130] attempts to exploit the linguistic syntactic and semantic characteristics of phrases in text. They encode phrases as graphs and use a substructure and pattern discovery algorithm for classification. Classification of graphs has other broad applications in bioinformatics and chemical informatics where protein sequences and molecular structures need to be classified according to structure [109].

3.3.2 The Naïve Bayes Classifier

The Naïve Bayes (NB) classifier is a simple probabilistic classifier based on the Bayes theorem. It has been widely used in classifying text documents in different domains and is known to perform well despite the fundamental naïve assumption that the document features used in the model are independent [119], [120], [131], [132]. The NB classifier essentially estimates the probability $\hat{P}(c|d)$ of a certain class given a document:

$$\hat{P}(c|d) = \frac{\hat{P}(c) \times \hat{P}(d|c)}{\hat{P}(d)}, \tag{3.1}$$

where $\hat{P}(c)$ is the estimated prior probability of a class *c*, that is the probability of a document being in class c when the document features are not considered in the computation; *d* is a document in the dataset *D* $(d\epsilon D)$ represented by its feature weights, which is referred to as $\boldsymbol{x}$. $\hat{P}(d)$ is constant as it does not depend on the class and thus the denominator can be dropped from the calculation.

Assuming a document is represented by a feature vector $\boldsymbol{x}$, $\hat{P}(d|c)$, the likelihood that a document *d* with features ***x*** belongs to a class *c*, can be calculated as such:

$$\hat{P}(d|c) = \prod_{j} \hat{P}(x_j|\, c)\,, \tag{3.2}$$

where $\widehat{P}(x|c)$, will be estimated according to the features used in each document representation as described in the next section.

The maximum a posteriori class ${c_{MAP}}_{c \in C}$, that is the class a document most likely belongs to can therefore be calculated as such:

$$c_{MAP} = argmax_{c\in C}\, \hat{P}(c) \prod_{j} \hat{P}(x_j|\, c) \tag{3.3}$$

3.4 Experiments

In this section the experimental setup, including the dataset and the different configurations and features used in representing the documents, is discussed. The classification results applied on each representation are then reported.

3.4.1 The Dataset

The dataset used in the experiments is comprised of 563 text documents. The documents are published articles collected from six journals spanning different topics in medicine. The journal categories are used as the target classes to be predicted for each article. The different journal categories that the articles were selected from are shown below in Table 3.1.

Table 3.1 Journal Categories of the Selected Articles

Class	**Journal Category**
P	Psychiatry
G	Gastroenterology
N	Neurology
M	Molecular Immunology
O	Ophthalmology
R	Respiratory Diseases

For each article, the titles, the author-provided keywords, and the full text are extracted. The full text is only used by a standard baseline classifier for evaluation and comparison purposes.

3.4.2 Graph Configurations

As pointed out in Section 3.1, the graphs representing the articles in the dataset are constructed using different parameters. In the following the process of how each representation is initialized and expanded using external knowledge-based concepts is explained.

3.4.2.1 Concept Features. As the full-text features are not considered in the proposed method, the graphs are initialized using either the author-provided keywords or keywords extracted from the articles' titles, as the base representation of documents. The length of each keyword list is variable across the dataset and not all keywords are guaranteed to have a match in UMLS, which is one limitation of the method. However, most of the keywords can be matched either exactly or partially (first-best or n-gram matching). The parameters of the graph expansion process are: the concept mapping mode and the level of related concepts added to the representation.

For each configuration, the concept nodes of the resulting graphs generated from the training dataset are indexed with respect to each class (journal category). Those concepts are used as the document features where the feature vector $\boldsymbol{x}$ is a vector of relative occurrence frequencies in a certain class. $\hat{P}(x|c)$ is thus estimated as the relative frequency of concept *j* indexed under class *c*:

$$\hat{P}(x|c) = \frac{N_{xc}}{N_x}, \tag{3.4}$$

where N_{xc} is the number of times a concept *x* is indexed under class *c* and N_x is the total number of occurrences of concept *x* in the whole dataset.

The prior probabilities $\hat{P}(c)$ are also estimated for each journal category as the relative frequency of documents indexed under the corresponding category (the number of documents indexed under class c divided by the total number of documents in the dataset). A constant $\alpha = 1$ is added to the relative frequencies of concepts to avoid zero probabilities resulting from the absence of certain concepts indexed under a certain class. A document's class can then be predicted using the following equation:

$$c_{MAP}(d) = argmax_{c\in C}\,\hat{P}(c)\prod_{j}(\alpha + \hat{P}(x_j|\,c)) \tag{3.5}$$

3.4.2.2 Relationship Features. The graph edges are also used in calculating the class likelihood values using the concept relationships features. The graph edges that represent the relationships among the concepts are also indexed in a similar fashion, allowing the calculation of their frequencies with respect to different classes. When those edges are used as features the vector ***r*** is used instead of ***x***. ***r*** is the features weight vector of the concept relationships which is calculated using the relative frequencies of both the edges and their corresponding connected nodes. In this setup $\hat{P}(r|c)$ is estimated for each edge in the graph as follows:

$$\hat{P}(r|c) = \frac{N_{rc}}{N_r} \times\, max\left(\frac{N_{x_1c}}{N_{x_1}}, \frac{N_{x_2c}}{N_{x_2}}\right) \tag{3.6}$$

where N_{rc} is the number of times a relationship *r* is indexed under *c* and N_r is the total number of times *r* is indexed in the dataset. x_1 and x_2 are the concept nodes connected by the edge corresponding to *r*. $\hat{P}(r|c)$ is then similarly used to find the maximum a posteriori classes:

$$c_{MAP} = argmax_{c\in C}\,\hat{P}(c)\prod_{j}(\alpha + \hat{P}(r_j|\,c)) \tag{3.7}$$

3.4.3 Results

After running a set of pilot experiments, the configuration resulting from using a combination of n-gram mapping (*ng*) and adding two levels of related concepts (*r2*) to the graph achieved the best performance. This configuration is used for constructing the graphs from both the author-provided keywords (*ap*) and from the titles (*tt*) of the articles. The results corresponding to using nodes only compared to nodes and edges (*ed*) are also reported. The results for the different combinations are shown in Table 3.3 (represented by *ap-ng-r2, ap-ng-r2-ed, tt-ng-r2-ed*).

Table 3.2 Classification Performance

Exp	**Configuration**	**Precision**	**Recall**	**F_1 Score**
A	*ap-ng-r2*	0.865	0.844	0.854
B	***ap-ng-r2-ed***	**0.878**	**0.868**	**0.873**
C	*tt-ng-r2-ed*	0.753	0.715	0.733
D	*NB + TF-IDF*	0.847	0.860	0.853

A 10-fold cross-validation on the 563 documents is performed, applying the NB classifier described in Section 3.3 using each graph configuration at a time. Experiments A and B correspond to the representations constructed from the author-provided keyword lists. Experiments B and C use the relationship feature weights calculated from the graph edge information. Experiment C corresponds to the representation constructed from keywords extracted from the articles' titles only, as opposed to the author- provided keywords used in experiments A and B.

Experiment D is a standard NB classification based on representing a document as a bag-of-words and using *TF-IDF* weighting of the terms. The results of this experiment are also obtained through 10-fold cross-validation. In this experiment the full-text content of each document is used to generate the term weight vector. This classifier provides a baseline performance comparison to the proposed method and highlights how the full text features and their occurrence weights can be substituted with external domain concepts and their relationships.

The performance results in Table 3.2 are reported in terms of micro-averaged precision, recall, and the corresponding F_1 scores. Precision is the proportion of documents predicted in a certain class that actually belong to that class. Precision is defined as $TP / (TP+FP)$. Recall is the proportion of documents that belong to a certain class and were predicted so. It is defined as $TP / (TP+FN)$. The F_1 score is a combined measure defined as $(2 \times precision \times recall)/(precision + recall)$. TP is the number of true positives, *TN* is the number of true negatives, *FN* is the number of false negatives, and FP is the number of false positives.

3.4.4 Discussion

In general, the experiments demonstrated good classification performance, despite the small number of key concepts that were used to construct the initial corresponding representations. Achieving such a relatively high classification performance, while ignoring the explicit full-text information in the model, is promising and underscores the significance of knowledge-based representations in learning.

In the pilot studies, using n-gram mapping of keywords showed significant improvement over first-best mapping of keywords, when only one level of related concepts is added to the graph. However, this was not the case when two levels of related concepts were added into the initial key-concepts list, since the number of concepts added to each representation was already increased significantly and that compensated for the low dimensionality of the initial representations resulting from first-best mapping. In other words, adding more n-grams to an expanded representation did not provide further discriminative information to the classifier.

Both representations in experiments A and B yielded better results than the *TF-IDF* representation of experiment D. In experiment B, incorporating the graph edges information shows around 2% improvement in performance over using the concept nodes alone, which supports the assumption that the semantic relations provide more information to the classifier. This information's contribution, however, might be constrained by the fact that the semantic information was implicitly included by adding the related concept nodes (in experiment A), even when the edges were not used explicitly. Both forms of additional information used in A and B can be considered semantic information, the former being implicit, while the latter explicit, determined by the corresponding edges. The use of edges and their corresponding weights is studied in more detail in the next chapter.

As for experiment C, using the keywords extracted from articles' titles to initialize the document representation achieved an expected lower performance. This is due to the fact that the titles contain only a small number of relevant terms. The title terms can also be ambiguous or sometimes misleading, even for human readers, as they often include inconsistent terminology and references. However, achieving an F_1 score of 0.733 is reasonable and shows that the method could be useful when titles are the only available information, which is a common scenario in some digital libraries and archive databases.

Overall, having the ability to incorporate external domain-knowledge is desirable in text categorization tasks, as it allows compensating for the lack of enough information about the topics embedded in the text, which often include high level concepts and semantic relationships within a certain domain.

Although the comparison might not seem fair at all levels, the experiments show how the full-text features can be 'guessed' and projected into the proposed knowledge-based representations, which give a good conceptualization of the underlying topics in the text documents, without using common statistics such as occurrence frequencies of terms within the text.

One limitation of the described method is the process of concept mapping from keywords extracted from the text to concepts defined in the external knowledge source. On one hand, matching terms with predefined concept descriptors is not always accurate, due to the inexact matching involved which introduces a precision/recall tradeoff. Another problem is the fact that some concepts have more than one meaning and could be incorrectly matched, unless advanced word disambiguation techniques are used. On the other hand, the predefined vocabulary sources are neither complete nor they are accurately defined, especially in terms of semantic relationships. Such knowledge sources require constant updating and refining, maintaining a certain level of knowledge 'quality', as new domain-specific concepts emerge in the literature.

Another issue that should be noted here is the intrinsic subjectivity in the authors' choice of keywords and titles. As a result, the performance of the proposed method, being dependent on such information in constructing the document representation, could be affected. Indeed, in the absence of full-text content, finding alternative keywords less susceptible to this subjectivity could be challenging.

3.5 Conclusion

In this chapter an alternative knowledge-based representation for text documents is presented and applied in classifying biomedical articles. The representations are initialized using a few concepts extracted from the articles' meta-data (author keywords or titles) and expanded into a graph structure that holds more domain information in terms of concepts and semantic relationships. A Naïve Bayes classifier is then applied on the resulting graphs and the journal categories (classes), where the articles were selected from, are predicted.

The results show how the commonly used textual statistics can be replaced by domain concepts and relationship features, while still achieving high classification accuracy. The proposed method also outperforms a standard baseline NB classifier that uses the common vector representation of the text.

In practice, the method could be useful in categorizing and indexing documents where the full-text content is not available or incomplete. A small list of available keywords can be expanded into a rich domain-specific representation using an external domain knowledge source. This method could also help in reducing the dimensionality of the documents' feature space as well as filtering irrelevant terms from the text, particularly in situations where the target documents are very large and classification is computationally expensive.

In reference to research question RQ1 stated in Chapter 1, the graphs used in the experiments show how additional information can be incorporated into the representation at hand. The richer representation provides a better ‘understanding’ of the text by incorporating concept relationships. This information is useful in the process of classifying documents, as it adds more discriminative and shared features within a topic in the dataset, even when the full-text information is not included in the representation.

3.6 Summary

The experiment presented in this chapter demonstrates how a Naïve Bayes classifier can be applied to a dataset of biomedical documents without using the original features that exist within the text content. The method shows how higher-level graph representations can be built using few key concepts and an external domain knowledge base. The proposed technique is compared to a standard classifier that uses the full-text content and term statistics calculated from the given dataset.

CHAPTER 4

APPLICATION: TEXT CATEGORIZATION USING WEIGHTED EDGE FEATURES OF GRAPHS

In this chapter an extension to the previous classification applications is presented. The proposed method also attempts to explore how the graph structural features can be quantified and used in a practical vector-based representation for text categorization. The results show great improvement in performance compared to the common *TF-IDF* representation.

4.1 Introduction

Motivated by the representation and experiment discussed in the previous chapter, this chapter presents another method of document representation, where concept relationships, extracted from the target dataset, are weighted and used as features in a vector-based representation. Compared to single terms, phrases, or even selected concepts, existing concept relationships indicate the presence of embedded semantic information that might express more meaning than any of the related concepts considered independently. For example, an article that contains two related concepts such as *brain* and *cognitive process* is more likely to have been selected from an article about *psychology* than from one about *brain cancer*. The relation between *brain* and *cognitive process* can be easily identified by a human expert or an external source of domain knowledge as explained in the next section.

The proposed method involves identifying a number of commonly occurring relationships in a dataset. Those semantic relationships are expressed using graph edges as described in the previous chapter, where each graph represents a text document. The text content of biomedical articles is used to construct the graphs, where the edges are assigned weights and used as features in classification. Feature weights can be calculated using statistical and structural information extracted from the related nodes in a graph. In particular, the weights are calculated from the corresponding nodes' occurrence frequencies, their inverse document frequencies, their connectivity value in a graph, and the size of their containing clusters. These weight components are aggregated to form a single value that is assigned to edges existing in a graph. A Naïve Bayes classifier is then applied to the set of graphs, where each graph is represented by its edges feature vector. Although the representation used here is based on the vector space model described earlier, the selection of features and their weights embed implicit and explicit semantic and structural information that exist in the documents.

