



FAKULTÄT INFORMATIK

EMCL Master Thesis

**Analysis of a machine learning algorithm and
corpus as a tool for managing the ambiguity
problem of search engines**

by

Natalia Macari

born on August 20th 1985, Chisinau, Moldova

Supervisor : **Prof.Dr. Michael Schroeder**

Advisors : **Dipl.-Inf. Heiko Dietze,**

Dr. Georgios Tsatsaronis

Technische Universität Dresden

Author: **Natalia Macari**
Matrikel-Nr.: **3553346**
Title: **Analysis of a machine learning algorithm
and corpus as a tool for managing the am-
biguity problem of search engines**
Degree: **Master of Science**
Date of Submission: **16/12/2010**

Declaration

Hereby I certify that the thesis has been written by me. Any help that I have received in my research work has been acknowledged. Additionally, I certify that I have not used any auxiliary sources and literature except those I cited in the thesis.

Signature of Author

Acknowledgments

The completion of this thesis is a result of cooperation, encouragement and support of number of people to whom I am greatly indebted.

First of all, I would like to give my great thanks to my supervisor Prof. Michael Schroeder for his supervision of this thesis and for giving me the opportunity to work in his research group.

I also wish to thank my advisors Dr. Heiko Dietze and Dr. George Tsatsaronis for their assistance, recommendations and suggestions. Their guidance has been very helpful throughout the work.

I would like to give a special thank to all Transinsight Gmbh members for warm atmosphere and for considering me as a part of their team. I sincerely thank Dr. Michael Alvers for reading this thesis and making useful comments for its improvement.

Special thanks should be given to all my professors in TUW and TUD for teaching and sharing their knowledge. Special thanks go to Prof. Steffen Hölldobler for giving the opportunity to receive European double master degree and to Prof. Alexander Leitsch for supporting my education in TUW.

Furthermore, I would like to thank all my friends from Vienna and Dresden for being my family, for supporting me, for sharing memorable moments with me. I want to thank my close friends from Moldova for keeping in touch these years through the Internet. I heartily thank Marco Gario and Irina Kaunenko for reading this thesis, giving useful comments and helping correcting my English.

The last, and surely the most, I want to thank my family and my beloved husband for their love and patience, for encouragement and understanding me.

This thesis is supported by Erasmus Mundus scholarship financial grant.

Abstract

The increasing number of scientific literature on the Internet and the absence of efficient tools used for its classification, structuring and setting the relationships among the articles influence the speed of the search and the quality of the result. The usage of ontologies - hierarchical structured controlled vocabularies - makes it possible to process information at the semantic level, which greatly improves the search for the relevant information. However, an imprecise query statement does not eliminate the possibility of ambiguity appearance in the search results.

The master thesis explores one of the possible ways to handle the ambiguity problem of search engines. The application is the concept-recognition to identify ontology concepts from text. The goal is to analyze the performance of the machine learning algorithm approach. The first step is to examine the PubMed corpus with the given training data in form of MeSH hand-annotations. By now, PubMed database contains over 20 million biomedical abstracts. Moreover, the MeSH vocabulary contains more than 25,000 biological terms. Based on the obtained results, experiments are conducted to provide input for the classification of concepts and textual labels as non-ambiguous or ambiguous.

The concept recognition approach, developed by Doms (2008), annotates the PubMed articles with the Medical Subject Headings (MeSH) concept labels. The approach uses maximum entropy models to identify the ambiguous biomedical terms and distinguish their meanings. The evaluation of the algorithm performance showed high results, namely 91% f-score (precision 90%, recall 92%) with accuracy threshold value of 0.6. The evaluation was done on the training data with the optimal size of 5000 documents, which is sufficient for the good prediction ability of the classifier. The term features title, abstract and year turned out to be important for the classification process, while feature journal brings insufficient improvement to the algorithm's performance. The negative training data – random selected articles with literal occurrence of the explored term – decreases the risk of overfitting and overestimation of the model. The share of ambiguous terms is 6.4% based on WordNet and Wikipedia results. The additional biomedical thesauri UMLS should be applied together with the used lexica to determine the ambiguity level of the biomedical terms and to correlate it with the algorithm's performance.

Contents

1	Introduction	10
1.1	Motivation	10
1.2	Organization of the Thesis	11
2	Background	12
2.1	Ontologies	12
2.1.1	Gene Ontology	12
2.1.2	Medical Subject Headings	13
2.1.3	WordNet	13
2.1.4	Wikipedia	14
2.2	Biomedical Search Engines	14
2.2.1	PubMed	14
2.2.2	GoPubMed	15
2.3	Machine Learning Techniques	16
2.4	Evaluation Measurements	20
2.5	Validation Techniques	23
2.6	Accuracy Threshold	24
2.7	Statistical measurements	24
2.7.1	Median	25
2.7.2	Average	25
2.7.3	Standard Deviation	25
2.7.4	Correlation Coefficient	26
2.8	Semantic Similarity Metrics	27
2.9	Sense of Ambiguity	29
3	Experiment	31
3.1	Examined hypotheses	31
3.2	Growth of PubMed and MeSH databases	32
3.3	General behavior of the algorithm	33
3.3.1	MeSH Terms	33
3.3.2	Classification Delta	35
3.3.3	Training data size	37
3.3.4	Feature Vectors	39
3.3.5	Negative Training Data	46
3.3.6	Re-Examination of Negative Training Data	48
3.4	Application for disambiguation	50
3.4.1	PubMed and GoPubMed vs Yahoo	52
3.4.2	Term occurrence in MeSH hand annotated documents	57
3.4.3	Term recognition by GoPubMed	58
3.4.4	WordNet and Wikipedia	61

4 Conclusion	65
4.1 Future Work	65

List of Figures

2.1	Mesh Tree Structure 2010. First level of MeSH Headings.	13
2.2	Growth of PubMed database through 1965 – 2010 years. Blue line represents numerical growth of articles added to PubMed. Red line represents exponential trend.	15
2.3	Example of hidden Markov model with x - hidden states, y - possible outputs, a - state transition probabilities, b - output probabilities	19
2.4	Example of (a) the precision-recall curve and (b) the ROC curve.	22
2.5	Example of calculating the distance between MeSH terms	29
2.6	Ambiguous MeSH terms association, serotonin and sepsis and their position in MeSH tree.	30
3.1	Growth of PubMed and MeSH hand annotations per year: (a) together with the results of previous years, (b) separately from the results of previous years.	33
3.2	Algorithm behavior according to different thresholds ($\delta = [0, 0.05, 0.1, 0.2, 0.3, 0.4]$). The performance is shown with averaged precision, recall and f-score. (a) - visualization of the algorithm behavior, (b) - visualization of the manually checked algorithm behavior, (c) - statistical information for the algorithm performance, (d) - statistical information for the manually checked algorithm performance.	36
3.3	Performance of the algorithm on (a) 10 MeSH terms, (b) 4078 MeSH terms and (c) 4078 MeSH terms without min f-score values for terms with more than 5000 documents.	38
3.4	Features combinations	44
3.5	Trend line of publications over the time for (a) Tricuspid Atresia and (b) omega-Agatoxin IVA.	45
3.6	Analysis of precision, recall and f-score for classification delta in range $[0, 0.05, 0.1, 0.2, 0.3, 0.4]$ for new set of random selected negative documents: (a) - visualization of average values, (b) - statistical information for average values	49
3.7	Features combinations	50
3.8	Correlation between the number of term occurrences in Yahoo and the number of documents where the term appears according to (a) PubMed and (c) GoPubMed. Axis X corresponds to the PubMed / GoPubMed extracted documents, axis Y - to the Yahoo. Both axes are log scaled. Medians are shown as red lines. Statistical measurements for (b) PubMed vs Yahoo and (d) GoPubMed vs Yahoo are calculated in the number of documents.	53
3.9	Combination of number of senses in WordNet and Wikipedia in range 0, 1.	54
3.10	Analysis of terms with more than one sense in WordNet and Wikipedia. These terms are colored in black color.	55

3.11	Correlation between number of MeSH hand annotated documents in PubMed and GoPubMed: (a) graphical visualization of the experiment. Axis X corresponds to the PubMed extracted documents, axis Y - to the GoPubMed. Medians are shown as red lines. (b) statistical measurements of PubMed and GoPubMed calculated in number of documents.	58
3.12	Correlation between PubMed and GoPubMed in MeSH hand annotated documents, and statistical measurements for it.	59
3.13	The correlation between MeSH hand annotated documents where term appears in title and abstract and MeSH hand annotated documents recognized by GPM algorithm (Figure 3.12) in range [0,15000].	60
3.14	Visualization of number of MeSH term-meanings in WordNet and Wikipedia. . .	62
3.15	Agreement between Wikipedia and WordNet in the number of senses extracted for each MeSH term.	63

List of Tables

2.1	Comparison of machine learning algorithms	21
2.2	Confusion Matrix	22
3.1	MeSH Terms	34
3.2	Example of the positive feature vector for term "Gallstones"	39
3.3	Proportion of term features in context model.	41
3.4	Top 10 positive features of the terms	43
3.5	Top 10 negative features of the terms	43
3.6	Average F-score measured for 4 options of selection negative examples.	47
3.7	Training and testing model with different combination of negative options. The results are given percentagewise.	48
3.8	Top 10 positive features of the terms for new random selected negative documents	51
3.9	Top 10 negative features of the terms for new random selected negative documents	51
3.10	An example of terms with more than one sense in WordNet and/or Wikipedia. Information about the number of documents in PubMed, GoPubMed, Yahoo and the number of senses in WordNet and Wikipedia is also provided.	56
3.11	An example of the terms-outliers. Information about the number of documents in PubMed, GoPubMed, Yahoo and the number of senses in WordNet and Wikipedia is provided.	57
3.12	An example of terms that are outliers from Figure 3.13 and stated reason why they are considered to be outliers. Information about the number of documents found in PubMed and GoPubMed is provided.	60

Chapter 1

Introduction

1.1 Motivation

With the rapid expansion of the Internet as a source of scientific and educational literature, the search for the specific data among the large amount of information became a difficult and time consuming process. The current state of the Internet can be characterized by weakly structured data and, in particular, the absence of relationships between data. Today's search engines, such as Google and Yahoo, offer a simple keyword-based search, which leads to a quality reduction of information processing and, may therefore generates search results that do not match to the search criteria. Besides, the information is processed at the syntactic level, i.e. without taking into account properties such as synonymy, polysemy and homonymy. The choice of relevant data from a large amount of information and their interpretation are the responsibility of the user. Thus, finding the necessary information among the semi-or unstructured heterogeneous data is a challenging task.