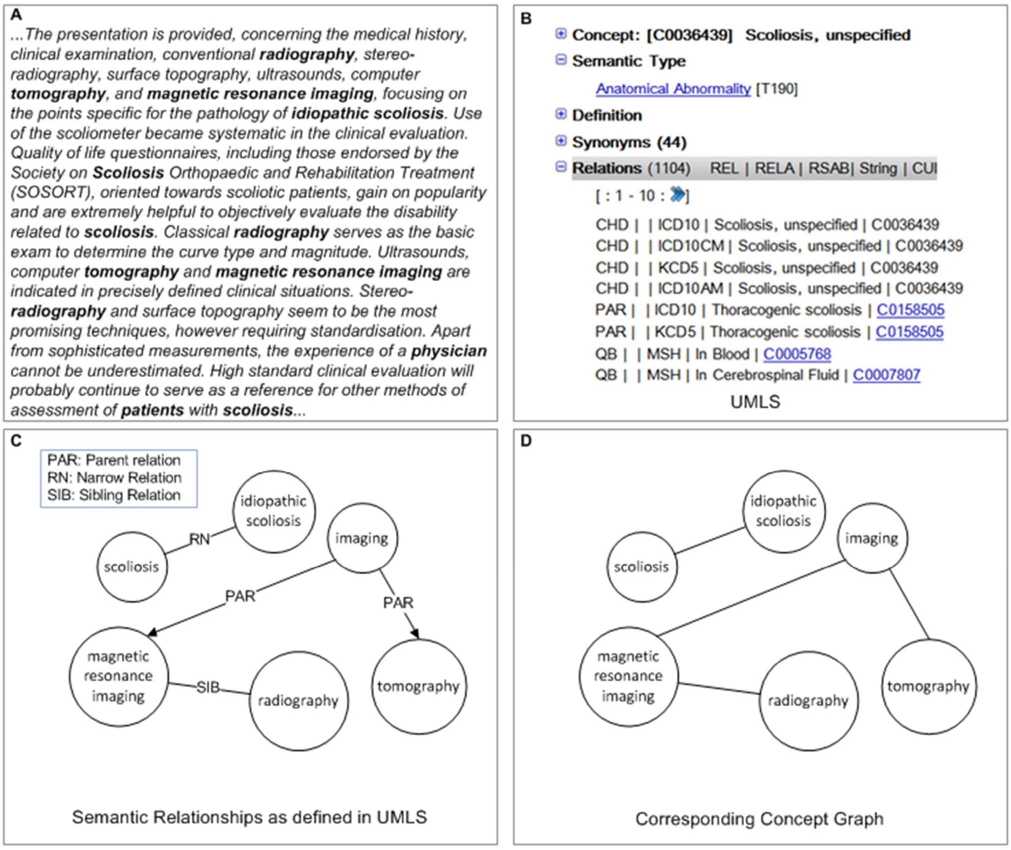
The classifier used in this experiment is compared to a standard Naïve Bayes classifier that uses the *TF-IDF* scheme to validate that using the graph edges information improves the classification performance. The two classifiers used are identical in terms of learning and predicting. However, the choice of features and the weighting schemes are the main point of comparison and argument of this experiment.

4.2 The Approach

The method presented in this chapter consists of two major components. The first is the graph construction part which involves mapping biomedical terms that are extracted from the text into predefined concepts of a controlled vocabulary. In addition, the relationships among the concepts are also identified and added to the representation. The second component is the application of a Naïve Bayes classifier to the documents represented by their weighted edges.

4.2.1 Graph Construction

Transforming a text document into a graph follows a similar procedure as in the previous chapter. However, in this method, the graphs are constructed from the original full-text documents. The first step involves identifying all noun phrases of the text, from which biomedical concepts can be extracted. All noun phrases are initially considered in the concept mapping phase. This is intended to increase recall by attempting to match any noun phrase to a UMLS concept. Although this method results in a more computationally expensive procedure and more non-biomedical concepts being included in the representation, it ensures that no concepts are missed in the mapping phase. Thus, some less relevant concept nodes are eventually added to the graph, as UMLS includes many non-biomedical concepts that often appear in the literature. However this does not affect the representation since non-relevant concept nodes are given less weights or dropped from the feature set as described in the next section. Figure 4.1 shows a sample text and the corresponding concept graph with the extracted nodes and edges. It is worth noting here that the specific types of relationships between concepts are not explicitly used. An edge is added to the graph whenever the corresponding concepts are related, regardless of what type of relationship exists between them.



A

...The presentation is provided, concerning the medical history, clinical examination, conventional ***radiography****, stereo-radiography, surface topography, ultrasounds, computer* ***tomography****, and* ***magnetic resonance imaging****, focusing on the points specific for the pathology of* ***idiopathic scoliosis****. Use of the scoliometer became systematic in the clinical evaluation. Quality of life questionnaires, including those endorsed by the Society on* ***Scoliosis*** *Orthopaedic and Rehabilitation Treatment (SOSORT), oriented towards scoliotic patients, gain on popularity and are extremely helpful to objectively evaluate the disability related to* ***scoliosis****. Classical* ***radiography*** *serves as the basic exam to determine the curve type and magnitude. Ultrasounds, computer* ***tomography*** *and* ***magnetic resonance imaging*** *are indicated in precisely defined clinical situations. Stereo-* ***radiography*** *and surface topography seem to be the most promising techniques, however requiring standardisation. Apart from sophisticated measurements, the experience of a* ***physician*** *cannot be underestimated. High standard clinical evaluation will probably continue to serve as a reference for other methods of assessment of* ***patients*** *with* ***scoliosis****...*

B

⊞ **Concept: [C0036439] Scoliosis, unspecified**

⊟ **Semantic Type**

Anatomical Abnormality [T190]

⊞ **Definition**

⊞ **Synonyms (44)**

⊟ **Relations** (1104) REL | RELA | RSAB| String | CUI

[: 1 - 10 : »]

CHD | | ICD10 | Scoliosis, unspecified | C0036439
CHD | | ICD10CM | Scoliosis, unspecified | C0036439
CHD | | KCD5 | Scoliosis, unspecified | C0036439
CHD | | ICD10AM | Scoliosis, unspecified | C0036439
PAR | | ICD10 | Thoracogenic scoliosis | C0158505
PAR | | KCD5 | Thoracogenic scoliosis | C0158505
QB | | MSH | In Blood | C0005768
QB | | MSH | In Cerebrospinal Fluid | C0007807

UMLS

Figure 4.1 Sample text and corresponding graph.

A part-of-speech (POS) tagger is used to identify all components of the sentences, from which all combinations of parts of speech that make up noun phrases are extracted. The n-grams of the noun phrases are then looked up in UMLS to check whether they are indexed as biomedical concepts and respectively added as graph nodes if the match is successful. The concept relationships among the concept nodes are also looked up in UMLS, and a corresponding edge is added whenever a relationship exists.

4.2.2 Features and Weights

All nodes in the graphs are consequently assigned four different weight components that correspond to their significance in a document. Below is a description of each.

1. $f_{i,d}$: Concept frequency, which is the number of times a concept term *i* appears in a document d. This value assigns more weight to concept terms with high occurrence frequency in a document.
2. idf_i: Inverse frequency of documents that contain a concept term i. This value ensures that common terms in the whole dataset are given lower weights while rare terms are favored.
3. cw_i: Connectivity weight of a concept node *i* in a graph. This is the calculated as the magnitude of the vector of $f \times idf$values of related nodes $c_1, c_2, \dots, c_j$:

$$cw_i = \sqrt{\sum_{k=1}^{j} \left(f_{j,d} \times idf_j\right)^2} \tag{4.1}$$

   This component assigns higher weight values to concept nodes that are better connected in a graph. Nodes that are connected to more nodes of high $f \times idf$ values would be favored.
4. cs_i: Cluster size, which is the number of nodes of the cluster containing the concept node *i* in a graph. In this experiment clusters are referred to as all connected components of the containing graph. These are the maximally connected subgraphs of the concept graph, which suggest a certain level of coherence of a certain topic. Therefore, a bigger cluster implies that the contained nodes might be more significant than others, in terms of their tight relationships within an underlying topic.

All values are then normalized using min-max normalization, and the product of the weight components is calculated for each concept node *i* in a document *d* as such:

$$nw_{i,d} = f_{i,d} \times idf_i \times cw_i \times cs_i \tag{4.2}$$

The related nodes’ weights are aggregated into a single value and assigned to the corresponding edges. The weight of an edge k is thus calculated as the sum of weights of the nodes *i* and *j* that it connects in a document d:

$$ew_{k,d} = nw_{i,d} + nw_{j,d} \tag{4.3}$$

The number of unique edges extracted from the dataset was initially around 60,000. To reduce the dimensionality of the feature space, edges having weights below a certain threshold were dropped from the feature set. Although the threshold used was very low, the number of unique edges was drastically reduced to around 10% of the original number, as most of the extracted edges are not significant and not representative of the documents. The resulting number of edge features used was 5802. The distribution of the original set of edge weights, shown in Figure 4.2, had a mean edge weight of 0.113 and a median of 0.073. All edges having a weight less than 0.1 were dropped from the dataset. In an additional classification experiment, non-weighted features were also used for comparison. In that case the values of edge features existing in a document were set to 1 and those of the non-existing edges to 0.

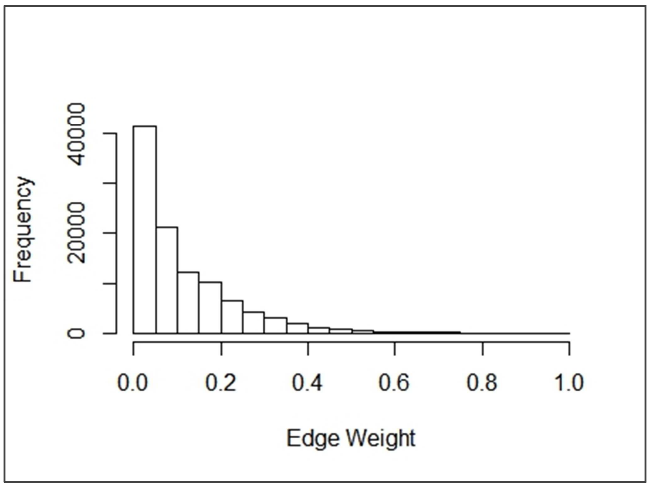


Figure 4.2 Edge weight distribution of original feature set.

4.2.3 Classification

To classify the documents, a standard Naïve Bayes classifier is used [133], [134]. As described in the previous chapter, the classifier estimates the probability of a certain document *d* belonging to a certain class *c*. Using Bayes Theorem that probability can be written as such:

$$\hat{P}(c|d) = \frac{\hat{P}(c) \times \hat{P}(d|c)}{\hat{P}(d)} \tag{4.4}$$

Since a document d is represented by its features, in this case the edges weight vector e, and since the Naïve Bayes classifier assumes that the features are independent, the likelihood $\hat{P}(d|c)$, can be written as such:

$$\hat{P}(d|c) = \prod_{j} \hat{P}(e_j|\, c)\,, \tag{4.5}$$

The features representing semantic relationships might not be strictly independent in reality due to possible correlations among them. However, the ‘naïve’ assumption of the NB classifier allows estimating the probability $\hat{P}(d|c)$ using the product of the edge probabilities regardless of the actual dependencies that might exist.

Each document d is represented by its edges vector e with weight values ew as shown in Table 4.1 below.

Table 4.1 Feature Vectors of the Documents

	$\boldsymbol{e_1}$	$\boldsymbol{e_2}$	…	$\boldsymbol{e_n}$
$\boldsymbol{d_1}$	$ew_{1,1}$	$ew_{2,1}$	…	$ew_{n,1}$
$\boldsymbol{d_2}$	$ew_{1,2}$	$ew_{2,2}$	…	$ew_{n,2}$
…	…	…	…	…
$\boldsymbol{d_m}$	$ew_{1,m}$	$ew_{2,m}$	…	$ew_{n,m}$

$\hat{P}(c)$, the prior probability for a class c can be estimated as the relative frequency of documents of that class. $\hat{P}(d)$ is constant since it does not depend on the class, and thus can be omitted from the calculation.

As for the likelihood of the document features being selected from a certain class c, the classifier assumes that the values of each edge feature e_j are normally distributed within that class with mean μ_{jc} and standard deviation σ_{jc}, and therefore, the corresponding conditional probabilities can be estimated as follows, using the Gaussian probability density function:

$$\hat{P}(e_j|\,c) = \frac{1}{\sqrt{2\pi\sigma_{jc}{}^2}}\ e^{-(ew_j-\mu_{jc})^2/(2\sigma_{jc}{}^2)} \tag{4.6}$$

In the testing phase, the predictions can be made by choosing the class with the highest posterior probability $\hat{P}(c|d)$ for each document, which is the maximum a posteriori (MAP) class. This is equivalent to:

$$c_{MAP}(d) = argmax_{c \in C} \ \hat{P}(c) \prod_{j} \hat{P}(e_j | \, c) \tag{4.7}$$

The same Naïve Bayes classifier is also used as the baseline method for comparison, where the feature values used are the *TF-IDF* values of document terms instead of the edge weight components.

4.3 Experiments

4.3.1 The Dataset

The dataset used is the same as the one described in the previous chapter, comprised of 563 full-text articles selected from 6 journals of medical sciences. The journal categories are: *Psychiatry*, *Gastroenterology*, *Neurology*, *Molecular Immunology*, *Ophthalmology*, and *Respiratory Diseases*. In this experiment, only half of the text content of each document is used to build the corresponding graph, as most of the topics can be inferred from the abstracts and the introductions of the articles. This reduction is meant to eliminate redundancy and to reduce the computational complexity of parsing the text, constructing the graphs, and applying the classifier learning and prediction.

4.3.2 Evaluation

The dataset is divided into ten partitions and a 10-fold cross validation is performed. In each iteration one partition is reserved for testing and the others are used for training the model. The results are evaluated in terms of precision, recall, and F_1 scores. Precision is the proportion of documents predicted in a certain class that actually belong to that class. Precision is defined as *TP / (TP+FP)*. Recall is the proportion of documents that belong to a certain class and were predicted so. It is defined as *TP / (TP+FN)*. The F_1 score is a combined measure defined as $(2 \times precision \times recall)/(precision + recall)$. TP is the number true positives, *TN* is the number of true negatives, *FN* is the number of false negatives, and *FP* is the number of false positives. The precision, recall, and F_1 scores are reported in Table 4.2 for both Naïve Bayes classifiers, one using the edge feature values (non-weighted and weighted values) and the other using *TF-IDF* values of document terms.

Table 4.2 Micro-averaged Evaluation Results

Experiment	**Precision**	**Recall**	**F_1 Score**
NB (Edges)	0.907	0.883	0.895
NB (Weighted Edges)	**0.925**	**0.924**	**0.924**
NB (*TF-IDF*)	0.847	0.860	0.853

4.3.3 Discussion

The results show that using the edge features significantly improved the classification performance, compared to a baseline classifier that uses the *TF-IDF* vector representation. Using edge weights showed an additional performance gain over that of the non-weighted representation. Overall, the precision was improved by 9.2% and the recall by 7.4%. Clearly the use of edges and their weights provided a better representation of the documents and their content. In the proposed method, each graph edge embeds information of the corresponding connected concept nodes as well as the semantic relationship that exists between them. Intuitively, an existing relationship found in a document provides additional details of one or more topics discussed in a document. Such information provides a classifier with additional discriminative capabilities when making predictions, especially when the data is unstructured as is the case for text documents with many underlying interlinked topics of the same or different categories.

The results presented in this chapter also attempt to answer the research question RQ1 stated in Chapter 1 by showing how concept relationships, represented by edges, can be used to significantly improve a classifier's performance.

4.4 Summary

This chapter describes an additional experiment showing how semantic information can be quantified in terms of graph edge weights and used in classification. The results further demonstrate how embedded semantic relationships can improve a classifier's performance when compared to standard representations.

CHAPTER 5

APPLICATION: BIOMEDICAL TEXT CATEGORIZATION USING GRAPH KERNELS

In order to further study the usefulness of the graph representations discussed in the previous chapters, this chapter introduces graph kernels and describes how they can be used in text classification tasks. Kernels allow computing similarities between graphs using their structural features, and thus can deal with sparseness of the graphs. Two different kernel functions are used: the first is a linear kernel and the second is a set-based kernel. Both kernels are edge-based and thus compare graphs based on their underlying structure. This method is also compared to a baseline non-graph classification approach.

5.1 Introduction

Kernel functions for structured data, including graphs, have garnered a particular interest as they provide elegant ways of handling the complexity of the data. In this chapter, two kernel functions are used to compute the similarity between graphs that represent text documents. The first is a set-based kernel function based on set matching. It computes the overall similarity of the graphs based on the similarity of their edges. This approach will evaluate two document graphs as similar if they both share a large number of concept relationships that might exist among them. The kernel function used allows dealing with disconnected graphs and is relatively simple to compute. In addition, the results of a simple linear kernel that computes the cosine similarity between the edge weight vectors of a pair of graphs are reported.

Several approaches to text categorization using graph representations have been explored as outlined in Chapter 3. The presented approach provides a consistent method of representing documents while generating the nodes and edges for each document graph. While previous works have focused on nodes that encode specific words or sentences, the approach described here focuses on higher level concept graphs that encode specific biomedical concepts as nodes in a document graph. These concept nodes and relationship edges are mapped from the text into a regular and controlled vocabulary for describing documents, and thus provide a more consistent representation of the terms used within different documents. Using such a controlled vocabulary ensures that matches between concept nodes reliably indicate similarities between documents, especially when the edge kernels are used.

The presented technique is applied to the same set of biomedical text documents collected from different journals of medicine and related fields. The documents are categorized by the journal they were published in.

5.2 Related Work

5.2.1 Graph Kernels

Graph kernels have been used for many learning tasks on both structured and unstructured data. A kernel function is a mapping between a pair of graphs into a real number. A common preprocessing used for graph classification is projecting the graph onto a kernel space using a kernel function. One possible kernel function can be defined as an inner product between two graphs and must be positive-semidefinite and symmetric. Such a function embeds graphs or any other objects into a Hilbert space, and is termed a Mercer kernel from Mercer's theorem. Kernel functions can enhance classification in two ways: first, by mapping vector objects into higher dimensional spaces; second, by embedding non-vector objects in an implicitly defined space. The advantages of mapping objects into a higher dimensional space, the so called kernel trick, are apparent in a variety of cases where objects are not separable by a linear decision boundary. This implicit embedding is not only useful for non-linear mappings, but also serves to decouple the object representation from the spatial embedding. A kernel function need only be defined between data objects in order to apply a kernel classifier. Such a kernel classifier can then be used for classification of graph objects by defining a kernel function between graphs, without explicitly defining any set of graph features.

Kernel functions for graphs have received much attention recently. The simplest kernels are defined in terms of set operations between nodes and edges. Some more sophisticated developments include kernels based on comparing simple structures such as paths between two graphs such as the shortest path [136], marginalized [137] and spectrum [138] kernels, as well as cycles [139]. Other kernels rely on more complicated structure comparisons such as between subtrees [140] and subgraphs [141]. Some rely on direct matching of graph substructures [142]. String kernels were used in text classification in [143]. The feature space was generated using all string subsequences and the kernel measured the similarity of documents based on the similarity of those subsequences of strings. [70] used a semantic kernel that incorporates Wikipedia background knowledge to enrich the document representation. They achieved improved accuracy in document classification when compared to traditional bag-of-words representation.

5.3 The Approach

As described in the previous chapter, the presented method consists of two major components. The first is the graph construction part, in which the graphs are created in the same way as described in section 4.2.1. Assigning the edge weights and the feature reduction procedures are also done in a similar fashion as described in section 4.2.2. The second component is the application of a graph kernel function to compute the similarities between the generated graphs and a kernel classifier to discriminate between the documents given their embedding in the kernel space.

Figure 5.1 shows the data flow of the procedure of extracting concepts and relationships as well as feeding them into a graph kernel function for classification. In brief, the process is as follows: first, a set of biomedical articles are selected from different journals; next, biomedical concepts are extracted from the documents and mapped to concepts from the UMLS database; concept relationships are then extracted and used to link the concepts, resulting in the concept graphs; a kernel matrix is prepared by computing similarities between the graphs; and finally, the kernel matrix is used for learning and prediction of the documents' target classes. The process of learning the classifier and making the predictions is described in the next section.

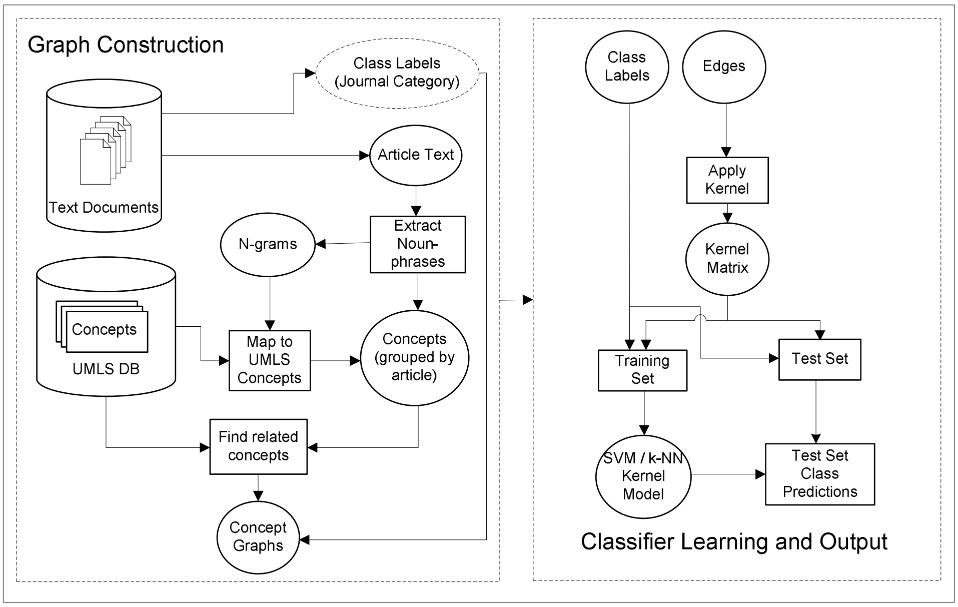


Figure 5.1 System architecture.

5.3.1 Classifier Learning with Kernels

After transforming the set of articles into a set of graphs, a graph kernel function is applied to compute the similarity between all pairs of graphs, and the resulting kernel matrix is used for classification. Two different edge kernels were used in the experiments.

The first is a simple set-based kernel that is used to measure concept graph similarity based on the number of shared edges. There are a couple properties that make a set-based kernel function attractive. The first reason is that the set computations used are easily implemented and understood, leading to a kernel function that is easy to interpret, which results in a greater confidence in producing reliable measures of graph similarity. The second reason is that many of the concept graphs are disconnected or sparse, with many more nodes than edges, which can pose problems for some graph mining algorithms. By using the edge kernel function the sparseness issue is eliminated, as the similarity between a pair of graphs will be highly dependent on the connected components that often represent the core of a document's topic or key concept sets. This kernel function is based on the Jaccard coefficient (also sometime referred to as the Tanimoto kernel) [144], [145]. It computes the similarity between two graphs *x* and *y* as the ratio of the cardinality of the intersection of the edges sets E_x and E_y to the cardinality of their union:

$$K(x, y) = \frac{|E_x \cap E_y|}{|E_x \cup E_y|}$$

The second is a common normalized linear kernel based on the cosine similarity between the edge weight vectors of a pair of graphs. The kernel function returns a normalized inner product of the weighted vectors:

$$K(x, y) = \frac{< w_x, w_y >}{\|w_x\| \|w_y\|}$$

where w_x and w_y are vectors of edge weights of graphs *x* and *y*.

Once a kernel between all graphs is computed, the graphs' similarities result in a kernel matrix. This matrix can then be used in a kernel-based classifier to make predictions on new data. The kernel matrix is input to a support vector machines (SVM) classifier and a k-nearest neighbor (k-NN) classifier to make classification predictions, or in other words, to predict to which journal a certain document belongs.

5.4 Experiments

In addition to the SVM and k-NN classifiers, three common text-based classifiers are used: Naïve Bayes (NB), SVM, and k-NN classifiers for comparison and evaluation. These classifiers use the common vector space model representation, where each document is represented as a vector of TF-IDF weights of the terms in the text [29]. This allows validating the utility of using graph structures over the vector-based representation, where concept relationships are not considered in a classifier's learning and prediction tasks. The same dataset described in Chapter 3 is also used in this experiment.

5.4.1 Model Evaluation

The training and test datasets were obtained from the kernel matrix and the documents' class labels using 10-fold cross-validation. In each validation trial one set was reserved for testing and the other nine were used for training. The evaluated models include those of the kernel-based SVM and k-NN classifiers as well as those of the text-based NB, SVM, and k-NN classifiers that use a vector space representation of the text documents.

For each classifier the micro-averaged accuracy, precision, recall and F_1 scores over all documents in the test dataset are reported. The results are averaged over the ten cross-validation trials. Accuracy (a) is defined as $a = (TP + TN) / S$ where TP stands for number of true positives, TN stands for number of true negatives and S is the total number of testing samples. Precision (p) is defined as the ratio of true positives to the total number of positives predicted by the classifier: $p = TP / (TP + FP)$ where FP is the number of false positives. Recall (r) is defined as the ratio of the number of true positives to the total number of positives present in the test dataset: $r = TP / (TP + FN)$

where *FN* is the number of false negatives. The F_1 score is defined as the inverse of the arithmetic mean of the reciprocal values of precision and recall: $F_1 = 2\ p\ r\ /\ (p + r)$. The performance results are shown in Table 5.2 below.

Table 5.1 Classification Performance

Classifier	Accuracy	Precision	Recall	F_1 Score
sk-SVM[1]	**0.933**	**0.937**	**0.932**	**0.935**
sk-kNN[2]	0.926	0.927	0.925	0.926
lk-SVM[3]	0.913	0.930	0.909	0.919
lk-kNN[4]	0.901	0.906	0.901	0.903
t-NB[5]	0.849	0.847	0.860	0.853
t-SVM[6]	0.849	0.864	0.841	0.852
t-kNN[7]	0.830	0.826	0.817	0.821

1. Set-based-kernel SVM classifier
2. Set-based-kernel k-NN classifier
3. Linear-kernel SVM classifier
4. Linear-kernel k-NN classifier
5. Text-based NB classifier
6. Text-based SVM classifier
7. Text-based k-NN classifier

5.4.2 Analysis of the Results

It is clear, as in any classification task, that the choice of features is a critical factor that significantly affects a classifier's performance. Compared to text features used in conventional classifiers, the proposed graph representation preserves significant structural information that is often embedded in a text document. This information, represented by graph edges, captures a significant level of a document's semantic concept relationships, and thus, provides a classifier with a better feature set that can help in the classification task. In practice, such features are often used by human domain experts for a better understanding of the topics embedded in the text and allow making better decisions and predictions in learning tasks.

The results show that using simpler models, not only provides a more elegant solution to the classification problem, but also results in considerable performance gain in terms of classification predictions. On one hand, the set-based edge kernel performed better than more complex kernel classifiers attempted in prior pilot experiments. On the other hand, it also outperformed the weighted linear kernel which also requires the additional overhead of computing the feature weights.

Overall, all kernel-based classifiers outperformed the standard text-based ones, whether using SVM or k-NN. SVM performed slightly better than k-NN using both kernels.

5.5 Conclusion

In this chapter, an additional graph-based approach for text categorization is presented. Using the graph kernels, the underlying structure of the text documents, whereby concept relationships are preserved, is explicitly incorporated into the representation used by the classifiers.

Two graph kernel functions are defined to compute the similarity between the graphs using both a set-based comparison of edges and a cosine similarity measure between edge weight vectors. An SVM classifier and a k-NN classifier using both kernel functions are applied on a set of documents collected from different medical journals and the classification performance is reported. The results show that the rich graph representation of documents improves the classification performance significantly, when compared to other common *TF-IDF* text-based classifiers.

In addition to the results of the previous chapters, this experiment also attempts to answer the research question RQ1 by showing how the graph structure can be used effectively in making classification decisions.

5.6 Summary

In this chapter a graph mining approach to the problem of text categorization is presented. The process of building concept graphs and the classification algorithm are described through a number of experiments. Experimental datasets, the model construction, evaluation, and the analysis of the results are presented, supporting the argument that using the graph structure improves the performance of the classification algorithm.

CHAPTER 6

APPLICATION: BIOMEDICAL CONCEPT EXTRACTION USING CONCEPT GRAPHS

To further study the effectiveness of concept graphs a concept extraction method that uses graph representations of published articles is evaluated in this chapter. Extracting key concepts from text documents not only involves identifying key terms but also requires understanding the content through those terms. Identifying relations between the terms in the text provides a better understanding of how the concepts behind those terms are contextually and inherently linked to each other and to the main topic in an article. In this chapter a graph representation of a document is proposed, where graph features are used to improve the ranking of concepts extracted from a text document.

Scientific publications are often associated with a set of keywords to describe their content. Automating the process of keyword extraction and assignment could be useful in indexing electronic documents and building digital libraries. In this study a new approach to biomedical concept extraction, using semantic features of concept graphs, is proposed. Full-text documents are represented by graphs and biomedical terms are mapped into predefined ontology concepts. Concept relationship weights are included in the representations to improve the ranking process of potential key concepts. Both objective and human-based subjective evaluations are performed. The results show that using the relation weights significantly improves the performance of concept extraction. The results also highlight the subjectivity of the concept extraction procedure and its evaluation.

6.1 Introduction

Digital collections are witnessing rapid growth in various domains. In the process of building digital libraries, labeling or assigning a set of keywords to text documents could be very expensive as it requires great effort and time as well as domain expertise. As a result, automating this process is of interest to organisations that maintain huge archives of digital content.

Authors usually provide a set of keywords or labels to represent their articles and describe the content briefly. The keywords are used to associate documents with different topics or concepts that would later help in classification and searching tasks within large collections. Nowadays, digital libraries require that authors provide a set of keywords together with their article before being indexed and published. In some cases, this process is automated where documents are labeled with the help of controlled vocabulary sources using domain knowledge or publishing information. However, much of the digital content especially from old un-indexed archives remains unlabeled [146]. As a result, merging those un-indexed documents into existing digital libraries could be very costly and in some cases infeasible without any automation.

Automatic keyword or concept extraction techniques have been proposed over the past decades to help label text documents and have been used in various applications. The applications include: text classification programs [147], browsing applications [146], indexing and searching documents in large collections thus improving retrieval performance [148], document summarization [149], and abstract generation [150]. Many techniques have shown satisfactory performance but not close enough to the manual human labeling. However, the available tools offer good keywords suggestions that can be used by humans for different labeling purposes.