Recently, ontologies, which are commonly shared, explicitly defined, generic conceptualizations (Gruber, 1993), are wildly used for annotating the objects of interest with the ontological concepts. The usage of ontologies provides a content-based access to the data, which makes it possible to process information at the semantic level and significantly improves the search of relevant documents. GoPubMed is a semantic search engine that uses ontologies: the Gene Ontology (GO) and the Medical Subject Headings (MeSH) as background knowledge for indexing biomedical literature. GoPubMed explores the PubMed database, the most extensive literature database in biomedical domain that comprises over 20 millions biomedical abstracts as of today. The combined application of the GO and MeSH provides the best coverage of ontological concepts in the PubMed abstracts, and returns the relevant results to the user for further usage (Doms, 2008). GoPubMed uses various text mining techniques and algorithms (stemming, tokenization, synonym detection) to identify relevant ontology concepts in the PubMed abstracts.

Due to the presence of ambiguous concepts, the classification process of relevant documents in the literature search becomes an even more challenging task. Biomedical terms can have a very specific meaning in biomedical domain, but mean other things in other contexts. For example, biomedical terms "head" and "back" are hard to disambiguate due to their resemblance to common English words. Moreover, ambiguity arises from identical abbreviations for different concepts. The concept recognition approach used to recognize the appropriate MeSH concepts in the PubMed abstracts has been designed and described in Doms (2008). The approach is based on the maximum entropy models that are used to disambiguate the MeSH concepts in PubMed articles. For each MeSH concept the algorithm creates the context models that characterize the term and that consist of the lexical tokens taken from related PubMed articles.

The aim of this thesis is to analyze and estimate the performance of the concept recognition algorithm with regard to the ambiguous biomedical concepts. The analysis comprises two main tasks that are described, studied and examined in detail. These tasks are formulated in a list of

hypothesis that will be explored in the Section 3.

The first part of the thesis concentrates on the analysis of the general performance of the concept recognition algorithm. The goal of this part is to explore and evaluate the correctness of the algorithm's performance. The problems about choosing the sufficient accuracy threshold, optimal size of training dataset, important term features and choice of negative dataset will be solved. Most of the experiments, which will be done in this part, improve the work done in Macari (2010).

The second part will consider the analysis of algorithm performance in relation to the ambiguous terms. The goal of this part is to explore the MeSH terms and detect the ambiguous concepts. Ambiguity of terms is researched in relation to their occurrences in PubMed, GoPubMed, Yahoo and the number of senses in WordNet and Wikipedia. Besides, the term occurrence in the MeSH hand annotated document and the ability of GoPubMed to recognize relevant articles for a given MeSH term will be discussed.

1.2 Organization of the Thesis

This thesis is organized in 4 chapters:

- **Introduction:** This chapter gives to the reader a brief introduction to our research.
- **Background:** This chapter contains all background knowledge related to our topic such as ontologies, machine learning techniques, evaluation and validation techniques, similarity metrics and ambiguity sense explanation.
- **Experiment:** This chapter describes the experimental part of our work. A set of hypotheses were assigned and examined during the experiment. This chapter is split into two major parts. Primarily, the general behavior of the algorithm is explored. Secondly, we evaluate the algorithm in relation to the ambiguous terms.
- **Conclusion:** As usual, this chapter summarizes the results received during our work, indicates the advantages and disadvantages, and discusses about future researches.

Chapter 2

Background

This chapter provides relevant background knowledge that is related to our topic such as definition of the ontology and its types, an introduction to biomedical search engines, a list of machine learning methods, techniques for evaluation and validation, statistical measurements and similarity metrics. Furthermore, since our research is about the ambiguity problem of search engines in application to biomedical domain, the sense of term *ambiguity* is discussed in the last section. The definition of phenomenon is given according to the Merriam Webster English Dictionary, linguistics and biomedical domain.

2.1 Ontologies

The term *ontology* has its origin in philosophy and is defined as study of being, existence or reality in general (Hofweber, 2010). Ontology deals with entities, which can be grouped according to similarities and differences, and, as a result, form a hierarchy. In the context of information science ontology is defined as "formal, explicit specification of a shared conceptualization" (Gruber, 1993). A conceptualization consists of the objects, concepts, their properties, and the relationships that connect concepts that exist in the area of interest. This set of concepts and relationships between them form a representational controlled vocabulary of the knowledge of a specific domain.

2.1.1 Gene Ontology

The *Gene Ontology*¹ (GO) project, started in 1998, is a joint project of three model organism databases - the FlyBase², the Mouse Genome Informatics³, and the Saccharomyces Genome Database⁴. The goal of the GO Consortium was to produce a structured, dynamic, controlled vocabulary for describing the gene roles and gene products in any organism (Ashburner et al., 2000). Since that time, many databases joined the GO Consortium including repositories for plant, animal and microbial genomes. By January 2008, GO contained over 25,000 biological terms.

GO consists of three controlled hierarchical vocabularies related to the genetic domain: molecular function, biological process and cellular component. *Molecular function* describes activities of individual gene products that can occur at the molecular level. For example, there is catalytic activity, transporter activity, or binding activity. *Biological Process* represents a collection of molecular functions or events. For example, there is cellular physiological process and alpha-glucoside transport. *Cellular component* describes the location of gene products in a cell or extra-cellular region. For example, ribosome or nuclear membrane.

¹<http://www.geneontology.org/>

²<http://flybase.org/>

³<http://www.informatics.jax.org/>

⁴<http://www.yeastgenome.org/>

Mesh Heading
Anatomy
Organisms
Diseases
Chemicals and Drugs
Analytical, Diagnostic and Therapeutic Techniques and Equipment
Psychiatry and Psychology
Phenomena and Processes
Disciplines and Occupations
Anthropology, Education, Sociology and Social Phenomena
Technology, Industry, Agriculture
Humanities
Information Science
Named Groups
Health Care
Publication Characteristics
Geographicals

Figure 2.1: Mesh Tree Structure 2010. First level of MeSH Headings.

GO is structured in a form of Directed Acyclic Graph. The vertexes correspond to biological terms and the directed edges between terms indicate the hierarchical relations between the terms. There are three types of relation *is_a* relation, *part_of* relation and *regulates* relation.

2.1.2 Medical Subject Headings

The *Medical Subject Headings*⁵ (MeSH) thesaurus is a controlled vocabulary of medical and biological terms produced by the National Library of Medicine and is used since 1960. MeSH is mainly used to annotate the conceptual content of biomedical articles in the PubMed literature database (Section 2.2.1). In the version of 2010, MeSH includes nearly 25,000 main headings. The structure of the MeSH is represented in a hierarchical tree. There are 16 different categories (Figure 2.1) that compose the most general level of the MeSH tree. Each category is divided into subcategories and so on. Each MeSH term appears at least once in the MeSH categories related to it. The relationships between the MeSH terms are defined as *broader than* and *narrower than* relations.

2.1.3 WordNet

*WordNet*⁶ is an open-source English scientific thesaurus that was created by psychologists and linguists from Princeton University. The main contribution of WordNet is the conceptual search, as in contrast to usual dictionaries with alphabetic index. Furthermore, WordNet distinguishes between nouns, verbs, adjectives and adverbs since they are characterized by different grammatical rules. It organizes nouns, verbs, adjectives and adverbs into synonym sets so called synsets, i.e. sets of semantically equivalent words. Moreover, for each word WordNet provides information not only about synonymy, but also about antonymy (a word with opposite meaning to the given word, for example, "poor" is antonym for "rich"), hyponymy (a word with more specific meaning of the given word, for example, "potato" is a hyponym of "vegetable"), hypernymy (a word with more general meaning of the given word, for example, "vegetable" is hypernym of

⁵<http://www.nlm.nih.gov/mesh/>

⁶<http://www.wordnet.princeton.edu>

"potato"), meronymy (word that denotes a constituent part of a given word, for example, "finger" is a meronym of "hand") (Miller et al., 1990). Thus, WordNet fits more to the thesaurus definition than dictionary's. As of 2010, the WordNet database contains more than 150,000 words organized in 120,000 synsets.