In this study, a biomedical concept extraction system is presented. The system can be applied to documents in the biomedical literature. The main goal of the technique is to extract key concepts that represent biomedical articles in a way similar to how authors assign keywords to articles. In the context of text mining, concepts can be defined as ideas or meanings behind specific terms in a text document. Usually, most of the concepts in the text are represented explicitly by biomedical terms. Some examples of biomedical concepts are protein names, gene names, diseases, or therapy types. Concepts can also be of higher level and not explicitly mentioned in the text. These are sometimes referred to as semantic types. For example the concept *Heart Failure* is a specific instance of *Heart Disease* which is considered as a concept itself.

Manual extraction of concepts representing papers in a large collection is a daunting and costly task. The difficulty lies in the fact that keywords extracted from the document refer to concepts of different semantic or abstraction levels and range from very specific to very general. In addition, there are no strict rules or methods of assigning keywords to an article. In most cases authors are given the freedom to provide a number of concepts they think are the most representative of the whole text. The task is somehow subjective as different experts might give a different set of concepts to the same paper.

The presented approach is based on concept graphs where the relationships between concepts are used to calculate concept weights for ranking and extracting key concepts that are considered the most important and representative of a biomedical article. Each article is represented by a graph structure where the nodes correspond to the biomedical concepts of the text and the edges correspond to the relationships among them. The proposed technique is applied to a set of published biomedical papers and compared to a simpler version that does not take relationships into account. The method is also compared to KEA, a well-known keyphrase extraction software [103]. Both unsupervised and supervised methods are used to rank the candidate concepts of the graphs. The evaluation measure used is the number of matches achieved by comparing the extracted concepts to author provided keywords from the text. In addition, two author involved experiments are conducted and the respective results are compared to KEA and to the author provided list of keywords, from the authors' point of view. The experiments' contribution is as follows. 1) A novel concept extraction technique based on concept graphs built using biomedical ontology mapping. The system uses additional semantic relationships of the graphs in weighting and recommends key concepts similar to author provided keywords in biomedical publications. 2) The results show that using the semantic concept relations in addition to occurrence frequency weights significantly improves the concept extraction process. 3) The results also show that on average, authors prefer the extracted concepts of the proposed method to KEA's extracted keyphrases. 4) The subjective experiments provide additional insight in the evaluation process of concept extraction. The results show that the importance of concepts cannot always be captured by simple comparison to the keywords used by authors for labeling their articles.

6.2 Related Work

6.2.1 Keyword Extraction

There are various approaches to handle the keywords extraction problem. Many of them are based on probabilistic approaches and statistical features such as word counting, inverse document frequency (*IDF*) and so forth. In [151], the authors identified the keywords of a document by using the inverse document frequency for finding the important nouns and their connectivity with other nouns and verbs. Similarly, [152] used term frequency to emphasize keyphrases in target documents based on the occurrence of words. They also made use of the HTML structure of web pages to evaluate term importance in the web page in an attempt to identify general concepts. Mei et al. [153] proposed a probabilistic approach to label multinomial word distributions with meaningful phrases and cast the labeling problem as an optimization problem involving minimizing the Kullback-Leibler divergence between word distributions and maximizing the mutual information between a label and a topic model. Their experimental results show that this approach is effective and robust when applied on different genres of text collections to label topics generated using various statistical topic models.

KEA is a widely used algorithm for extracting keywords from text documents. It is usually evaluated by comparison to the keywords provided by the authors. For instance, based on a large test corpus, KEA's performance was assessed by comparing the extracted keyphrases to the ones chosen by the documents' authors, when a fixed number of keywords are extracted [103]. Arguing that a document's author-specified keyphrases might not be its best possible set of keywords and might not be exhaustive and appropriate for the purposes of summarization, [154] described a human evaluation of KEA. Their results show that KEA is also able to extract good keywords, as measured by human subjects. However, KEA was primarily used to extract keywords and was evaluated based on its capability of extracting keywords. Little work has been done to explore its ability of identifying key concepts from texts in biomedical domain. In this study, KEA's ability to extract key concepts, based on both objective assessment and human judgment, is evaluated.

6.2.2 Biomedical Concept Extraction

One of the most widely used concept extraction systems in the biomedical domain is MetaMap [57] which maps biomedical terms to concepts in the UMLS Metathesaurus [30]. It uses a knowledge intensive approach based on symbolic, natural language processing and computational linguistic techniques to identify all biomedical concepts from textual input. [148] evaluated the performance of MetaMap using a selected subset of curriculum documents and found out that MetaMap identified key medical concepts with a recall of 81% and a precision of 89%. A study reported by [155] compares the performance of MetaMap against that of six people. Their results indicated that MetaMap was able to identify most concepts that were represented in the UMLS and also many other concepts that people did not.

In reference [156], the authors used a domain based dictionary look-up for recognizing known terms and a rule engine that can be easily modified to identify a different class of entities for discovering new terms. Their results indicated that the combination of dictionary look up and rules was able to achieve a precision of 87% and a recall of 94% on the GENIA [20] 1.1 corpus for extracting general biological terms based on an approximate matching criterion. Similarly, [157] developed a biomedical concept extraction system called POSTDOC which also uses UMLS Metathesaurus to recognize relevant main concepts terms. They evaluated POSTDOC's ability to identify UMLS Metathesaurus biomedical concepts in medical school lecture outlines and found the precision and recall varied over a wide range. Another dictionary-based biomedical concepts extraction approach was developed by [158]. Instead of capturing all words of a concept, their approach, referred to as *approximate dictionary lookup,* captured only the significant words. Using UMLS as the dictionary and compared to basic exact dictionary lookup their system was able to increase the recall from 26% to 58% when evaluated on the GENIA corpus.

Heuristic approaches were also applied to biomedical concept extraction. In [114], the authors proposed a graph model to simultaneously extract keywords and summaries from a single document based on an iterative reinforcement method. In [159], a modified Markov heuristic is proposed to identify the relevant concepts in the biomedical domain. Their idea is to automate the retagging of certain verbs as adjectives when in the vicinity of other parts of a noun phrase by incorporating existing sets of curated phrases into the training process.

6.2.3 Semantic Features in Text

Semantic approaches are also widely used to identify important terms that describe the topic of a document. In [108] the authors exploited the semantic structure at both sentence and document levels. Their models combined statistical features and the conceptual ontological graph representation that represents the sentence structure while maintaining the sentence semantics in the original document. Similarly, linguistic knowledge such as syntactic features is often adopted in the keywords extraction task.

[160] observed the performance of keyword extraction using simple statistical measures as well as syntactic information. The experimental results indicated a dramatic improvement of the keyword extraction performance when syntactic information was added to the terms as additional features. In [64] a similar technique that uses semantic hyperlinks that exist in Wikipedia to connect nodes in a concept graph is proposed. The concepts are ranked based on frequency and link saliency scores. In [161] an ontology-based conceptual representation of biomedical content is proposed. The authors exploit semantic relationships to enhance scientific domain search experience.

In [162] a news video retrieval technique that utilizes extracted concepts from video shots is described. The semantic relations between concepts are used to build a graph and the interactions between the concepts are used as features for classification. Huang et al. [111] presented a keywords extraction algorithm that treats each document as a semantic network that holds both syntactic and statistical information. A semantic network model developed treats each term as a node and a relation between two terms as an edge. Their supervised system was able to provide an overall precision of 80%. In [163] the authors present KEA++ which improves automatic keywords extraction by using semantic information on terms and phrases gleaned from a domain-specific thesaurus. Their approach to keyphrase indexing used a machine learning technique and semantic information about terms encoded in a structured controlled vocabulary. Knowing that a keyword of a text should be semantically related with the words of the text, [110] designed a lexical chain that holds a set of semantically related words of a document and used it to represent the semantic content of a portion of the document. They presented a keywords extraction method that uses the features based on the lexical chains in the selection of keywords for a document. In [164] semantic relationships were used to derive concept hierarchies from documents using subsumption, a type of co-occurrence among concepts. The resulting hierarchy resembles a directed acyclic graph, mainly showing parent-child relationships between a pair of topics extracted from the text. They used subsumption as a means to associate related terms, by checking whether the documents in which the child term occurs are a subset of the documents in which the parent term occurs.

6.3 The Approach

In this section the proposed approach is presented. The details of graph construction, concept mapping, and concept ranking are explained. Figure 6.1 shows the system diagram. Step 1 is the named entity recognition (NER) process. Step 2 is mapping the recognized entities to concepts from a controlled vocabulary database. Step 3 is the process of connecting related concepts. Step 4 is ranking the concepts by their weights and Step 5 is merging similar concepts into one label. The detailed description follows in the next section.

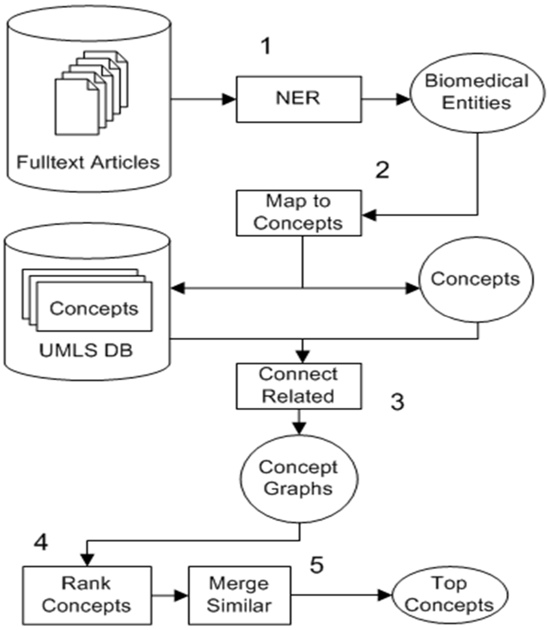


Figure 6.1 Graph construction and concept extraction.

6.3.1 Graph Construction

As previously mentioned, each full-text document is represented by a graph of concept nodes and relationship edges. For each text document all the concepts are identified and added to the graph as nodes. To extract the concepts from the text, LingPipe's [19] NER package (trained on the Genia corpus [20]) is used to identify biomedical named entities. The extracted named entities are biomedical keyphrases in the text like *"5 and 10 lM parthenolide"*, *"endoscopy"*, or *"myocardial infarction"*. To ensure that the identified named entities correspond to a controlled set of vocabulary, the phrases are mapped to concepts from the UMLS database (Step 2). Mapping the named entities into UMLS concepts involves comparing all potential substrings of the keyphrases extracted by NER since those keyphrases are sometimes longer than the concepts in UMLS and contain additional adjectives or terms. In case multiple concepts can be mapped from one named entity string, all corresponding concept nodes are added to the graph and the ones with higher weights will be favored in the final concept extraction process as described in the next section. For example, the phrase *"acute renal failure"* can be mapped to the concepts *"acute renal failure"* and *"renal failure"* from UMLS and thus both concepts are added to the graph. The mapping process can be done through exact and inexact matching of strings between the text and UMLS. Although inexact matching would increase recall, it would decrease the precision by mapping irrelevant concepts. Exact string matching is used in this experiment, since the number of identified concepts is large enough for the purpose of the proposed method (around 128 concepts per full text document). Also, UMLS contains millions of records that span most of the known biomedical concepts and are available in different common written formats.

The graph nodes hold the string descriptions of each concept and the corresponding concept unique identifiers (CUIs). A concept in UMLS has only one unique identifier and a set of corresponding string descriptions. A concept string might refer to multiple concepts with different meanings whereas a CUI refers to only one concept associated with one or more string descriptors. Concept names might slightly vary because of the different vocabulary sources merged in UMLS. The multiple CUIs are implicitly disambiguated by possible relations that might be added to the graph. For example, the term *Ganglion* might refer to 2 different concepts in the biomedical domain. In UMLS, the first (CUI=C0017067) is *a cluster of nervous tissue* and the second (CUI=C1258666) is *a tumor-like lesion.* If concepts like *Nerve*, *Synapse,* and *Basal Nucleus* are present in the same text as *Ganglion*, then the first meaning is implicitly suggested and that will be emphasized later in the weighting process.

Mapping the biomedical entities into predefined concepts also allows looking for possible relationships among them within the ontology. After adding all concept nodes to the graph, the related concepts using UMLS can be identified. Relations in UMLS are based on the CUI as a reference key. For each pair of nodes, if a semantic relationship between them exists in UMLS it is added as an edge between the corresponding nodes (Step 3). As in previous chapters, the relationships are of semantic nature and include *synonym, similar, narrow, broad, qualified-by, parent, child,* and *sibling,* relationships.

6.3.2 Concept Weights

Three weight components are used for ranking the top concepts to be extracted (Step 4).

1. *cf*: The concept occurrence frequency in the text document.

2. *idfw*: The inverse document frequency weight of a concept:

$$idfw_i = 1 - \left(\frac{\log(idf_i)}{\log(N)} \right) \tag{6.1}$$

where idf_i is the number of documents term i occurs in, and N is the total number of documents indexed. This weight is similar to the traditional inverse document frequency (*IDF*) measure [165] except that the index is built beforehand only once using a fixed dataset of over 20,000 Pubmed documents spanning different topics. This weight ensures common biomedical concepts are given lower weights due to their lower discriminatory value. *idfw* is a value between 0 and 1 where lower values indicate that a concept term is a very common one in the biomedical domain.

3. *cw*: The connectivity weight of a concept node. This weight quantifies the importance of a concept in terms of its relationships to other concepts in the text. In other words, it is a measure of the node connectivity within the graph. Two versions of this weight are used in the experiments. The first (*cw₁*) is simply the number of edges of a concept node. The second (*cw₂*) is the magnitude of the relations weights vector for a concept:

$$cw_1 = n; \quad cw_2 = \sqrt{\textstyle\sum_1^n cf_i^{\,2}}; \tag{6.2}$$

where *n* is the number of concepts related to concept *i* and cf_i is the frequency of a related concept *i*. The value of cw_2 not only captures how much a concept is related to other concepts but also how much it is related to important concepts of high frequencies in a document. Later in Section 6.4, the results demonstrate that using cw_2 yields better results than using cw_1.

The first two components are combined into *cfidf*, a weight similar to the well-known *TF-IDF* measure that is widely used in information retrieval [165]. This measure ensures that concepts of high intra-document and inter-document significance are given higher scores. *cfidf* is further normalized using min-max normalization, as shown below, before it is combined with the *cw* weight:

$$cfidf = \frac{(cf\ .\ idfw) - \mathrm{min}\_cfidf}{\mathrm{max}\_cfidf - \mathrm{min}\_cfidf} \tag{6.3}$$

where *min_cfidf* and *max_cfidf* are the minimum and maximum *cfidf* values in a document.

The connectivity weight is also normalized as such:

$$cw' = \frac{cw - min\_cw}{max\_cw - min\_cw} \tag{6.4}$$

where *min_cw* and *max_cw* are the minimum and maximum connectivity weight values in a document.