2.1.4 Wikipedia

*Wikipedia*⁷ is a free, multilingual, web-based encyclopedia. It is based on the volunteer work of a large number of users that can collaboratively edit articles or create new ones. Wikipedia grows exponentially in its size and, by now, it is the largest collection of freely available knowledge. Currently, Wikipedia contains more than 3 million English articles. Furthermore, Wikipedia provides disambiguation pages, each of which represents a list of the possible meanings for a given word.

Due to the complex structure of Wikipedia's page and its content, and to allow a rapid exploration, the Wikipedia Miner Toolkit⁸ can be used to access and extract the meanings of a given term from the Wikipedia's content.

Note that among the specified ontologies only the Gene Ontology fits the ontology definition. The others provide information that is structured on different levels of abstraction and formalization. The most informal one is Wikipedia since it is formed by volunteers. WordNet represents a lexical database that is not specialized in any particular domains, especially in biomedical domain. It doesn't cover most concepts from gene product symbols and cellular components (Bodenreider et al., 2003). Despite the hierarchical structure of the MeSH tree, the relations between the terms are loosely defined with the relationships *narrower than* and *broader than* that are not as explicit as *is_a* and *part_of* relations defined in GO (Bresell, 2002).

2.2 Biomedical Search Engines

The biomedical literature grows at a tremendous rate. As a result, finding relevant literature is an important and difficult problem. There are a number of search engines that process biomedical databases and return information to the user so he/she can further use it (Doms, 2008). In this paper we will focus on PubMed and GoPubMed search engines.

2.2.1 PubMed

*PubMed*⁹ is the most widely used free literature database specialized in the biomedical field. It is developed by the National Center for Biotechnology Information at the National Library of Medicine. The size of PubMed database increases exponentially and, by now, PubMed database contains over 20 million biomedical abstracts. Figure 2.2 shows the growth of PubMed since middle 1960's. PubMed increased the total number of publications through 1965 – 2005 with doubling time of 20 years approximately (Biglu, 2007). The trend can be divided into two parts. The first part, from 1965 to 1999, corresponds to a period when the total number of publications in PubMed increased exponentially. The second part exhibits a linear trend from 1999 till 2009. The analysis of the trend line shows that the number of publications in PubMed, which will be published in 2030 year, will reach 1,5 million.

Each PubMed article contains the title, abstract, affiliated institution, authors, publication date, MeSH headings, and other information. Furthermore, many PubMed articles have web-links to the full-text articles at the publishers' web-sites that are freely available. PubMed provides access to the MEDLINE database of life science and biomedical information that includes articles from

⁷ <http://en.wikipedia.org>

⁸ <http://wikipedia-miner.sourceforge.net>

⁹ <http://www.ncbi.nlm.nih.gov/pubmed>

more than 5,000 academic journals covering medicine, nursing, pharmacy, etc. PubMed offers access to the OLDMEDLINE database that contains printed and digitalized article citations from before 1966, to in-process citations that were not yet indexed by the MeSH headings, to citations that were added to the MEDLINE and considered out-of-scope for biomedicine. Furthermore, the user can reach the citations of journals that were published before the journal was selected for MEDLINE indexing and citations of journals submitted to the PubMed Central¹⁰ (Doms, 2008).

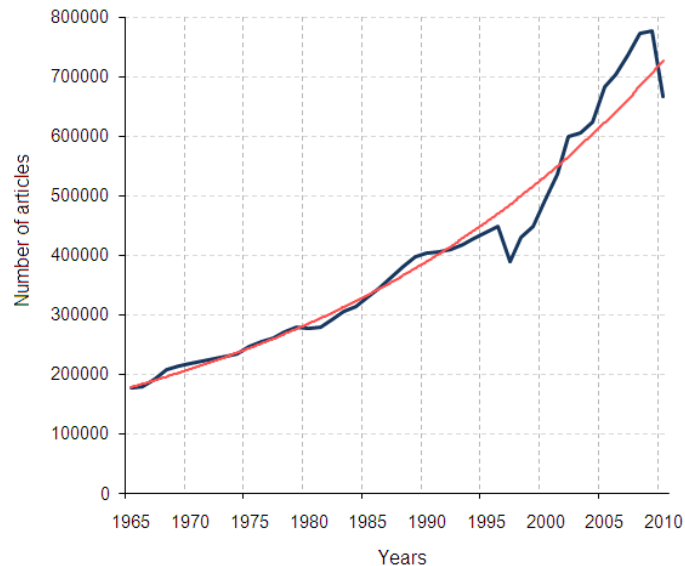


Figure 2.2: Growth of PubMed database through 1965 – 2010 years. Blue line represents numerical growth of articles added to PubMed. Red line represents exponential trend.

2.2.2 GoPubMed

*GoPubMed*¹¹ is a semantic search engine that explores PubMed by using background knowledge of the ontologies, such as the Gene Ontology (GO) and the Medical Subject Headings (MeSH), for answering biomedical questions (Doms, 2008). The advantages of GoPubMed are the following (Doms and Schroeder, 2005). First, the abstracts are categorized according to the ontologies so that user can quickly navigate through abstracts by category. Second, the general concepts, that are related to the query, but do not appear in the abstract, are provided automatically. Third, the user can easily verify the classification results since the ontology terms are highlighted in the abstracts. Fourth, additional information about the queried terms is given, like definition, synonyms, level in the tree, links to Wikipedia pages. Additionally, the user can get an overview about the research trends over the time, relevant journals, top authors, and regional research interests. This information can be used to refine the search.

The concept recognition algorithm, that was implemented by Doms (2008) and will be analyzed in this work, is a machine learning method that is used to annotate the PubMed articles with the Medical Subject Headings (MeSH) concept labels. The purpose of the algorithm is to recognize appropriate ontology terms in the title and abstract of the article. Furthermore, the algorithm has to be able to distinguish the ambiguous concept labels as well. This is done with the help of the context models that are generated from the training data for each term and consist of the positive and negative examples that characterize the term.

¹⁰<http://www.pubmedcentral.nih.gov/>

¹¹<http://gopubmed.org>

2.3 Machine Learning Techniques

Machine Learning is a scientific discipline that focuses on the problem of making accurate predictions for a given set of data based on the past observations. This section describes machine learning techniques that are widely used in different domains. Among them there are Bayesian networks, Naive Bayes classifier, Neural Networks, Support Vector machines, Hidden Markov models, Boosting approach and Maximum Entropy approach. All of them are used for statistical classification of the given data. We will look through all of the mentioned algorithms. At the end we will summarize the advantages and disadvantages of each of them in relation to our problem.

Bayesian Network, also known as belief network, represents a probabilistic graphical model that consists of set of random variables and their dependencies, and forms a directed acyclic graph (DAG). Each node in the graph represents variables that can be latent variables, observable quantities, unknown parameters or hypotheses. The edges between nodes represent probabilistic dependencies among the corresponding variables.

Bayesian Network, a powerful data mining tool with a robust probabilistic model, has several advantages for the data analysis (Heckerman, 1996). It can easily handle incomplete noisy data sets and learn causal relationships. In conjunction with Bayesian statistical techniques it simplifies the combination of domain knowledge and data. Based on the combination of prior knowledge and data, a Bayesian network has the ability to learn about the structure of the system and its parameters. Also, Bayesian statistical methods in conjunction with Bayesian networks offer an efficient and principled approach to avoid the overfitting of data. Beside this, Bayesian network handles situations when some data entries are missing. A Bayesian network is transparent, making it easier for the user to understand the meaning of used parameters and make improvements (Kim and Nevatia, 2000). Unfortunately, they are not applicable to complex problems. It requires a huge amount of data to train the model. The Bayesian Network Classifier Toolbox (jBNC) is a Java toolkit used for training and testing Bayesian Network Classifiers. It is included in WEKA¹² that is an open source product.

Bayesian networks are used for modeling the knowledge in bioinformatics (Chen et al., 2010; Zou and Conzen, 2005), medical diagnosis (Lucas, 2001), vision (Levitt et al., 1990), engineering (Fenton et al., 2007; Mohamed Addin, 2007) and language understanding (Charniak and Goldman, 1989).

Naive Bayes Classifier (NBC) is a probabilistic classifier that is based on Bayes's theorem with naive independent assumptions. The user has to provide the prior probabilities to the assumptions (Fomby, 2008). The independence among assumptions implies that all attributes from a given training data are independent from each other. Naive Bayes produces good results for classification problems, even if the assumption is false in the real world (McCallum and Nigam, 1998). Furthermore, it requires a small amount of training data for the classification. In addition, Domingos and Pazzani (1996) defined the following advantages of Bayesian classifier: "simplicity, learning speed, classification speed and storage space". Naive Bayes classifier should be preferred when the dataset size is small. Also, the model of NBC is robust and self-correcting, i.e. when there are changes in data, the same happens with the result (Islam et al., 2007). Also, the transparency of the naive Bayes classifier model is rated highly (Hilario and Kalousis, 1999). It can handle data noise and missing values (De Ferrari and Aitken, 2006), but its ability to prevent overfitting is the worst in comparison to the other classifiers (Elkan, 2006). A Naive Bayes Classifier implementation is freely available in Weka.

A Naive Bayes Classifier is widely used in areas such as text classification (Robertson and Jones, 1976; Lewis, 1992; Koller and Sahami, 1997) and spam filtering (Metsis et al., 2006).

¹²<http://www.cs.waikato.ac.nz/ml/weka/>

Neural Network, or artificial neural network in the modern usage, is a mathematical model that is built on the principle of biological neural networks — nerve cells' networks in a living organism. So called "neurons" (artificial nodes) are connected together with junctions or "synapses" through which they signal to each other. Such a network of nodes imitates a biological neural network and can be represented as a graph. There are different numbers of connecting layers in the network. The first layer consists of input neurons that send data through the synapses to the second layer of neurons and so on until the data reach the last layer of output neurons. The manipulation of the data is performed thanks to the weights that are attached to the synapses.

The neural networks are efficient classification tool that can be applied to the complex problems with many parameters (White, 1996). It has the ability to learn "on the fly" according to the previous experiences (Lacher, 1999). Furthermore, the neural network is reliable in the predicting in classification and regression problems (Ripley, 2008). The reliability of predictions is degraded with the presence of multiple solutions caused by many local minima found by neural networks (Intrator and Intrator, 1997). Moreover, the neural network model is identified as "black box", i.e. it is difficult to determine and understand its performance. The neural network model is parametric, i.e. a large number of parameters need to be carefully set by the user in order to obtain good results (White, 1996). The neural networks are very slow, equally in training and in validation phases. Weigend (1994) shows that in practice the overfitting can be present in small networks as well as in large neural networks. However, Smith (1993) proposed to increase the number of validation sets in addition to the training set which would make model resistant to the overfitting. The author divides the data set into three subsets: training set (40%), testing set (30%) and overfitting prevention set (30%). Furthermore, the neural networks provide good classification in noisy environments (Cannady, 1998). Unfortunately, they are sensitive to the incompleteness of data set and can behave unpredictably on data that differs from the training set. The Neural Network Toolbox software is available as a commercial plug-in for the MatLab¹³ environment. Also, it is freely available as a part of Tool R¹⁴, which is an open source tool for statistical data analysis.

The neural networks are widely used in chemical and biochemical studies (Baskin I.I., 1999), psychology (Dorrer et al., 1995) and business (Hutchison and Stephens, 1987); they have been successfully applied to various areas of medicine like diagnostic systems, image analysis, drug development (Gant, 2001; Ohno-Machado, 1996).

Support Vector Machines (SVM) are a group of supervised learning methods that are used for statistical classification and regression analysis. The SVM represents a classifier that classifies the objects into two categories according to the fitting properties of these categories. For a given set of training objects, the SVM classifier iteratively builds a model that will predict correctly whether new object belongs to one category or to the other. The training objects are represented as vectors. The SVM classifier constructs a hyperplane between vectors of different categories that maximize the distance between training data points of any categories. The larger the distance is between the categories, the lower will be the generalization error of the classifier. The number of hyperplanes can be high, same as the number of categories and the dimensional space of the data. The decisions made by the SVM classifier are not always easily explainable to the user, i.e. support vector machines are a "black box" classifiers (Barbella et al.). Burges (1998) states that "although SVMs have good generalization performance, they can be abysmally slow in test phase". The SVM has high algorithmic complexity and extensive memory requirements in large-scale tasks (Yu et al., 2010). They are sensitive to the noisy data and thus, they are prone to the overfitting (Abu-Nimeh et al.) which negative affect to the overall performance. Handling missing values is possible with the additional integration of the optimization methods (Pelckmans et al.). While training, the kernel function and its parameters need to be set by the user (Abe, 2010). More detailed explanation about SVM can be found in Burges (1998). The SVM methods are available

¹³<http://www.mathworks.com/>

¹⁴<http://www.r-project.org/>

for free for scientific use from the Support Vector Machine Group¹⁵.

SVM has been successfully applied in such tasks like face identification (Osuna et al., 1997), text categorization (Drucker et al., 1999) (Joachims et al., 1997), medicine, chemistry (Ivanciuc, 2007), bioinformatics (Furey et al., 2000), engineering or database marketing (Bennett et al., 1998).

Maximum Entropy (MaxEnt) is a machine-learning approach that is used for the statistical modeling. In Berger et al. (1996) the authors describe the general idea of MaxEnt as follows: *"Model all that is known and assume nothing about that which is unknown. In other words, given a collection of facts, choose a model consistent with all the facts, but otherwise as uniform as possible."*

The principle of maximum entropy is the following. The testing data is classified into finite number of classes A_n . We assume that each of the classes A_i is assigned with the probability of occupancy $p(A_i)$, where i is the index running over all the possible classes. Probabilities $p(A_i)$ are expressed, as usual, as a number from 0 to 1. Also, we assume that the sum of the probabilities of all classes equals to 1, i.e.:

$$\sum_i^n p(A_i) = 1$$

In case one of the probabilities equals to 1, it follows that all the others are equal to 0. That is we know exactly in which class the data is located and, thus, there is no uncertainty. The uncertainty (also known as entropy) is expressed by some information that we don't have about the class occupied by the data. In case if there is no additional knowledge about the data, we add constraints that will describe possible behavior of the process in order to decrease the uncertainty. Usually, only one constraint is needed. Multiple constraints are possible, but they make mathematical computations more complex. At the end, from a given collection of facts, the probability distribution that leads to the highest value of the uncertainty, i.e. maximum entropy, is selected by the improved iterative scaling algorithm (Nigam, 1999). The selected model must be consistent with the constraints but also uniform.

The Maximum Entropy method is insensitive to the noisy data and is capable to process incomplete data such as sparse data or with missing attributes (Zhao et al.). However, because of sparseness, Maximum Entropy models can suffer from overfitting (Li, 2006). The overfitting can be reduced and the performance improved with the integration of Gaussian prior into maximum entropy (Nigam, 1999). The MaxEnt models can be trained on massive data sets (Mann et al., 2009). However, the MaxEnt has complex mathematical background and it is treated as a "black box" (Borthwick, 1999). The MaxEnt project is one of the open source OpenNLP¹⁶ projects.

The Maximum Entropy Method has been used in Part-of-speech Tagging (Ratnaparkhi, 1996), Prepositional Phrase Attachment (Ratnaparkhi et al., 1994), Named Entity Recognition (Borthwick, 1999) and others.

Hidden Markov Model (HMM) is a statistical Markov model that represents a Markov process with hidden, unobserved states and a corresponding sequence of related and observable outputs. Each state has a set of probability distributions, i.e. state transition probabilities and output probabilities. The main task of HMM is the following: knowing only the sequence of observable outputs, find the values of the unobservable states. An example of a HMM can be viewed in Figure 2.3. A good explanation of hidden Markov models is given in Rabiner and Juang (1986).

The Hidden Markov model is an effective classification tool that can be applied to the complex tasks. The advantage of Hidden Markov model is the ability to create an elegant and understandable model, that can be easily analyzed and improved if it is needed. It uses prior knowledge for

¹⁵<http://www.support-vector-machines.org/>

¹⁶<http://opennlp.sourceforge.net/index.html>

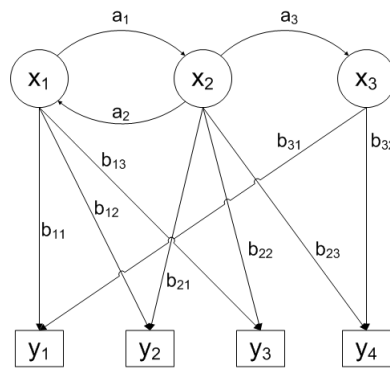


Figure 2.3: Example of hidden Markov model with x - hidden states, y - possible outputs, a - state transition probabilities, b - output probabilities

constraining the training process (Cherry, 2001). The result model is robust and can be a combination of smaller HMMs. Furthermore, the missing data does not effect on the classification accuracy of HMM (Peursum et al., 2005).

However, the Hidden Markov models have several disadvantages (Kadous, 2002). They require a large amount of data for training and this must be only positive data. The number of parameters that need to be set is huge. Also, this method cannot avoid overfitting the training set and is sensitive to the noisy data (Sanches, 2000). In addition, Hidden Markov models are slow in computation in comparison with other classification methods. There is no guaranty that the built model will converge to the truly optimal parameter set of the training set. Additional heuristics need to be applied to solve this problem (Karchin). According to Li (2006) HMM generates inferior results compared to SVM. The Hidden Markov Model Toolkit (HTK) is a portable toolkit that can be used for building and working with HMMs. HTK was originated at the CUED Machine Learning Intelligence Laboratory¹⁷ and it is available free of charge.

Hidden Markov Models have been applied to speech and handwriting recognition, image recognition (Dementhon et al., 2000), machine translation and cryptanalysis (Karlof et al., 2003). Also, HMMs have found applications in many areas of computational biology, e.g. gene finding (Krogh, 1997) and protein structure prediction (Sonnhammer, 1998).

Boosting approach is the machine learning method that is based on the question whether a set of weak learners can create a single strong learner for a given training data (Kearns, 1988). A weak learner is a classifier that is slightly correlated with the training data, whereas a strong learner is strongly correlated to the set. Boosting repeatedly calls weak learners to produce a final hypothesis, i.e. a single strong learner (Kudo and Matsumoto, 2004). Boosting is an efficient classification tool, which is fast and provides transparent models to the user, and shows the model performance and useful features. Schapire (1990) showed that boosting algorithm has polynomial run-time. The AdaBoost, the most used branch of Boosting approach, is a robust method that is efficient in cases with large amounts of data. The approach handles uncertainties, performs error analysis and is insensitive to overfitting while training (Sadeghi et al.). However, the AdaBoost is sensitive to noisy data and outliers. Its performance deteriorates rapidly when the noise is added to the training set. Additional techniques are required to be applied for handling the noise (Freund, 2009). Furthermore, the number of iterations for the learning process must be set by the user.