The overall weight of a concept is the product of the 2 normalized weights:

$$w = cfidf \, . \, cw' \tag{6.5}$$

6.3.3 Merging Similar Concepts

Before the top ranked concepts are extracted, similar concepts are merged and given one label to avoid redundant results and to achieve better ranking (Step 5 in Figure 6.1). For example, the concepts *ganglion, ganglion cell,* and *retinal ganglion cell,*defined as different concepts in UMLS, are merged and labeled as *'ganglion / ganglion cell / retinal ganglion cell'*. Concepts are merged if either their stem word versions are the same or if one is a substring of the other and the string distance is below a certain threshold. The average of both the edit distance and the Jaccard distance [166] are used. Based on the weighting scheme described earlier, the top ten concepts are extracted from the ranked list of concepts.

It is worth mentioning here that in an earlier pilot study clustering was applied to the list of top concepts in order to extract the top concepts from each cluster. The idea behind this was to span all different key topics in the document and avoid extracting redundant concepts with similar meanings. This was done in order to incorporate the semantic similarities in addition to the string similarities described above. k-medoids, a variant of the k-means algorithm, was used, where the distances between the nodes of a graph were calculated using string and node relationship distances. Compared to the author provided list, this approach performed slightly worse than the one described earlier. For this reason this technique was not used in the final experiments as it did not show significant improvement over the proposed method. Also the merging procedure described above took care of much of the concepts grouping. One interpretation of the fact that clustering did not show significant improvement is that in many cases authors choose similar keywords that are not necessarily distinct enough to be in different clusters. After all, most of the key concepts happen to be related somehow within a document and although the publication authorities might recommend that the author keywords be distinct, generally it is not a strict requirement.

6.3.4 SVM Ranking

The method discussed above is unsupervised and ranks the concepts by the composite weight described in Section 6.3.2. In addition, another semi-supervised version of ranking is presented in another experiment where a model is built using the same graph node weights as features. In particular, the Support Vector Machine ranking algorithm SVM^{rank} [167] is used. Using the model built from the training data, the SVM^{rank} classifier predicts the ranking of the candidate concepts, where the ones ranked towards the top have a higher probability of being key concepts representing an article. More usage details are discussed in the experiments section. SVM^{rank} is based on Vapnik's Support Vector Machines [168], [169] and aims to order a new set of objects as accurately as possible by learning a function from preference examples. In SVM^{rank}, a model can be learned to select a ranking function from a family of ranking functions which generalize well beyond the training data. SVM^{rank} has been applied to document retrieval [170], where click-through data was used to deduce pair-wise training data for learning ranking models.

6.4 Experiments and Evaluation

To evaluate the performance of the proposed technique, four different sets of experiments were conducted. The first two experiments provide an objective comparison of the approach to KEA. The other two are subject-based evaluations where the articles' authors were involved in the evaluation. The results of each experiment show a different aspect of the usefulness and effectiveness of the proposed system in addition to the subjectivity of the labeling process.

6.4.1 KEA

KEA is an automatic keyphrase extraction algorithm developed by members of the New Zealand Digital Library Project. It uses the Naïve Bayes machine learning algorithm for training and keyphrase extraction. KEA builds a prediction model using training documents with known keyphrases, and then the model is used to identify keyphrases in new documents. The implementation of KEA used in the experiments is available for public [171]. KEA was trained using 450 biomedical documents to tune its parameters of the extraction algorithm and learn a model that was used to extract keyphrases from the test documents. Every phrase that occurs in the document is thus a potential keyphrase of the document. Using KEA, ten keywords from each test document are extracted and the precision results are compared against the proposed method.

6.4.2 Objective Comparison

In this section, the two experiments performed to evaluate the proposed method against KEA are presented. The first is based on unsupervised ranking of concepts while the second uses a semi-supervised ranking algorithm based on support vector machines.

6.4.2.1 Unsupervised Ranking. In this experiment, the performance of the proposed technique is compared to that of KEA, first using the *cfidf* weights only and then using the compound weight $w = cfidf \, . \, cw$ that incorporates the connectivity weight. The two versions of the connectivity weight described earlier are used; the first is the number of edges or relationships of a concept node (cw_1) and the second is the magnitude of the frequencies vector of related concepts (cw_2). 100 Pubmed articles of different topics were used in this experiment. The chosen articles contain the author provided (AP) keywords in the text. In total there were 651 keywords associated with the 100 documents (on average, 6.51 keywords per document). To determine whether the output concept strings match with the original AP strings, the similarity measure described below is used. A match occurs if any of the following is true:

1. *Exact match: both strings are exactly the same.*
2. *Stem match: stem words of both strings are the same.*
3. *Substring match: AP string is a substring of output.*
4. *Relation match: a relation exists in UMLS between an AP keyword and output.*
5. *String distance: the string distance is below a certain threshold (average of Edit and Jaccard distance).*

Note that this is not intended to be an exact match evaluation since it would fail to match many relevant results. Although the relation match is somehow weak compared to the other criteria, it is used here as a means for evaluating output that can be regarded as semantically close to an author’s keyword. Practically, the related concepts could serve as synonyms or other related alternatives to the original AP keywords. This match check is applied to the proposed algorithm (CE) and to KEA’s output and thus is used as a relative measure to compare the two methods. The relationships are determined using the same method described earlier in the graph construction section.

The experiment shows that the proposed method is comparable to KEA which is a leading keyphrase extraction software based on a supervised learning technique. The number of matches for the top 3, 5, and 10 extracted concepts are reported in Table 6.1.

Table 6.1 Number of matches for both CE and KEA

	Top 3		**Top 5**		**Top 10**	
	Matches[1]	**Avg**[2]	**Matches**	**Avg**	**Matches**	**Avg**
KEA	214	2.14	331	3.31	**610**	**6.10**
CE[3]	205	2.05	300	3.00	480	4.80
CE*[4]	213	2.13	311	3.11	526	5.26
CE**[5]	**218**	**2.18**	**331**	**3.31**	556	5.56

1. Matches: total number of matches out of the 651 AP keywords.
2. Avg: the average number AP keywords matched per paper.
3. CE: is the concept extraction technique using *the occurrence frequencies* only.
4. CE*: is the concept extraction technique using *the occurrence frequencies* and the additional connectivity weight cw_1.
5. CE**: is the concept extraction technique using *the occurrence frequencies* and the additional connectivity weight cw_2.

The results show that when the semantic relationships are used in ranking the concepts, the number of matches increases significantly: 6%, 10%, and 16% in the case of 3, 5, and 10 extracted concepts, respectively. Using the weights of related concepts also shows an improvement over using only the number of related concepts (cw_2 *vs.* cw_1). This further confirms that capturing additional information from the relationships enhances the ranking procedure. Compared to KEA, CE performs slightly better in the top 3 extracted concepts list, whereas KEA's performance is better for the top 10 list.

6.4.2.2 Semi-Supervised Ranking. For this experiment, the training set used consists of 137 documents and the test set consists of 100 documents (673 AP keywords). Each concept in a graph is considered as a sample in both training and test sets. The feature weights used in this experiment are the occurrence frequency *cf.idfw* and the connectivity weight cw_2 described in section 6.3.2. The target value used in the training process is set to 0 when the concept does not match an author provided keyword and is set to 1 when it is an exact match, stem word match, or substring match. Relationship matches (where a concept from the paper is semantically related to an AP keyword) were not used in the training. However they were included in the test dataset for evaluation purposes (The target value for the relationship matches was set to 0.5). It is worth noting here that using the relationship matches during the training phase introduced an expected precision/recall tradeoff. Although the classifier ranked more related concepts towards the top, the number of exact matches significantly dropped. For that reason, those relationship matches were not included in the training process in the final experiments. In practice, this tradeoff can be optimized according to the application and user requirements. For example, the target value can have more specific values in the range 0 to 1 depending on the type of relation between a concept and an AP keyword. Also, some relations that exist in UMLS might be considered irrelevant and thus can be excluded from both the target value calculation and the connectivity weight calculation. A sample of the test set input used in SVM^{rank} is shown below (similar format is used for the training set except that the target values can only be 0 or 1). The columns respectively, are: the target value, the document ID, feature weights, and descriptions of the document and concept.

```
1.0 qid:1 1:3.55828061479804 2:3.78163975598787 #2408639#Werner syndrome#
0.0 qid:1 1:2.22097590495905 2:1.9287283701509834 #2408639#aging#
0.5 qid:1 1:15.90921796570065 2:5.268294443748759 #2408639#recombination#
```

The results of this experiment are shown in Table 6.2 below. Using both the occurrence frequencies and the connectivity weight resulted in the best performance in terms of number of total matches. Figure 6.3 shows the number of exact matches compared to the number of relation matches using KEA and CE (with and without the connectivity weights). CE outperforms KEA when the top 3 concepts are extracted in both exact and relation matches. As the number of extracted concepts increase (to 5 and 10) KEA extracts more exact matches but CE achieves higher recall as it extracts significantly more related concepts.

Table 6.2 Number of Matches using SVM^{rank}

	Top 3		**Top 5**		**Top 10**	
	Matches[1]	**Avg**[2]	**Matches**	**Avg**	**Matches**	**Avg**
KEA	169	1.69	259	2.59	461	4.61
CE[3]	163	1.63	239	2.39	363	3.63
CE*[4]	**202**	**2.02**	**307**	**3.07**	**511**	**5.11**

1. Matches: total number of matches out of the 673 AP keywords.
2. Avg: the average number AP keywords matched per paper.
3. CE: is the concept extraction technique using the occurrence frequencies only.
4. CE*: is the concept extraction technique using the occurrence frequencies and the additional connectivity weight cw_2.

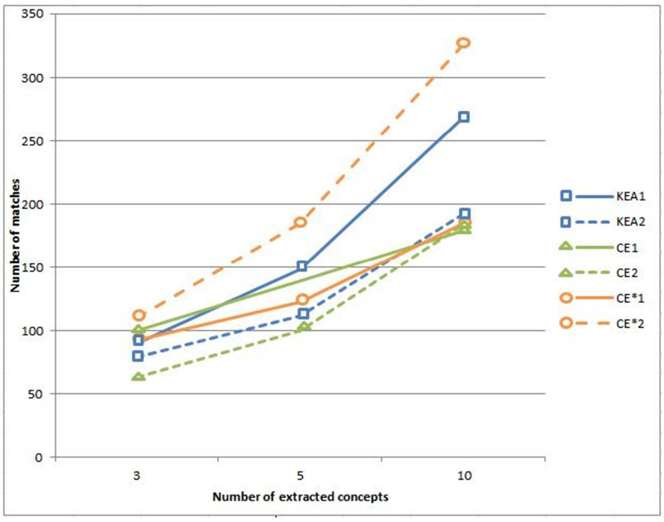


Figure 6.2 Number of extracted concepts: exact vs. related.

KEA^1: The number of exact matches using KEA
KEA^2: The number of relation matches using KEA
CE^1: The number of exact matches using CE without relation weights
CE^2: The number of relation matches using CE without relation weights
CE^{*1}: The number of exact matches using CE with relation weights
CE^{*2}: The number of relation matches using CE with relation weights

6.4.3 Author-Involved Experiments

In most cases the author keywords list serves as a good representation of the paper in terms of key concepts. However, the author’s choice might be affected by personal or external factors. For instance, the keyword list might be limited to certain number of keywords or the author might provide a list that increases the likelihood of publication of the paper [172]. Moreover, the list is not always comprehensive enough to cover all ideas or topics of a paper. For such reasons comparing to the original list of keywords might not be sufficient and thus in the next set of experiments the authors’ feedback is considered to further evaluate the capabilities of the proposed concept extraction system.

The author-involved experiments are divided into two different sets. The first is used to compare author provided (AP) keywords to concept candidates selected by the proposed concept extraction (CE) algorithm. This experiment is intended to validate the effectiveness of the concept ranking algorithm and to justify the importance of the authors' feedback in such evaluation context. The second set was used to compare the performance of the CE technique to that of KEA's using human subjective judgment.

6.4.3.1 CE vs. AP. The first dataset comprises 32 scientific papers of various biomedical topics chosen from several Elsevier biomedical-related journals. 18 authors of those papers were contacted and asked for their help in the evaluation. The authors are either medical doctors or researchers in biological sciences. For each paper, the original AP keywords in the text were extracted and the proposed CE algorithm was applied to the text to extract the top ten candidate key concepts. A list combining both the CE results and the original AP keywords was then formed. Duplicates or merged concepts (for example, '*ganglion cell*' and '*ganglion cell / retinal ganglion cell*') are only displayed once. The shuffled list of concepts and keywords was then sent as an electronic survey form and the authors were instructed to mark the ones they think are key concepts of the paper. No limit on the number of items to be marked was specified. The authors were allowed to mark as many concepts as they thought are relevant key concepts. Moreover, the authors were not asked to provide negative feedback in this study. On average, each author was asked to evaluate two of their own papers, and every paper was evaluated by only one of its authors.

From the results shown in Table 6.3, it can be noticed that the authors have chosen 85% of the AP keywords that were originally listed in the paper. Interestingly, they left out 15% which they did not choose as key concepts. As for the CE concepts, out of the 10 concepts extracted for each paper, they have chosen on average 4.6 as relevant key concepts (3.5 of them are additional concepts not part of the AP list). On average, more than 8.4 concepts are picked per paper whereas the keywords section contains only 5.56. This shows that the AP keywords list might not always cover all key concepts in a paper. The precision of CE is 0.56 whereas that of AP is 0.61. The author-provided keywords are expected to give better performance results. However the precision of CE is not far off, and thus the extracted concepts can be regarded as good candidates for the documents' keyword lists. Another interesting result observed in this experiment is that a substantial proportion of concepts (35%) extracted by the CE algorithm were not originally present in the keywords list but were selected by the evaluators as key concepts.

Table 6.3 Author-evaluated Results: AP vs. CE

AP keywords	**Checked**[1]	**Total**	**Precision**[2]
Mean	4.72 (85%)	5.56	61%
CE concepts	**Checked**[3]	**Total**	**Precision**[4]
Mean	4.62 (46%)	10	56%

1. The number of AP keywords selected by authors as relevant key concepts (including overlaps with CE).
2. The proportion of checked AP keywords out of all the checked concepts for a paper.
3. The number of CE keywords selected by authors as relevant key concepts (including overlaps with AP).
4. The proportion of checked CE keywords out of all the checked concepts for a paper. Note that CE Precision + AP Precision > 1, that's because there are some overlapping terms.

In addition, most CE candidate concepts that were chosen by the authors were ranked among the top 5 in the list. Figure 6.3 shows the number of relevant concepts grouped by their rank. The figure shows the consistency of the proposed technique in terms of ranking concepts.

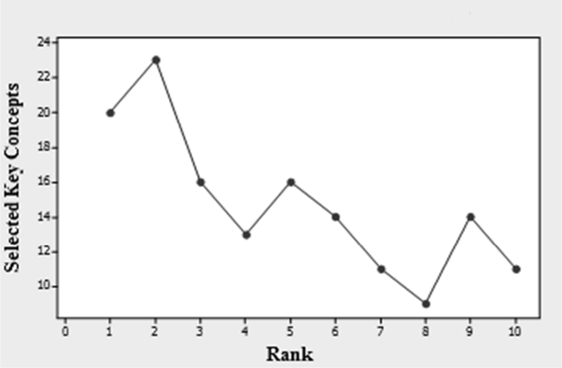


Figure 6.3 CE ranking vs. frequency of selected key concepts.