Examples of algorithms that are based on the boosting approach are AdaBoost (Schapire, 2002), Linear Programming Boosting (Demiriz et al., 2002), BrownBoost (Freund, 2001). Boosting

¹⁷<http://htk.eng.cam.ac.uk/>

approach has been applied to text filtering (Schapire et al., 1998), text and speech categorization (Rochery et al., 2002; Rochery et al.), medical diagnosis (Merler et al., 2001), etc.. *AdaBoost* algorithm, the most well known boosting algorithm, is included into MatLab software. Also, *mboost* (model-based boosting) package is available in the statistical tool R.

Comparison of the machine learning algorithms

The advantages and disadvantages of the described above machine learning algorithms have been summarized in the Table 2.1. We emphasize the first two approaches that lead according to the number of advantages they perform: boosting and maximum entropy approaches. These approaches can be equally well applied to the natural language tasks. Both of the approaches provide a robust unique classification model which is fast in the performance. The availability of the transparent model, the requirement for additional parameters to be set by the user and the ability of the model to handle the noise or overfitting differentiate these approaches. The later problem can be solved with the integration of additional techniques. The rest of the approaches are not efficient classification tools and are sensible to the noise and uncertainty of the data and are prone to overfitting. Moreover, some of them are not applicable on the complex tasks of disambiguation. In Doms (2008), the author recommended to use the maximum entropy approach as a machine learning method for annotation the PubMed documents with the MeSH terms.

2.4 Evaluation Measurements

It is important to know the quality of the used machine learning algorithm. Several statistical measurements can be used to estimate the algorithm performance. These measurements can be collected from a confusion matrix (Table 2.2) that contains information about the real and predicted classifications done by the algorithm.

True positives (TP) - the number of correct predictions that an instance is positive

True negatives (TN) - the number of correct predictions that an instance is negative

False positives (FP) - the number of incorrect predictions that an instance is positive

False negatives (FN) - the number of incorrect predictions that an instance is negative

The aim of the algorithm is to maximize the true positives (TP) and true negatives (TN) predictions. The effectiveness of the algorithm can be characterized with the recall and precision measurements.

Recall measures the ability of the algorithm to find all relevant entities.

$$Recall = \frac{TP}{TP + FN}$$

A high recall score tells that most of the relevant entities were retrieved by the algorithm, while a low recall indicates that the most relevant entities were missed by the algorithm.

Precision measures the ability of the algorithm to retrieve only relevant entities.

$$Precision = \frac{TP}{TP + FP}$$

A high precision score indicates that most of the retrieved entities by the algorithm were relevant. A low precision means that the algorithm cannot distinguish relevant entities while retrieving all entities.

Approach	Robust model	Unique solution	Transparent model	Efficient for complex problems	Handle noise	Handle overfitting	Handle incomplete data	Training / Test time	Ability to learn	Available implementation	No additional parameters
Maximum Entropy	+	+	-	+	+	-	+	slow / fast	+	+	+
Boosting approach	+	+	+	+	-	+	+	slow / fast	+	+	-
Hidden Markov Models	+	+	+	+	-	-	+	slow / slow	+	+	-
Neural Networks	+	-	-	+	+	+ / -	-	slow / slow	+	+	-
Bayesian Networks	+ / -	+	+	-	+	+	+	slow / slow	+	+	+
Naive Bayes Classifier	+ / -	+	+	-	+	-	+	slow / slow	+	+	-
Support Vector Machines	+ / -	+	-	+	-	-	-	slow / slow	+	+	-

Table 2.1: Comparison of machine learning algorithms

		Actual Class	
		Yes	No
Predicted Class	Yes	TP	FP
	No	FN	TN

Table 2.2: Confusion Matrix

F-score is the efficiency measure that combines recall and precision together.

$$F\text{-score} = \frac{2PR}{P + R}$$

F-score is the harmonic mean of the precision and recall, reaching its best value at 1.0 and the worst score at 0.0.

Despite the fact that precision and recall do not depend on each other in theory, in practice they are inversely related. The precision increase leads to the recall reduction and vice-versa. The recall level can be improved artificially. In order to obtain desired recall, more documents must be retrieved by the algorithm. The *precision-recall curve* can be drawn based on the precision and recall values at each iteration (Liu, 2007). The precision-recall curve examines the algorithm effectiveness and overall performance. The goal in precision-recall analysis is to obtain a curve that is closer to the upper-right corner. With the perfect algorithm, only relevant documents will be retrieved and the precision would be 1.0 at any level of the recall. For this case, the precision-recall curve would be a horizontal line. Figure 2.4a shows an example of the precision-recall curve that is not optimal and needs to be improved.

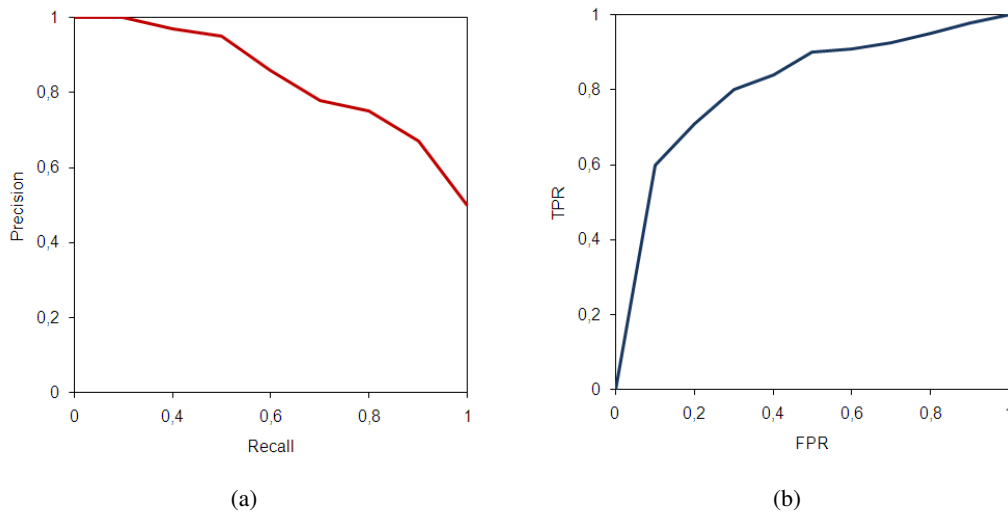


Figure 2.4: Example of (a) the precision-recall curve and (b) the ROC curve.

Other statistical measures that can be applied in the performance evaluation of the algorithm are *sensitivity* and *specificity* (Altman and Bland, 1994). The sensitivity is the recall rate that measures the proportion of correctly identified true positives entities. The sensitivity of 100% means that the algorithm recognized all true positive cases. The specificity is complementary to the sensitivity. It gives information only about the proportion of correctly identified true negatives entities. The specificity of 100% means that no negative cases are incorrectly identified as positive.

$$Specificity = \frac{TN}{TN + FP}$$

The sensitivity is also known as true positive rate (TPR), while $(1 - \textit{Specificity})$ is called false positive rate (FPR). The TPR and FPR are used in the *receiver operating characteristic (ROC)* analysis or the ROC curve, which represents a graphical illustration of TPR versus FPR (Pepe, 2004) and it is an alternative to the precision-recall curve. The ROC curve is a monotone increasing function mapping $(0, 1)$ onto $(0, 1)$. The models with better predictions have the ROC curves closer to the upper left corner (coordinate $(0, 1)$), that represents 100% of the sensitivity and 100% of the specificity and corresponds to the ideal test. It means that the TPR of a test is higher than those of the other tests at any FPR point. The model, which gives always incorrect predictions, is the reverse of the ideal model. The uninformative model has TPR equal to FPR and its ROC curve is a straight diagonal line from $(0, 0)$ to $(1, 1)$, i.e. the model provides no useful additional information (Jones et al., 2010). The accuracy of the testing model is measured by the area under the ROC curve. The ideal ROC curve has the highest area under the curve. The different methods for computing the area under the curve are explained in Le (2009). Figure 2.4b shows an example of the ROC curve.

2.5 Validation Techniques

Once the model is created, different methods can be applied to evaluate its performance. Simple split, holdout, bootstrapping and cross-validation are among the most used evaluation methods.

Simple-split works by splitting the available dataset into the training set and testing set. One part is used to train the model and the other part is used to evaluate the model performance with the help of recall, precision and F-score. However, it is not known exactly how data must be split and which part of data will be used for training and which for testing. An example of such method is *Modified Apte split* that is using the Reuters-22173¹⁸ collection of documents.

Holdout method is the evaluation method that randomly partitions the available dataset into two mutually exclusive subsets, one for the training and one for the testing. The training set contains 66% of the data, and the test set is 33%. The advanced version of this approach is named *repeated holdout* method and it is an iterative process. On each iteration the training set is split again using a 66%/33% split to create two subsets, and so on. The error rates from the different iterations are averaged to generate the overall error rate. The major drawback of this method is the difficulty in predicting the exact number of iterations needed to obtain a particular overall error rate. The holdout method is discussed in Blattberg et al. (2008).

Bootstrapping, introduced by Efron (1979), is a method for estimating the generalization error based on "resampling". The principle of bootstrapping is to select samples of size n with replacement from the original data set of size n in order to form a new dataset on n instances. The newly formed data set is used for the training. Since some instances are sampled more than once, there are cases that are not picked. Those observations that don't occur in the new data set are used for the testing. One version of the bootstrap method is called 0.632 bootstrap. According to it, the probability of an instance not being picked from the original data set is $(1 - 1/n)^n \approx e^{-1} = 0.368$, i.e. the size of the testing set will be 36.8% of the original dataset. Therefore, the training data will contain approximately 63.2% of all instances n . The error rate is a combination of the error from the training set and the error from the testing set. The process can be repeated several times with different replacement samples. At the end, the results are averaged in order to get a better estimate of prediction error. Unfortunately, the recommended number of bootstrap samples that will be enough for estimation is not known and is difficult to predict. The bootstrapping method is discussed in Blattberg et al. (2008).

¹⁸<http://www.daviddlewis.com/resources/testcollections/reuters21578/>

Cross-validation (Liu and Zsu, 2009) is a statistical method that is used to estimate the performance of the algorithm. The method divides data into two sets: the training data set that is used to train the model and the validation data set that is used to test the model. One of the forms of cross-validation is k -fold cross-validation.

In k -fold cross-validation the data is randomly partitioned into the k equally sized parts or folds. One part is used for testing the model, and the remaining $k - 1$ parts are used for the training. The process is repeated k times such that each of k -parts of the data is used exactly once for the testing. A recall, precision and F-score are calculated for each run. The computation of the overall estimation represents the averaging of the recall, precision and F-score from the k -runs. The advantage of this method is that it is not influenced by how the data is divided because every data point is involved in both the testing and the training process. The variance of the resulting estimation decreases as the value of k increases (Kohavi, 1995). The disadvantage of this method is that the algorithm needs to be rerun k times, but this allows us to obtain a more precise estimation.

Various numbers of tests were done on different datasets in order to define the best choice of the fold's number. As a result, it was shown that ten folds is the right number of folds even though there are no strong theoretical explanations for this (Witten and Frank, 2005). Despite the fact that the debates on this topic still exist, the 10-fold cross-validation has been commonly used in the practical tasks.

Specified number of folds, iterative process of the algorithm and unessential knowledge about dataset division are the reasons why we chose to use the 10-fold cross-validation method in our research.

2.6 Accuracy Threshold

As discussed in the previous section we use 10-fold cross-validation as validation technique to evaluate the constructed model. This technique requires setting the accuracy threshold parameter. Setting the accuracy threshold allows us to control the process of measuring the accuracy of model predictions (Mic, 2010). A threshold represents an accuracy bar. Each prediction is assigned with a prediction probability that indicates the correctness of the predicted value. If the prediction probability is above the accuracy threshold, then the prediction is considered to be correct. Otherwise, the prediction is incorrect. The threshold value can be between 0.0 and 1.0. Setting a value closer to 1.0 means that we require high probability for any particular prediction in order to have good predictions. On the other hand, if the threshold value is closer to 0.0, then the predictions with lower probabilities are considered to be good. Setting the threshold to 0.0 is meaningless because every prediction made by the model will be considered correct. The default value for threshold is NULL, which means that the prediction with highest probability is considered to be the target value for choosing the correct predictions. There is no specific recommendation for better threshold value. The type of used data influences on the prediction probability of the model. Different experiments with the different threshold values need to be done in order to determine the appropriate accuracy threshold for a given data set.

2.7 Statistical measurements

So far we reviewed the validation and evaluation techniques that are used to estimate the performance of the algorithm. Nevertheless, there are statistical measurements that can aid us in understanding the data that we are working with. For the statistical analysis we will focus on the median, average, standard deviation and correlation coefficient measurements (Jerome L. Myers, 2003; Walpole et al., 2007).

2.7.1 Median

Median is the "middle" value in the data set that is sorted in the ascending order. If the total number of data set is even, then the median is the mean of the two middle numbers. More precisely, for a given ordered data set $X = \{x_1, x_2, \dots, x_n\}$, where $x_1 = \min_i(X_i)$ and $x_n = \max_i(X_i)$, the median of X will be defined as follows:

$$\tilde{x} = \begin{cases} x_{(n+1)/2} & \text{if } n \text{ is odd,} \\ \frac{1}{2}(x_{n/2} + x_{1+n/2}) & \text{if } n \text{ is even} \end{cases}$$

where n is the total number of data values.

The purpose of the median is to reflect the central tendency of the population in a way that it is not influenced by extreme values or outliers. Thereby, the median emphasizes the true "center" of the data set, whereas the average value is influenced considerably by the presence of the extreme observations.

2.7.2 Average

Average value or arithmetic mean of a given data set is a measure of the middle of this data set, so called centroid of the data set. It is the sum of all data values divided by the total number of data values. Thus, for a given data set $X = \{x_1, x_2, \dots, x_n\}$, the average of X is defined as:

$$\bar{x} = \frac{1}{n} \sum_i x_i$$

where n is the total number of data values.

Average, in its sense, is a point at which a fulcrum is placed in order to balance the system of "weights" which are the locations of individual data from examined data set.

There are several other methods to evaluate the center of location of the data set. Among them are *mode*, *harmonic mean* and *geometric mean* (Beri, 2009). The mode of a given data set is the value that occurs most frequently in the data set. In other words, the mode value represents the most popular value among the rest. Same as for median, the mode value is insensitive to the outliers. Furthermore, the mode value can have multiple meanings since the data set can have several maximum data points. The geometric mean is defined as the n -th root of the product of n positive observations. It is not useful when the number of items is large. The harmonic mean is the reciprocal of the arithmetic mean of the reciprocals. As with geometric mean, the calculation becomes complex when a distribution has a large number of observations. The harmonic mean is less influenced by the outliers than the arithmetic and geometric means, since it is based on the reciprocals.

Due to the existence of many measures of central tendency, there are many measures of spread or variability. The most often used measure of variability is the standard deviation.

2.7.3 Standard Deviation

Standard Deviation measures how far the values of the statistical population are spread out from the average value. A low standard deviation indicates that values tend to be very close to the average, while high standard deviation means that values are overspread in a large range of values.

Given the data set $X = \{x_1, x_2, \dots, x_n\}$. The standard deviation of X is defined as follows:

$$\sigma = \sqrt{\frac{\sum_i (x_i - \bar{x})^2}{n - 1}}$$

where \bar{x} is the average value of X .

2.7.4 Correlation Coefficient

Correlation coefficient measures the strength of the relationship between two or more data sets. Correlation coefficient can assume values in the range $[-1;1]$. If there is an agreement between data sets than the coefficient is close or equal to 1. In case of disagreement between viewed data sets, the correlation coefficient has value -1 or value close to it. The correlation coefficient is equal to zero, if the data sets X and Y are independent from each other. Three major methods for measuring the correlation between variables are mentioned in Hinton (2004).

Method Pearson. It is the most common method for measuring the strength of the linear relation between two data sets (X, Y) . A tendency for high scores on X -axis together with high scores of Y -axis indicates a positive linear relationship between X and Y . If larger scores of X -axis tend to have smaller scores of Y -axis, than we have a negative linear relation. A correlation of zero means that there is no linear relation between two sets. Note that outliers can strongly affect the correlation result (Hinton, 2004).

Given two data sets $X = \{x_1, x_2, \dots, x_n\}$ and $Y = \{y_1, y_2, \dots, y_n\}$ with the average values \bar{x} and \bar{y} of X and Y . The Pearson (r) correlation between X and Y will be defined as follows:

$$r_{xy} = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{(n - 1)\sigma_x\sigma_y}$$

Method Kendall. It is used to identify the relationship between the quantitative and qualitative indicators by the use of ranking (Nelson, 2001). Given two data sets $X = \{x_1, x_2, \dots, x_n\}$ and $Y = \{y_1, y_2, \dots, y_n\}$. The values of X, Y are sorted in ascending order and ranked. There must be only unique values in data sets. Pairs (x_i, y_i) and (x_j, y_j) are concordant if the ranks of both elements agree, i.e. $x_i < x_j$ and $y_i < y_j$ or $x_i > x_j$ and $y_i > y_j$. Otherwise, they are called discordant.

The Kendall (τ) coefficient between X and Y is

$$\tau = \frac{(\text{number of concordant pairs}) - (\text{number of discordant pairs})}{\frac{1}{2}n(n - 1)}$$

Method Spearman. Given two data sets $X = \{x_1, x_2, \dots, x_n\}$ and $Y = \{y_1, y_2, \dots, y_n\}$. The values of X, Y are sorted in ascending order and ranked. The differences $d_i = x_i - y_i$ between ranks of each pair (x_i, y_i) are calculated. The Spearman (ρ) coefficient will be:

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}$$

The values of Pearson, Spearman, and Kendall coefficients are usually similar in the same set of data. The most common measure of correlation is the Pearson correlation. It is used when examined variables have large and continuous values, and thus it will be used in our work.

Spearman's ρ and Kendall's τ differ from Pearson's correlation only in the computations that are done after the numbers are converted into ranks. They perform the analysis based on the ranks instead of on the actual data values that makes them more resistant against the extreme points.

2.8 Semantic Similarity Metrics

In this section we will explain a semantic similarity between words or concepts based on the analysis done by Tsatsaronis et al. (2010) and Hliaoutakis (2005). The semantic relatedness is a measure that shows how relative two or more concepts are. Word thesauri, like WordNet, form a knowledge base for the concept similarity tasks. Beside this, there are approaches that use Wikipedia as structured world knowledge about the concepts of interest.