6.4.3.2 CE vs. KEA. In this section the results of the second author-involved experiment are presented. This experiment compares the proposed technique to KEA. The dataset used in this experiment is composed of 25 biomedical technical papers, collected from the Elsevier's electronic archive as well. There were 11 authors who participated in this evaluation. The procedure is similar to the previous one. In this experiment the top ten results from each of CE and KEA's output for each paper are shuffled and combined into one list. Again, duplicate items are only listed once.

Table 6.4 below shows the results for the 25 papers. For each technique (CE and KEA), the precision is calculated as the proportion of items selected as key concepts out of the total number of key concepts chosen by the authors.

Note that in Section 6.4.1, CE and KEA's performances were analyzed based on the AP keywords list using string and relation matching. In this experiment, the assessment is based on the authors' preferences.

Table 6.4 CE vs. KEA Under Human Evaluation

	CE		KEA	
	Selected[1]	**P**[2]	**Selected**	**P**
AVG	3.64	0.60	2.76	0.40
Stdev	1.75	0.22	2.03	0.22

1. Average number of selected key concepts
2. Precision

The mean precision of CE is 0.6, compared to 0.4 for KEA. A t-test was performed on the mean value and the result validates that the mean value of CE's precision is significantly larger than that of KEA (P-Value=0.0536). The results show that the authors prefer CE's extracted concepts to KEA's keyphrases.

6.5 Discussion

The results of the objective comparison show that the proposed system is comparable to KEA in terms of keyword suggestions. As the number of extracted concepts increases, KEA performed better in exact matching of keywords while the proposed technique provided more related matches (higher recall), especially in the semi-supervised version discussed in section 6.3.4. This is an expected precision/recall tradeoff that arises when semantic relations are considered. One component that can be further improved is the mapping process of terms in the text into UMLS concepts. This is a non-trivial task and may require advanced natural language processing techniques since not all forms of biomedical terms are present as concepts in UMLS.

Based on the author-involved subjective evaluation, the following points were observed. First, a significant number of concepts, which were chosen by authors as key concepts, were not originally present in the keywords list of the paper. Second, some original AP keywords of a number of papers were not selected by the authors as key concepts of those papers. This suggests that the keyword list is not exhaustive and does not represent all the concepts contained in a paper. Also, as mentioned in [172], some keywords might be listed for other purposes where the keyword list may not necessarily be a precise representative of a certain article. Furthermore, the authors who participated in the evaluation are all coauthors of their papers and thus their opinions on whether the terms are key concepts or not may conflict. It is often not uncommon that even experts might have biases or disagreements on the choice of terms [156].

In the subjective evaluation of CE vs. KEA, the results confidently support that CE outperforms KEA and provides more desirable key concepts. Also, Figure 6.4 shows that most of the selected key concepts are ranked high in the list of concept candidates. This shows that the proposed technique is quite effective in terms of weighting and ranking. The results also validate the assumption that the author-involved subjective experiment is necessary to supplement the objective experiment. Compared to the automatic string matching evaluation, human judgment and reasoning allow authors to pick as many or as few concepts from the candidate list as they see fit.

6.6 Conclusion

Extracting concepts from full-text biomedical documents is an important but challenging task. In this chapter a new approach to concept extraction is presented, where concept graphs and their semantic features are used for weighting and ranking concepts in an article. Predefined ontology concept relationships are used, in addition to traditional occurrence frequency weights, to rank the top concepts extracted from a text document. The proposed technique yields promising results when evaluated against the author-provided keywords and against KEA. Referring to research question RQ2, this experiment shows how the structural features of graphs, that represent concept relationships in the text, enhance the ranking process in concept extraction tasks, especially in terms of recall. This is emphasized by the high number of non-exact matches that could be ignored in other baseline methods. The developed automatic concept extraction technique can help authors in labeling their scientific publications by recommending keywords. The technique can also be used in document summarization applications and indexing algorithms of digital libraries.

Exploiting additional features of concept graphs could further improve the ranking procedure. Concept extraction techniques can also be applied to other domains such as the general Web and educational document collections. In addition, concept extraction can be incorporated into text categorization applications where the extracted concepts serve as a reduced feature set of full-text documents, as pointed out in Chapter 3.

6.7 Summary

In this chapter a concept extraction technique that uses graph representations of text documents is presented. The process of constructing graphs from text documents, demonstrating how they can be used in ranking key concepts, is described. The results show that using graph structural features improves the ranking of key concepts extracted from text, especially in terms of recall.

CHAPTER 7

SUMMARY AND FUTURE WORK

7.1 Conclusions

In this work a number of experiments have been studied to explore how text documents can be represented by graph structures that allow capturing the semantic relationships of the content and how this additional information can be used in learning algorithms. The results attempt to answer the following research question: **Can graph representations of text, in which relationships among concepts are preserved, improve the performance of text mining applications, when compared to baseline methods?** This question is divided into two parts, each studied through a set of experiments and evaluations. Chapters 3, 4, and 5 present different approaches to the problem of text categorization and attempt to investigate how concept relationships and external related concepts, captured in graph form, provide a better representation for classifiers to discriminate text content and to make more accurate classification decisions using supervised learning methods. Chapter 6 presents a method of concept extraction and attempts to investigate how the structural properties of a graph provide additional useful attributes for a text document's feature set to improve the ranking of key concepts present in that document.

In Chapter 3 the first method of representing text documents by graphs is presented. The graphs are constructed using a knowledge-based approach that is less dependent on the text content. The representation can be constructed from minimal information extracted from the target documents. This representation encodes concepts and their relationships in the form of graph nodes and edges by mapping them into ontology-based concepts and relationships. This method shows how a knowledge-based representation can be used as an alternative practical representation in the absence of full-text content. A Naïve Bayes classifier is applied on a set of biomedical documents using the aforementioned representation. The results show that the proposed representation can match the performance of a standard Naïve Bayes classifier that uses statistical information from the full text and can outperform it when edge-information is used in calculating class probabilities.

In Chapter 4 the previous experiment is extended by using weighted graph edges as document features. The edge weights are quantified to reflect the significance of the corresponding concept terms and their relationships in the text and are used as feature vectors of the documents. A Naïve Bayes classifier is applied using the edge features representation. The results show a substantial performance increase when compared to a baseline *TF-IDF* Naïve Bayes classifier.

In Chapter 5 graph kernels are introduced and applied to the graph representations of text documents. The kernels are edge-based and compare the graphs based on their underlying structure. Two different kernel functions are used to classify the graphs using a k-NN and an SVM classifiers. The results outperform the baseline text-based methods and further show how the concept relationships could be used as an effective feature set in document categorization.

In Chapter 6 a method of concept extraction from text documents using graph representations is described. Graph structural features are used to enhance the ranking of key concepts of a document and to extract a set of representative concepts that can be used as labels or tags for that document. A set of experiments is presented demonstrating unsupervised and semi-supervised ranking approaches. The method is compared to a common key-phrase extraction tool. The proposed techniques demonstrate a practical method of assigning keywords to documents and show a significant improvement in terms of precision and recall.

7.2 Contributions

The main research contributions of this study to the field of text mining are listed as follows:

- The work presents a practical graph representation framework for several text mining applications, through experiments and evaluations of text categorization and concept extraction techniques. The work is not constrained to those specific applications, as graph representations can be applied to similar text mining applications such as document summarization, document or concept clustering, and topic identification.
- The proposed methods emphasize the importance of representation, semantic features, and structural properties and their impact on the underlying learning algorithms. The motivation behind using those elements is to embed additional information that could be useful in making decisions or predictions in text mining applications.
- The methods can be applied in literature-based discovery applications, where insights and hidden relationships could be mined from large collections of knowledge buried in text documents within certain domains. Improving the

representation, classification, and ranking of text elements is key in finding associations between topics in a dataset.

7.3 Limitations

The discussed methods rely on domain knowledge in constructing the graph representations of documents. This could introduce a limitation in implementation when such knowledge is not available or when the dataset does not represent a specific domain. One could overcome this limitation by using general domain ontologies or controlled vocabularies such as WordNet or Wikipedia to map concept terms from the text and extract their corresponding semantic relationships. Alternatively, natural language processing methods can be applied to the content to extract relevant semantic knowledge from the language structure and the syntax. In situations where external knowledge cannot be incorporated into the representation, the whole dataset could be mined for links that represent semantic relationships or interactions between the entities present in the text, perhaps using statistical learning methods and co-occurrence information, clustering, or classification to predict unknown relationships.

Another consequent limitation present in the methods lies in the concept mapping process. Concept terms present in the text might not always be found in the domain ontology used. Inexact matching can alleviate the issue of not finding exact matches but could introduce a precision/recall tradeoff as some non-relevant concepts can be mapped to the concept terms. A domain ontology should be updated regularly to ensure integrity and information quality as new concepts are added or updated in the data source. In addition, building representations using different mapping techniques could also be tested to find an optimal representation for a certain application.

As far as computation and scalability are concerned, processing graphs can be problematic as the size and number of documents increase in the target dataset. The complex nature of graphs often poses limitations in computation and algorithm development. In this case, parallel or distributed environments could be used to alleviate the computational complexity and to allow efficient processing of large graphs, such as those representing books or documents collected from the web.

7.4 Future Work

The methods presented in this study can be extended in different directions of research in text and graph mining. Additional structural features of graphs can be explored to emphasize concept significance and centrality in a document. This could help in formulating new weighting techniques for document features. Methods of network theory and link analysis can be borrowed to allow finding better associations between pairs of concepts, to improve the ranking of concepts, and to calculate centrality measures of concept nodes. In addition, graph kernels can be further explored to find better ways of computing similarity measures between graphs when applied to document classification tasks. Paths within a graph can also be studied to find relative distances between nodes or subgraphs where those distance measures can be used in learning algorithms. Graph indexing, frequent subgraphs, and graph matching techniques can be applied to text documents to enhance indexing, retrieval, and classification of those documents.

Additional representations of text documents can be explored and compared in different application contexts. Existing feature selection and extraction techniques can also be applied to graphs, where a set of candidate features could be identified and used in algorithms such as ranking and classification. In addition different representations can be constructed in a manner that allows extracting document features efficiently, exploiting certain structural features of graphs.

Another direction that could be investigated is applying the proposed techniques to non-biomedical datasets. Further experiments would give better insight on the scalability and efficiency of those techniques when applied to different domains. Such techniques would also involve constructing domain specific ontologies and evaluating their impact on learning algorithms, such as classification or clustering.

Finally, additional experiments can be conducted using significantly larger datasets to test the scalability of the methods in real world scenarios. When the number and size of documents are considerably large, the text processing and graph construction components might be extremely expensive in terms of computational costs. However, one could take advantage of the recent developments in distributed computing and analytics for ‘big data’, such as the MapReduce paradigm [173]. By using such a framework, the computationally expensive modules, including graph generation, kernel matrix computation, and cross validation, can be performed in parallel on distributed clusters of computers and the results can be combined afterwards, reducing the overall complexity significantly.

7.5 Summary

This chapter concludes the dissertation by summarizing the results of each experiment within the context of the main research question. A brief overview of the contributions and limitations of the current work is given and potential future work directions are highlighted.

REFERENCES

[1] J. Natarajan, D. Berrar, C. J. Hack, and W. Dubitzky, “Knowledge discovery in biology and biotechnology texts: a review of techniques, evaluation strategies, and applications,” *Critical reviews in biotechnology*, vol. 25, no. 1–2, pp. 31–52, Jun. 2005.

[2] S. Ananiadou, D. B. Kell, and J. Tsujii, “Text mining and its potential applications in systems biology,” *Trends in Biotechnology*, vol. 24, no. 12, pp. 571–579, Dec. 2006.

[3] I. Spasic, S. Ananiadou, J. McNaught, and A. Kumar, “Text mining and ontologies in biomedicine: Making sense of raw text,” *Briefings in Bioinformatics*, vol. 6, no. 3, pp. 239 –251, 2005.

[4] J. Dorre, P. Gerstl, and R. Seiffert, “Text mining: finding nuggets in mountains of textual data,” *in Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, 1999, pp. 398–401.

[5] A. H. Tan, “Text mining: The state of the art and the challenges,” *Proceedings of the PAKDD 1999 Workshop on Knowledge Disocovery from Advanced Databases*. 1999.

[6] G. Salton, A. Wong, and C. S. Yang, “A vector space model for automatic indexing,” *Communications of the ACM*, vol. 18, no. 11, pp. 613–620, 1975.

[7] T. Washio and H. Motoda, “State of the art of graph-based data mining,” *ACM SIGKDD Explorations Newsletter*, vol. 5, no. 1, pp. 59–68, 2003.

[8] D. Chakrabarti and C. Faloutsos, “Graph Mining: Laws, Generators, and Algorithms,” *ACM Computing Surveys*, vol. 38, no. 1, p. 2, 2006.

[9] N. Christofides, Graph theory: An algorithmic approach, vol. 8. *Academic press London*, 1975.

[10] A. Schenker, M. Last, H. Bunke, and A. Kandel, “Classification of Web documents using a graph model,” *in Document Analysis and Recognition, Proceedings. Seventh International Conference on*, 2003, pp. 240–244 vol. 1.

[11] J. Liang and D. Doermann, “Logical labeling of document images using layout graph matching with adaptive learning,” *Document Analysis Systems V*, pp. 169–175, 2002.

[12] D. Lopresti and G. Wilfong, “Applications of graph probing to web document analysis,” Web document analysis: challenges and opportunities, p. 19, 2003.

[13] J. Lafferty, A. McCallum, and F. Pereira, “Conditional random fields: Probabilistic models for segmenting and labeling sequence data,” 2001, pp. 282–289.

[14] F. Sha and F. Pereira, “Shallow parsing with conditional random fields,” *in Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, vol. 1, 2003, pp. 134–141.

[15] A. McCallum and W. Li, “Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons,” *in Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003*, vol. 4, 2003, pp. 188–191.

[16] R. McDonald and F. Pereira, “Identifying gene and protein mentions in text using conditional random fields,” *BMC bioinformatics*, vol. 6, no. Suppl 1, p. S6, 2005.

[17] B. Settles, “Biomedical named entity recognition using conditional random fields and rich feature sets,” *in Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications*, 2004, pp. 104–107.

[18] B. Settles, “ABNER: an open source tool for automatically tagging genes, proteins and other entity names in text,” *Bioinformatics*, vol. 21, no. 14, p. 3191, 2005.

[19] “LingPipe.” [Online]. Available: http://alias-i.com/lingpipe/. [Accessed: 23-Mar-2013].

[20] J. D. Kim, T. Ohta, Y. Tateisi, and J. Tsujii, “GENIA corpus-a semantically annotated corpus for bio-textmining,” *Bioinformatics-Oxford*, vol. 19, no. 1, pp. 180–182, 2003.

[21] “BioCreAtIvE I- Critical Assessment of Information Extraction systems in Biology.” [Online]. Available: http://www.pdg.cnb.uam.es/BioLINK/BioCreative.eval.html. [Accessed: 23-Mar-2013].

[22] T. R. Gruber and others, “A translation approach to portable ontology specifications,” *Knowledge acquisition*, vol. 5, no. 2, pp. 199–220, 1993.