Leacock et al. (1998) propose the semantic similarity measure between a pair of concepts that consists of both the shortest path connecting concepts and the maximum depth of the taxonomy. The length of the shortest path is calculated as a number of nodes involved in the path. The taxonomy used in this method is WordNet. The semantic similarity between two concepts c_1 and c_2 can be calculated in the following way:

$$sim(c_1, c_2) = -\log \frac{length}{2 \cdot D}$$

where *length* is the length of the shortest path between c_1 and c_2 , and D is the maximum depth of the used taxonomy.

Another method, that is also based on the idea of shortest path, was discussed in Rada et al. (1989). The measure of closeness between two concepts in the taxonomy will be:

$$sim(c_1, c_2) = 2MAX - L$$

where MAX is the maximum path length between concepts c_1 and c_2 , and L is the minimum number of links between two concepts.

Wu and Wu (1994) proposed to measure similarity between two concepts c_1 and c_2 in the taxonomy according to their position relatively to the position of the least specific common concept c :

$$sim(c_1, c_2) = \frac{2H}{N_1 + N_2 + 2H}$$

where N_1 and N_2 are the number of nodes on the paths from c_1 and c_2 respectively, and H is the number of nodes on the path from the least common concept c to root of the taxonomy. The result is ranged between 1 (similar concepts) and 0.

The method proposed by Gabrilovich and Markovitch (2007), Explicit Semantic Analysis (ESA), uses Wikipedia as a knowledge base. ESA represents the meaning of compared texts in a high-dimensional space of concepts derived from Wikipedia. It is done with the help of machine learning techniques that build a semantic interpreter that maps a given text into a weighted sequence of Wikipedia concepts ordered by their relevance to the input. Semantic relatedness of the texts corresponds to the comparison of their concept vectors using, for example, cosine similarity.

Another approach, the Wikipedia Link-based Measure (WLM), proposed by Milne and Witten (2008), calculates semantic affinity between concepts using the links from their Wikipedia articles. This approach provides accurate measures by using only the links between articles and not their textual content. WLM approach is cheaper in usage and more accurate than the ESA, because it requires far less data and resources. For measuring the relatedness between two terms the algorithm performs the following steps. First, it identifies the articles that correspond to the terms. Second, the algorithm checks the similarity between candidate articles. This can be done with two

measures. One considers the outgoing links from the articles, the other the incoming links to these articles. Based on the results from step one and two, the algorithm selects one candidate article to represent each term.

Some approaches can be classified as hybrid measures because they combine both the hierarchy of the thesaurus and statistical information for compared concepts from large corpora. For example, the similarity approach proposed by Resnik (1995) is a hybrid approach. It measures two concepts according to their Information Content (IC) that represent the least common ancestor of the concepts. More specifically, the semantic similarity between two given concepts c_1 and c_2 will be defined in the following way:

$$sim(c_1, c_2) = IC(c)$$

where c is the least common ancestor of c_1 and c_2 . The Information Content (IC) of a concept is defined as:

$$IC(c) = -\log p(c)$$

where $p(c)$ is the probability of occurrence of c in a large corpus.

Another hybrid metric was proposed by Jiang and Conrath (1997). It is also based on the notion of Information Content. The semantic similarity between two given concepts c_1 and c_2 , and their least common ancestor c_0 is defined as:

$$sim(c_1, c_2) = \frac{1}{IC(c_1) + IC(c_2) - 2 \cdot IC(c)}$$

Development of a Tailor-made Metric

In our research we defined our own metric for measuring the semantic similarity between concepts that is fast and simple to compute and can be applied to the hierarchical structure of the used thesaurus. We define the similarity by means of the semantic distance that is opposite to the similarity. The greater the semantic distance is, the weaker is the relatedness of concepts, and vice versa. We consider MeSH ontology as a knowledge base. A MeSH term is considered to be semantically distant from the PubMed document if the distance in the MeSH tree between the MeSH terms of this document and the current term is above the specified depth threshold. In our case the depth threshold is equal to 6. We define the distance between two MeSH terms t_1 and t_2 as follows:

$$Distance(t_1, t_2) = MeSHLevel(t_1) + MeSHLevel(t_2) - 2 * NumberOfCommonParents,$$

where $MeSHLevel(t_i)$ is the minimal depth-level of the term i in the MeSH Tree.

Figure 2.5 represents an example of calculating the semantic distance between terms A and B in the MeSH tree. Terms A, B, C are indicated in the tree and term B appears twice. The distance between terms A and B is calculated in the following way:

$$Distance(A, B) = 2 + 2 - 2 * 0 = 4$$

The following remarks are considered during the calculations. First, the semantic distance of the PubMed document is the maximum distance among the distances of MeSH terms to the current term. Second, if one of the MeSH terms of the PubMed document is in the child branches of the

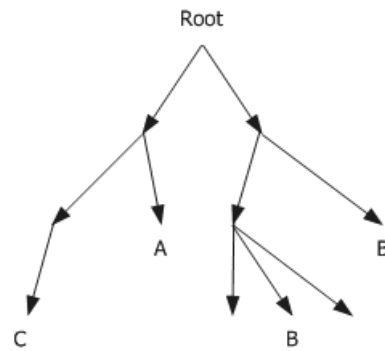


Figure 2.5: Example of calculating the distance between MeSH terms

term, than we skip this document. Third, if a MeSH term of the document is repeated several times in the MeSH tree, the algorithm will take the shortest distance between it and the current term.

2.9 Sense of Ambiguity

We will describe ambiguity phenomenon from different points of view, namely from English dictionary definition and linguistics.

Dictionary Definition

According to the Merriam Webster English Dictionary¹⁹, the noun *ambiguity* has the following meanings:

1. the quality or state of being ambiguous especially in meaning
2. uncertainty

where adjective *ambiguous* is defined as:

1. doubtful or uncertain
2. capable of being understood in two or more possible senses or ways

Thus, *ambiguity* has two interpretations:

1. uncertainty
2. the capability of being understood in two or more possible senses or ways

The uncertainty meaning will be ignored here, because the lack of sureness deals with the user's knowledge about the domain, while the document content itself could be precise and clear. Therefore, according to the dictionary definition, *ambiguity* means the capability of being understood in two or more possible ways.

Linguistic Definition

Linguistic ambiguity can be split into three different classes, namely the lexical ambiguity, syntactic ambiguity and semantic ambiguity. This classification is not mutually exclusive. The ambiguity

¹⁹ <http://www.merriam-webster.com/dictionary>

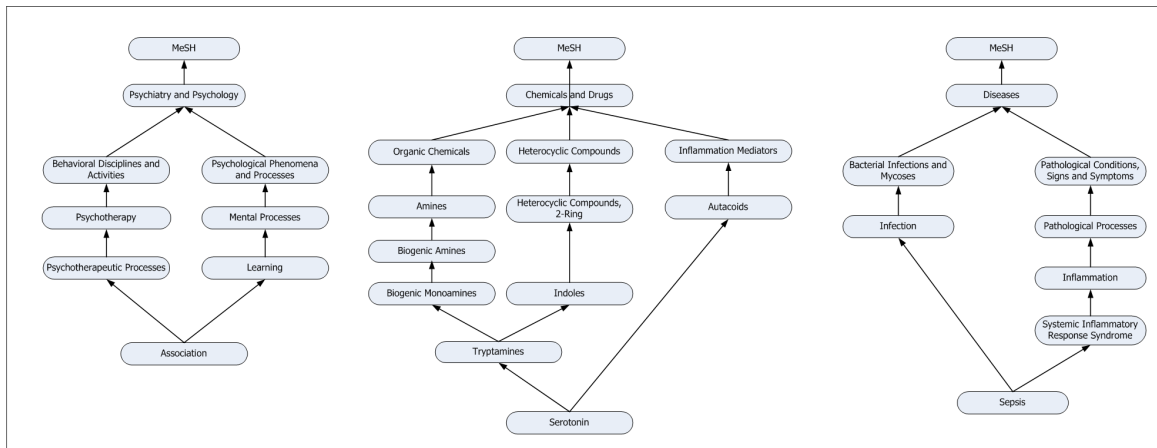


Figure 2.6: Ambiguous MeSH terms association, serotonin and sepsis and their position in MeSH tree.

can occur as a result of the combination of several classes.

The *lexical ambiguity* of a word or phrase occurs if a word has more than one meaning. The lexical ambiguity can be subdivided into homonymy and polysemy. The *homonyms* are group of words that share the same spelling and pronunciation, but have different meanings. Usually such words have different histories of origin. For example, the word "bank" has meanings "financial institution", "building" and "edge of the river". The *polysemes* are group of words with the multiple, related meanings, but one etymology. For example, word "wood" has meanings "something made of wood" and "a dense growth of trees", that differs, but with common etymology.

The *syntactic ambiguity* occurs when a given sentence of words can be parsed in more than one way. For example, the sentence "Flying planes can be dangerous" could mean either the act of flying planes is dangerous, or it could mean that planes that are flying are dangerous.

The *semantic ambiguity* arises when a given sentence has more than one way of being read within its context although it contains no lexical or structural ambiguity. For example, the sentence "All linguists prefer a theory" means that all linguists love the same one theory, or, that each linguists loves a, perhaps different, theory.