[23] O. Bodenreider, “Biomedical ontologies in action: role in knowledge management, data integration and decision support,” *in ‘IMIA Yearbook Medical Informatics*, vol. 67, p. 79, 2008.

[24] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, and others, “Gene Ontology: tool for the unification of biology,” *Nature genetics*, vol. 25, no. 1, p. 25, 2000.

[25] A. D. Baxevanis, “The Molecular Biology Database Collection: an online compilation of relevant database resources,” *Nucleic Acids Research*, vol. 28, no. 1, p. 1, 2000.

[26] D. P. Hill, J. A. Blake, J. E. Richardson, and M. Ringwald, “Extension and integration of the gene ontology (GO): combining GO vocabularies with external vocabularies,” *Genome Res*., vol. 12, no. 12, pp. 1982–1991, Dec. 2002.

[27] B. Peters and A. Sette, “Integrating epitope data into the emerging web of biomedical knowledge resources,” *in Nature Reviews Immunology*, vol. 7, no. 6, pp. 485–490, Jun. 2007.

[28] J. Kohler and S. Schulze-Kremer, “The semantic metadatabase (SEMEDA): ontology based integration of federated molecular biological data sources,” *in Silico Biology*, vol. 2, no. 3, pp. 219–231, 2002.

[29] K. Wolstencroft, R. McEntire, R. Stevens, L. Tabernero, and A. Brass, “Constructing ontology-driven protein family databases,” *Bioinformatics*, vol. 21, no. 8, pp. 1685–1692, Apr. 2005.

[30] D. A. Lindberg, B. L. Humphreys, and A. T. McCray, “The Unified Medical Language System.,” *Methods of information in Medicine*, vol. 32, no. 4, p. 281, 1993.

[31] “Unified Medical Language System (UMLS) - Home,”. [Online]. Available: http://www.nlm.nih.gov/research/umls/. [Accessed: 23-Mar-2013].

[32] A. Burgun and O. Bodenreider, “Mapping the UMLS Semantic Network into general ontologies.,” *in Proceedings of the AMIA Symposium*, 2001, p. 81.

[33] F. Mougin, A. Burgun, and O. Bodenreider, “Using WordNet to improve the mapping of data elements to UMLS for data sources integration,” *In AMIA Annual Symposium Proceedings*, vol. 2006, p. 574, 2006.

[34] O. Bodenreider, S. J. Nelson, W. T. Hole, and H. F. Chang, “Beyond synonymy: exploiting the UMLS semantics in mapping vocabularies.,” *in Proceedings of the AMIA symposium*, 1998, p. 815.

[35] F. Volot, P. Zweigenbaum, B. Bachimont, M. B. Said, J. Bouaud, M. Fieschi, and J. F. Boisvieux, “Structuration and acquisition of medical knowledge. Using UMLS in the conceptual graph formalism.,” *in Proceedings of the Annual Symposium on Computer Application in Medical Care*, 1993, p. 710.

[36] Y. Huang, H. J. Lowe, D. Klein, and R. J. Cucina, “Improved identification of noun phrases in clinical radiology reports using a high-performance statistical natural language parser augmented with the UMLS specialist lexicon,” *Journal of the American Medical Informatics Association*, vol. 12, no. 3, p. 275, 2005.

[37] A. Keselman, G. Rosemblat, H. Kilicoglu, M. Fiszman, H. Jin, D. Shin, and T. C. Rindflesch, “Adapting semantic natural language processing technology to address information overload in influenza epidemic management,” *Journal of the American Society for Information Science and Technology*, 2010.

[38] M. Swift, N. Blaylock, J. Allen, W. de Beaumont, L. Galescu, and H. Jung, “Augmenting a Deep Natural Language Processing System with UMLS,” *in Proceedings of the Fourth Symposium on Semantic Mining in Biomedicine (SMBM 2010)*, Hinxton, UK, 2010.

[39] L. P. Morales, A. D. Esteban, and P. Gervas, “Concept-graph based biomedical automatic summarization using ontologies,” *in 22nd International Conference on Computational Linguistics*, 2008, p. 53.

[40] L. H. Reeve, H. Han, and A. D. Brooks, “The use of domain-specific concepts in biomedical text summarization,” *Information Processing & Management*, vol. 43, no. 6, pp. 1765–1776, 2007.

[41] R. Verma, P. Chen, and W. Lu, “A semantic free-text summarization system using ontology knowledge,” *in Proceedings of Document Understanding Conference*, 2007.

[42] A. R. Aronson and T. C. Rindflesch, “Query expansion using the UMLS Metathesaurus.,” *in Proceedings of the AMIA Annual Fall Symposium*, 1997, p. 485.

[43] W. Hersh, S. Price, and L. Donohoe, “Assessing thesaurus-based query expansion using the UMLS Metathesaurus.,” *in Proceedings of the AMIA Symposium*, 2000, p. 344.

[44] W. Zhu, X. Xu, X. Hu, I. Y. Song, and R. B. Allen, “Using UMLS-based re-weighting terms as a query expansion strategy,” *in IEEE International Conference on Granular Computing*, May10-12, 2006.

[45] D. Eichmann, M. E. Ruiz, and P. Srinivasan, “Cross-language information retrieval with the UMLS metathesaurus,” *in Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, 1998, pp. 72–80.

[46] W. Pratt, “Dynamic organization of search results using the UMLS.,” *in Proceedings of the AMIA Annual Fall Symposium*, 1997, p. 480.

[47] C. Friedman, “Discovering novel adverse drug events using natural language processing and mining of the electronic health record,” *Artificial Intelligence in Medicine*, pp. 1–5, 2009.

[48] X. Wang, G. Hripcsak, M. Markatou, and C. Friedman, “Active Computerized Pharmacovigilance Using Natural Language Processing, Statistics, and Electronic Health Records: A Feasibility Study,” *Journal of the American Medical Informatics Association*, vol. 16, no. 3, pp. 328–337, May.

[49] G. K. Savova, J. Fan, Z. Ye, S. P. Murphy, J. Zheng, C. G. Chute, and I. J. Kullo, “Discovering Peripheral Arterial Disease Cases from Radiology Notes Using Natural Language Processing,” *in AMIA Annual Symposium Proceedings*, vol. 2010, pp. 722–726, 2010.

[50] D. Hristovski, J. Stare, B. Peterlin, and S. Dzeroski, “Supporting discovery in medicine by association rule mining in Medline and UMLS,” *Studies in health technology and informatics*, pp. 1344–1348, 2001.

[51] P. Gottgtroy, N. Kasabov, and S. MacDonell, “An ontology driven approach for knowledge discovery in Biomedicine,” *Proceedings of the VIII Pacific Rim International Conferences on Artificial Intelligence (PRICAI)*, 2004.

[52] H. Yu and Y. G. Cao, “Using the Weighted Keyword Model to Improve Information Retrieval for Answering Biomedical Questions,” *Summit on translational bioinformatics*, vol. 2009, p. 143, 2009.

[53] S. Tiun, R. Abdullah, and T. E. Kong, “Automatic Topic Identification Using Ontology Hierarchy,” *in Computational Linguistics and Intelligent Text Processing*, pp. 444–453.

[54] R. Berlanga-Llavori, H. Anaya-Sanchez, A. Pons-Porrata, and E. Jiménez-Ruiz, “Conceptual Subtopic Identification in the Medical Domain,” *in Advances in Artificial Intelligence–IBERAMIA*, pp. 312–321.

[55] M. Yetisgen-Yildiz and W. Pratt, “The effect of feature representation on MEDLINE document classification,” *in AMIA Annual Symposium Proceedings*, 2005, vol. 2005, p. 849.

[56] X. Zhou, H. Han, I. Chankai, A. Prestrud, and A. Brooks, “Approaches to text mining for clinical medical records,” *in Proceedings of the 2006 ACM symposium on Applied computing* 2006, p. 235.

[57] A. R. Aronson, “Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program.,” *in Proceedings of the AMIA Symposium*, 2001, p. 17.

[58] D. J. Cook and L. B. Holder, “Graph-based data mining,” *IEEE Intelligent Systems and their Applications*, vol. 15, no. 2, pp. 32–41, Apr. 2000.

[59] J. Tomita, H. Nakawatase, and M. Ishii, “Calculating similarity between texts using graph-based text representation model,” *in Proceedings of the thirteenth ACM international conference on Information and knowledge management*, 2004, pp. 248–249.

[60] Y. Ohsawa, N. E. Benson, and M. Yachida, “KeyGraph: Automatic indexing by co-occurrence graph based on building construction metaphor,” *in Research and Technology Advances in Digital Libraries*, 1998. ADL 98. Proceedings. IEEE International Forum on, 1998, pp. 12–18.

[61] O. Madani and J. Yu, “Discovery of numerous specific topics via term co-occurrence analysis,” *in Proceedings of the 19th ACM international conference on Information and knowledge management*, 2010, pp. 1841–1844.

[62] W. Jin and R. K. Srihari, “Graph-based text representation and knowledge discovery,” *in Proceedings of the 2007 ACM symposium on Applied computing*, 2007, p. 811.

[63] D. Widdows and B. Dorow, “A graph model for unsupervised lexical acquisition,” *in Proceedings of the 19th international conference on Computational linguistics* - vol. 1, 2002, pp. 1–7.

[64] B. Zhou, P. Luo, Y. Xiong, and W. Liu, “Wikipedia-Graph Based Key Concept Extraction towards News Analysis,” *in IEEE Conference on Commerce and Enterprise Computing*, Vienna, Austria, 2009, pp. 121–128.

[65] Z. Minier, Z. Bodo, and L. Csato, “Wikipedia-Based Kernels for Text Categorization,” *in Ninth International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC 2007)*, Timisoara, Romania, 2007, pp. 157–164.

[66] E. Gabrilovich and S. Markovitch, “Computing semantic relatedness using wikipedia-based explicit semantic analysis,” *in Proceedings of the 20th international joint conference on Artifical intelligence*, 2007, pp. 1606–1611.

[67] E. Gabrilovich and S. Markovitch, “Overcoming the brittleness bottleneck using Wikipedia: Enhancing text categorization with encyclopedic knowledge,” *in Proceedings of the National Conference on Artificial Intelligence*, 2006, vol. 21, p. 1301.

[68] X. Hu, X. Zhang, C. Lu, E. Park, and X. Zhou, “Exploiting Wikipedia as external knowledge for document clustering,” *in Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009, pp. 389–396.

[69] J. Hu, L. Fang, Y. Cao, H. J. Zeng, H. Li, Q. Yang, and Z. Chen, “Enhancing text clustering by leveraging Wikipedia semantics,” *in Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Rretrieval*, 2008, pp. 179–186.

[70] P. Wang and C. Domeniconi, “Building semantic kernels for text classification using wikipedia,” *in Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008, pp. 713–721.

[71] G. A. Miller, “WordNet: a lexical database for English,” *Communications of the ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[72] P. Tarau, R. Mihalcea, and E. Figa, “Semantic document engineering with WordNet and PageRank,” *in Proceedings of the 2005 ACM symposium on Applied computing*, 2005, pp. 782–786.

[73] J. Kamps, M. Marx, R. J. Mokken, and M. De Rijke, “Using wordnet to measure semantic orientations of adjectives,” 2004.

[74] B. Jin, B. Muller, C. Zhai, and X. Lu, “Multi-label literature classification based on the Gene Ontology graph,” *BMC Bioinformatics*, vol. 9, pp. 525–525.

[75] G. Liu, A. E. Loraine, R. Shigeta, M. Cline, J. Cheng, V. Valmeekam, S. Sun, D. Kulp, and M. A. Siani-Rose, “NetAffx: Affymetrix probesets and annotations,” *Nucleic Acids Research*, vol. 31, no. 1, pp. 82 –86, Jan. 2003.

[76] K. A. Spackman, P. D, K. E. Campbell, P. D, R. A. Côté, “SNOMED RT: A reference terminology for health care,” *Journal of the American Medical Informatics Association*, pp. 640–644, 1997.

[77] K. E. Campbell and M. A. Musen, “Representation of clinical data using SNOMED III and conceptual graphs.,” *in Proceedings of the Annual Symposium on Computer Application in Medical Care, American Medical Informatics Association*, pp. 354–358, 1992.

[78] S. B. Johnson, A. Aguirre, P. Peng, and J. Cimino, “Interpreting natural language queries using the UMLS.,” *in Proceedings of the Annual Symposium on Computer Application in Medical Care,* 1993, p. 294.

[79] M. Joubert, M. Fieschi, and J. J. Robert, “A conceptual model for information retrieval with UMLS.,” *in Proceedings of the Annual Symposium on Computer Application in Medical Care*, 1993, p. 715.

[80] J. J. Robert, M. Joubert, L. Nal, and M. Fieschi, “A computational model of information retrieval with UMLS.,” *in Proceedings of the Annual Symposium on Computer Application in Medical Care*, 1994, p. 167.

[81] J. F. Sowa, “Conceptual graphs for a data base interface,” *IBM Journal of Research and Development*, vol. 20, no. 4, pp. 336–357, 1976.

[82] J. F. Sowa, “Conceptual structures: information processing in mind and machine,” 1983.

[83] P. H. . Nguyen and D. Corbett, “A basic mathematical framework for conceptual graphs,” *IEEE Transactions on Knowledge and Data Engineering*, pp. 261–271, 2006.

[84] M. Chein and M. L. Mugnier, “Conceptual graphs: Fundamental notions,” *in Revue d’intelligence artificielle*, 1992.

[85] A. Smalter, J. Huan, and G. Lushington, “GPM: A graph pattern matching kernel with diffusion for chemical compound classification,” *in 8th IEEE International Conference on BioInformatics and BioEngineering*, 2008. BIBE 2008, 2008, pp. 1–6.

[86] S. Berretti, A. Del Bimbo, and E. Vicario, “Efficient matching and indexing of graph models in content-based retrieval,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 10, p. 1089, 2001.

[87] M. Kuramochi and G. Karypis, “Frequent subgraph discovery,” *in Proceedings 2001 IEEE International Conference on Data Mining*, San Jose, CA, USA, pp. 313–320.

[88] L. B. Holder, D. J. Cook, and S. Djoko, “Substructure discovery in the subdue system,” *in Proceedings of the Workshop on Knowledge Discovery in Databases*, 1994, pp. 169–180.

[89] X. Yan, P. S. Yu, and J. Han, “Graph indexing: A frequent structure-based approach,” *in Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, 2004, pp. 335–346.

[90] N. Wale, I. A. Watson, and G. Karypis, “Comparison of descriptor spaces for chemical compound retrieval and classification,” *Knowledge and Information Systems*, vol. 14, no. 3, pp. 347–375, 2008.

[91] B. Berendt, “Using and learning semantics in frequent subgraph mining,” *Advances in Web Mining and Web Usage Analysis*, pp. 18–38.

[92] J. Huan, W. Wang, A. Washington, J. Prins, R. Shah, and A. Tropsha, “Accurate classification of protein structural families using coherent subgraph analysis,” *in Pacific Symposium on Biocomputing 2004*: Hawaii, USA, 6-10 January 2004, 2003, p. 411.

[93] M. Lahiri and T. Y. Berger-Wolf, “Structure prediction in temporal networks using frequent subgraphs,” *in IEEE Symposium on Computational Intelligence and Data Mining*, 2007. CIDM 2007, 2007, pp. 35–42.

[94] N. Nagamine and Y. Sakakibara, “Statistical prediction of protein chemical interactions based on chemical structure and mass spectrometry data,” *Bioinformatics*, vol. 23, no. 15, p. 2004, 2007.

[95] L. Getoor and C. P. Diehl, “Link mining: a survey,” *ACM SIGKDD Explorations Newsletter*, vol. 7, no. 2, p. 12, 2005.

[96] R. Feldman, “Link analysis: Current state of the art,” *KDD-02 Tutorial*, 2002.

[97] L. Page, S. Brin, R. Motwani, and T. Winograd, “The PageRank citation ranking: Bringing order to the web.,” 1999.

[98] J. M. Kleinberg, “Authoritative sources in a hyperlinked environment,” *Journal of the ACM (JACM)*, vol. 46, no. 5, pp. 604–632, 1999.

[99] A. W. Wolfe, “Social network analysis: Methods and applications,” *American Ethnologist*, vol. 24, no. 1, pp. 219–220, 1997.

[100] D. Liben-Nowell and J. Kleinberg, “The link-prediction problem for social networks,” *Journal of the American society for information science and technology*, vol. 58, no. 7, pp. 1019–1031, 2007.

[101] A. Popescul, R. Popescul, and L. H. Ungar, “Statistical Relational Learning for Link Prediction,” 2003.

[102] S. Bleik, W. Xiong, Y. Wang, and M. Song, “Biomedical concept extraction using concept graphs and ontology-based mapping,” *in Bioinformatics and Biomedicine (BIBM), 2010 IEEE International Conference on*, 2010, pp. 553–556.

[103] I. H. Witten, G. W. Paynter, E. Frank, C. Gutwin, and C. G. Nevill-Manning, “KEA: Practical automatic keyphrase extraction,” *in Proceedings of the fourth ACM conference on Digital libraries*, 1999, p. 255.

[104] G. Salton and C. Buckley, “Term-weighting approaches in automatic text retrieval,” *Information processing & management*, vol. 24, no. 5, pp. 513–523, 1988.

[105] M. Damashek, “Gauging similarity with n-grams: Language-independent categorization of text,” *Science*, vol. 267, no. 5199, pp. 843–848, 1995.

[106] W. B. Cavnar and J. M. Trenkle, “N-gram-based text categorization,” *Ann Arbor MI*, vol. 48113, no. 2, pp. 161–175, 1994.

[107] Y. Gong and X. Liu, “Generic text summarization using relevance measure and latent semantic analysis,” *in Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, 2001, pp. 19–25.

[108] S. Shehata, F. Karray, and M. Kamel, “A concept-based model for enhancing text categorization,” *in Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2007, pp. 629–637.

[109] T. Andreasen, H. Bulskov, P. Jensen, and T. Lassen, “Conceptual indexing of text using ontologies and lexical resources,” *Flexible Query Answering Systems*, pp. 323–332, 2009.

[110] G. Ercan and I. Cicekli, “Using lexical chains for keyword extraction,” Information *Processing and Management*, vol. 43, no. 6, pp. 1705–1714, 2007.

[111] C. Huang, Y. Tian, Z. Zhou, C. X. Ling, and T. Huang, “Keyphrase extraction using semantic networks structure analysis,” *in Data Mining*, 2006. ICDM’06. Sixth International Conference on, 2007, pp. 275–284.

[112] P. D. Turney, P. Pantel, and others, “From frequency to meaning: Vector space models of semantics,” *Journal of Artificial Intelligence Research*, vol. 37, no. 1, pp. 141–188, 2010.

[113] Y. M. Chen, X. L. Wang, and B. Q. Liu, “Multi-document summarization based on lexical chains,” *in Proceedings of the 2005 international conference on machine learning and cybernetics*, 2005, pp. 1937–1942.

[114] X. Wan, J. Yang, and J. Xiao, “Towards an iterative reinforcement approach for simultaneous document summarization and keyword extraction,” *in Annual meeting-association for computational linguistics*, 2007, vol. 45, p. 552.

[115] S. Bleik, M. Song, A. Smalter, J. Huan, and G. Lushington, “CGM: A biomedical text categorization approach using concept graph mining,” *in Bioinformatics and Biomedicine Workshop, 2009. BIBMW 2009. IEEE International Conference on*, 2009, pp. 38–43.

[116] Z. Elberrichi, A. Rahmoun, and M. A. Bentaalah, “Using WordNet for Text Categorization.” *The International Arab Journal of Information Technology*, vol. 5(1), pp. 16-24

[117] M. Janik and K. J. Kochut, “Wikipedia in action: Ontological knowledge in text categorization,” *in International Conference on Semantic Computing*, 0, 2008, pp. 268–275.

[118] F. M. Suchanek, G. Kasneci, and G. Weikum, “Yago: A large ontology from wikipedia and wordnet,” *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 6, no. 3, pp. 203–217, 2008.

[119] F. Sebastiani, “Machine learning in automated text categorization,” *ACM computing surveys (CSUR)*, vol. 34, no. 1, pp. 1–47, 2002.

[120] M. E. Maron, “Automatic indexing: an experimental inquiry,” *Journal of the ACM (JACM)*, vol. 8, no. 3, pp. 404–417, 1961.

[121] B. V. Dasarathy, “Nearest Neighbor ({NN}) Norms:{NN} Pattern Classification Techniques,” 1991.

[122] T. Joachims, “Text categorization with support vector machines: Learning with many relevant features,” *Machine Learning: ECML-98*, pp. 137–142, 1998.

[123] C. Apté, F. Damerau, and S. M. Weiss, “Automated learning of decision rules for text categorization,” *ACM Transactions on Information Systems (TOIS)*, vol. 12, no. 3, pp. 233–251, 1994.

[124] W. Lehnert, S. Soderland, D. Aronow, F. Feng, and A. Shmueli, “Inductive text classification for medical applications,” *Journal of Experimental and Theoretical Artificial Intelligence*, vol. 7, pp. 49–49, 1995.

[125] W. Mao and W. W. Chu, “Free-text medical document retrieval via phrase-based vector space model.,” *in Proceedings of the AMIA Symposium*, 2002, p. 489.

[126] A. Wilcox, G. Hripcsak, and C. Friedman, “Using knowledge sources to improve classification of medical text reports,” *in KDD-2000 Workshop on Text Mining*, 2000.

[127] R. Angelova and G. Weikum, “Graph-based text classification: learn from your neighbors,” *in Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, 2006, p. 492.

[128] C. Jiang, F. Coenen, R. Sanderson, and M. Zito, “Text Classification using Graph Mining-based Feature Extraction,” *Research and Development in Intelligent Systems XXVI*, pp. 21–34.

[129] M. Arey and S. Chakravarthy, “InfoSift: Adapting Graph Mining Techniques for Text Classification,” *in Proceedings of the Eighteenth International FLAIRS Conference*, 2005.

[130] K. R. Gee and D. J. Cook, “Text Classification Using Graph-Encoded Linguistic Elements,” *in Proceedings of the 18th International FLAIRS Conference* , 2005.

[131] A. McCallum and K. Nigam, “A comparison of event models for naive bayes text classification,” *in AAAI-98 workshop on learning for text categorization*, 1998, vol. 752, pp. 41–48.

[132] Y. Yang and X. Liu, “A re-examination of text categorization methods,” *in Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 1999, pp. 42–49.

[133] F. Sebastiani, “A tutorial on automated text categorisation,” *in Proceedings of ASAI-99, 1st Argentinian Symposium on Artificial Intelligence*, 1999, pp. 7–35.

[134] D. Lewis, “Naive (Bayes) at forty: The independence assumption in information retrieval,” *Machine Learning: ECML-98*, pp. 4–15, 1998.

[135] M. Mishra, J. Huan, S. Bleik, and M. Song, “Biomedical text categorization with concept graph representations using a controlled vocabulary,” *in Proceedings of the 11th International Workshop on Data Mining in Bioinformatics*, New York, NY, USA, 2012, pp. 26–32.

[136] K. M. Borgwardt and H. P. Kriegel, “Shortest-path kernels on graphs,” *Data Mining, Fifth IEEE International Conference on.*, 2005.

[137] H. Kashima, K. Tsuda, and A. Inokuchi, “Marginalized kernels between labeled graphs,” *in Proceedings of the Twentieth International Conference on Machine Learning* -, 2003, vol. 20, p. 321.

[138] C. Leslie, E. Eskin, and W. S. Noble, “The spectrum kernel: A string kernel for SVM protein classification,” *in Proceedings of the Pacific Symposium on Biocomputing*, 2002, vol. 7, pp. 566–575.

[139] T. Horvath, T. Gartner, and S. Wrobel, “Cyclic pattern kernels for predictive graph mining,” *in Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004, pp. 158–167.

[140] P. Mahé and J. P. Vert, “Graph kernels based on tree patterns for molecules,” *Machine learning*, vol. 75, no. 1, pp. 3–35, 2009.

[141] J. Huan, D. Bandyopadhyay, J. Prins, J. Snoeyink, A. Tropsha, and W. Wang, “Distance-based identification of structure motifs in proteins using constrained frequent subgraph mining,” *in Computational systems bioinformatics: CSB2006 conference proceedings*, Stanford CA, 14-18 August 2006, 2006, vol. 4, p. 227.

[142] H. Frohlich, J. K. Wegner, F. Sieker, and A. Zell, “Optimal assignment kernels for attributed molecular graphs,” *in Proceedings of the 22nd international conference on Machine learning*, 2005, pp. 225–232.

[143] H. Lodhi, C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins, “Text classification using string kernels,” *The Journal of Machine Learning Research*, vol. 2, pp. 419–444, 2002.

[144] J. C. Gower, “A general coefficient of similarity and some of its properties,” *Biometrics*, pp. 857–871, 1971.

[145] P. Baldi and L. Ralaivola, “Graph kernels for molecular classification and prediction of mutagenicity, toxicity, and anticancer activity,” *in Computational Biology Workshop of Neural Information Processing Systems*, NIPS2004, 2004.

[146] C. Gutwin, G. Paynter, I. Witten, C. Nevill-Manning, and E. Frank, “Improving browsing in digital libraries with keyphrase indexes,” *Decision Support Systems*, vol. 27, no. 1–2, pp. 81–104, 1999.

[147] S. Jones and M. Mahoui, “Hierarchical document clustering using automatically extracted keyphrases,” 2000.

[148] J. C. Denny, J. D. Smithers, and others, “A new tool to identify key biomedical concepts in text documents, with special application to curriculum content,” *in Proceedings of the AMIA Symposium*, 2002, p. 1007.

[149] A. L. Berger and V. O. Mittal, "OCELOT: a system for summarizing Web pages," *in Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, 2000, pp. 144–151.

[150] H. P. Luhn, "The automatic creation of literature abstracts," *IBM Journal of research and development*, vol. 2, no. 2, pp. 159–165, 2010.

[151] K. Chen, "Topic identification in discourse," *in Proceedings of the Seventh Meeting of the European Association for Computational Linguistics*, 1995, pp. 267–271.

[152] P. M. Ramirez and C. Mattmann, "ACE: improving search engines via Automatic Concept Extraction," *in Proceedings of the 2004 IEEE International Conference on Information Reuse and Integration*, 2004, pp. 229–234.

[153] Q. Mei, X. Shen, and C. X. Zhai, "Automatic labeling of multinomial topic models," *in Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2007, p. 499.

[154] S. Jones and G. W. Paynter, "Human evaluation of Kea, an automatic keyphrasing system," *in Proceedings of the 1st ACM/IEEE-CS joint conference on Digital libraries*, 2001, p. 156.

[155] W. Pratt and M. Yetisgen-Yildiz, "A study of biomedical concept identification: MetaMap vs. people," 2003.

[156] L. V. Subramaniam, S. Mukherjea, P. Kankar, B. Srivastava, V. S. Batra, P. V. Kamesam, and R. Kothari, "Information extraction from biomedical literature: methodology, evaluation and an application," *in Proceedings of the twelfth international conference on Information and knowledge management*, 2003, p. 417.

[157] S. L. Kanter, R. A. Miller, M. Tan, and J. Schwartz, "Using POSTDOC to recognize biomedical concepts in medical school curricular documents.*," Bulletin of the Medical Library Association*, vol. 82, no. 3, p. 283, 1994.

[158] X. Zhou, X. Zhang, and X. Hu, "MaxMatcher: Biological concept extraction using approximate dictionary lookup," *PRICAI 2006: Trends in Artificial Intelligence*, pp. 1145–1149, 2006.

[159] W. H. Majoros, G. M. Subramanian, and M. D. Yandell, "Identification of key concepts in biomedical literature using a modified Markov heuristic," *Bioinformatics*, vol. 19, no. 3, p. 402, 2003.

[160] A. Hulth, “Improved automatic keyword extraction given more linguistic knowledge,” *in Proceedings of the 2003 conference on Empirical methods in natural language processing*, 2003, vol. 10, pp. 216–223.

[161] T. Andreasen, H. Bulskov, T. Lassen, S. Zambach, P. A. Jensen, B. N. Madsen, H. E. Thomsen, J. F. Nilsson, and B. A. Szymczak, “SIABO: Semantic Information Access through Biomedical Ontologies,” *in International Conference on Knowledge Engeneering and Ontology*, Madeira, Portugal, 2009.

[162] D. Crow and J. DeSanto, “A hybrid approach to concept extraction and recognition-based matching in the domain of human resources,” *in Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence*, 2004, p. 539.

[163] O. Medelyan and I. H. Witten, “Thesaurus based automatic keyphrase indexing,” *in Digital Libraries, 2006. JCDL’06. Proceedings of the 6th ACM/IEEE-CS Joint Conference on*, 2006, pp. 296–297.

[164] M. Sanderson and B. Croft, “Deriving concept hierarchies from text,” *in Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 1999, pp. 206–213.

[165] G. Salton, “Developments in automatic text retrieval,” *Science*, vol. 253, no. 5023, p. 974, 1991.

[166] “LingPipe: String Comparison and String Distance Tutorial.” [Online]. Available: http://alias-i.com/lingpipe/demos/tutorial/stringCompare/read-me.html. [Accessed: 23-Mar-2013].

[167] T. Joachims, “Training linear SVMs in linear time,” *in Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2006, pp. 217–226.

[168] V. N. Vapnik, The nature of statistical learning theory. *Springer Verlag*, 2000.

[169] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[170] T. Joachims, “Optimizing search engines using clickthrough data,” *in Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2002, pp. 133–142.

[171] “Kea.” [Online]. Available: http://www.nzdl.org/Kea/. [Accessed: 23-Mar-2013].

[172] W. B. Croft, H. R. Turtle, and D. D. Lewis, “The use of phrases and structured queries in information retrieval,” *in Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval*, 1991, pp. 32–45.

[173] J. Dean and S. Ghemawat, “MapReduce: simplified data processing on large clusters,” *Communications of the ACM*, vol. 51, no. 1, pp. 107–113, 2008.