In our work we are dealing with lexical ambiguity of the MeSH terms, i.e. the terms have more than one meaning. In the biomedical domain, this can be detected by the term appearing in at least two places in the MeSH tree. In other words, if a MeSH term has more than one parent, then it is considered to be ambiguous. For example, the MeSH term "Association" (meshID: 1244) might be ambiguous in biomedical articles because it is located in two branches "Behavioral Disciplines and Activities" and "Psychological Phenomena and Process" from MeSH category "Psychiatry and Psychology". Some other examples of ambiguous MeSH terms are "Serotonin" (meshID: 12701) and "Sepsis" (meshID: 18805) shown in Figure 2.6.

Chapter 3

Experiment

The significant part of any research project is the experiment stage, in which the researcher is checking the validity of his/her set of hypothesis. This chapter contains the analysis of the concept recognition algorithm that is an efficient tool for the annotation of the biomedical abstracts with the ontology concepts of the Medical Heading Subjects (MeSH). The algorithm creates the context models that characterize the MeSH terms and that consist of lexical tokens taken from the related PubMed articles.

The chapter contains both the formulation of hypotheses and their analysis. The first part examines the general behavior of the algorithm with respect to the different thresholds levels, diverse limits in training datasets size, the four term features, and the four different options for selecting negative dataset. The experiments done in this section improve the work done in Macari (2010). In the second part, the analysis of the algorithm in relation to the ambiguous biomedical terms is performed. The biomedical search engines PubMed and GoPubMed and the generic search engine Yahoo were used to obtain the statistical information about the frequency of documents presence in the biomedical database and on the Web. Furthermore, the analysis is provided for the term recognition made by GoPubMed and the number of senses in Wikipedia and WordNet for MeSH terms.

For the experimental work we used three MeSH branches characterized by the following categories: Anatomy, Diseases and Psychiatry and Psychology; thus considering a total of 6654 Mesh terms. For more details on the MeSH and its branches, please see Section 2.1.2.

3.1 Examined hypotheses

For the performance evaluation of the concept recognition algorithm, we set up the list of questions and hypotheses that will be explored in this part of our work. In general, we are looking for the answers for the following hypotheses:

Question 1. Are all PubMed articles annotated with the MeSH concepts in reasonable time frame?

Question 2. What is the optimal accuracy threshold?

Question 3. How many documents are sufficient for the training?

Question 4. What is the influence of the term features namely the title, the abstract, the journal and the publication year on the algorithm efficacy?

Question 5. How should the negative examples be selected?

Hypothesis 6. The terms that are more general in their sense will have a large number of documents in PubMed, GoPubMed and Yahoo.

Hypothesis 7. MeSH hand annotations are often taken from full text and don't appear literally in the title and abstract.

Hypothesis 8. MeSH hand annotations that appear literally in the title and abstract, don't appear according to GoPubMed.

Hypothesis 9. There is a correlation among thesauri WordNet and Wikipedia.

3.2 Growth of PubMed and MeSH databases

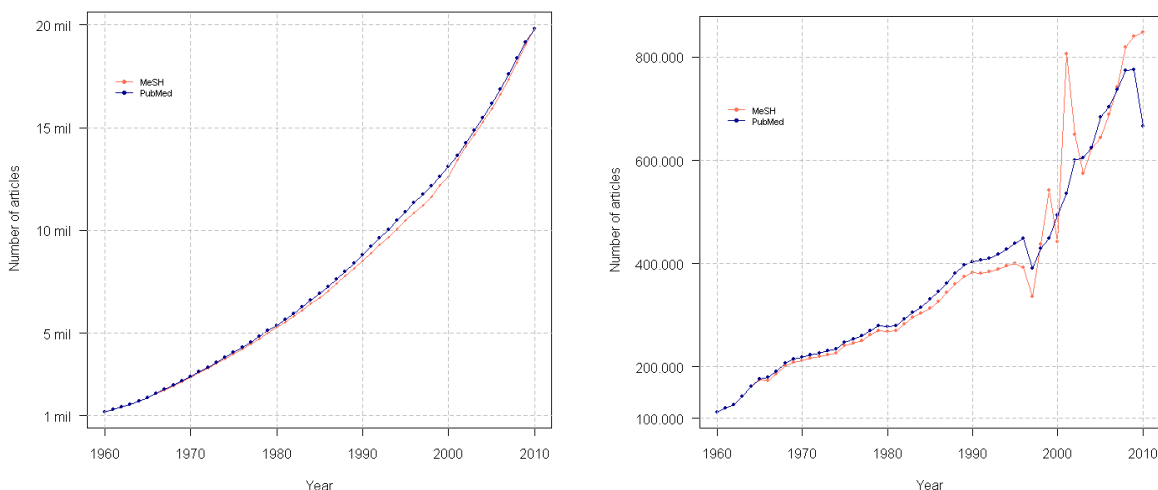
Before starting the experiment, let us analyze the performance of PubMed literature database and MeSH ontology in terms of their capacity. As mentioned in Section 2.2.1, the PubMed database is growing exponentially. Each year a huge number of scientific articles are submitted to it. Interesting to know if the manual annotation process evolves with the same tempo and if the annotation process for newly added articles is fast.

Question 1. Are all PubMed articles annotated with the MeSH concepts in reasonable time frame?

Experiment. The experiment involves the extraction of the number of published documents and the number of hand annotated documents from PubMed database. The analysis corresponds to both: the analysis of the results extracted in combination with previous years and the analysis of the results examined separately from the results of previous years.

Results. Figure 3.1 visualizes the growth of PubMed and MeSH since 1960. Figures 3.1a and 3.1b show the total number of publications and the number of publications for each year of PubMed and MeSH. From Figure 3.1a we can conclude that the development of PubMed and MeSH is on the same level, i.e. all PubMed articles are MeSH annotated in a reasonable time frame. A big gap between PubMed and MeSH in year 2001 (Figure 3.1b) indicates the fact that usually not all of the documents are annotated in the same year of submission to PubMed. Manual annotation of the PubMed articles, which is based on relevant literature, is a tedious, volumetric and time-consuming process. Due to the large number of PubMed articles submitted for annotation each year, the large number of the MeSH terms (Section 2.1.2), the labor-intensive annotation process, the necessity to train human indexers, the high cost of indexers, etc., the annotated process became semiautomatic (Aronson et al. (2008)). It was reasonable to develop alternative methods in order to improve the efficiency of indexing bibliographic data. The Medical Text Indexer System was developed and is available for indexing since 2002. The purpose of the system is to help indexers to index the biomedical articles, namely the title, the abstract and the full text. The system is a combination of two MeSH indexing methods (Név  ol et al., 2005): the Natural Language Processing (NLP) approach based on MetaMap indexing and the PubMed Related Citations approach. To guarantee the correctness, the received MeSH hand annotated documents are manually verified.

Conclusions. The performed analysis showed that MeSH and PubMed evolve with the same high speed tempo despite the fact that the manual annotation is a time consuming process. All PubMed articles are annotated with relevant MeSH concepts in a reasonable time period. The usage of the Medical Text Indexer System makes the annotation process semiautomatic and improves



(a) Combined results with previous years

(b) Separate results for each year

Figure 3.1: Growth of PubMed and MeSH hand annotations per year: (a) together with the results of previous years, (b) separately from the results of previous years.

the efficiency of indexing PubMed articles.

3.3 General behavior of the algorithm

In this section we will analyze the general behavior of the concept recognition algorithm. The answers for the first four questions will be given. At first we define the MeSH terms that will be used for this part of work. Then, the evaluation of the algorithm will be done according to the given questions.

3.3.1 MeSH Terms

For the first part of the experiment we selected randomly 10 MeSH terms from the MeSH branches Anatomy, Diseases and Psychiatry and Psychology (Section 2.1.2). Table 3.1 contains information about these terms like name, definition, synonyms, number of documents that contain the term literally, number of hand annotated documents and the first year of publication of the term. The data was extracted with the help of PubMed and GoPubMed. The numbers about documents that contain the term literally and hand annotated documents by MeSH were extracted with the help of PubMed by querying "*term name*" and "*term name*" [*mh:noexp*] respectively, where [*mh:noexp*] turns off the automatic inclusion of MeSH subheadings for the term. The information about terms was extracted on May, 2010.

Name	Definition	Synonyms	Documents that contain the term literally	Hand annotated documents	Publications since
Sepsis	Systemic inflammatory response syndrome with a proven or suspected infectious etiology. When sepsis is associated with organ dysfunction distant from the site of infection, it is called severe sepsis. When sepsis is accompanied by HYPOTENSION despite adequate fluid infusion, it is called SEPTIC SHOCK.	Severe Sepsis Blood Poisoning Septicemia	69 461	34 183	1886
Tricuspid Atresia	Absence of the orifice between the RIGHT ATRIUM and RIGHT VENTRICLE, with the presence of an atrial defect through which all the systemic venous return reaches the left heart. As a result, there is left ventricular hypertrophy (HYPERTROPHY, LEFT VENTRICULAR) because the right ventricle is absent or not functional.	Tricuspid Atresias Absent Right Atrioventricular Connection Tricuspid Valve Atresia	1 174	278	1936
Merkel Cells	Modified epidermal cells located in the stratum basale. They are found mostly in areas where sensory perception is acute, such as the fingertips. Merkel cells are closely associated with an expanded terminal bulb of an afferent myelinated nerve fiber. Do not confuse with Merkel's corpuscle which is a combination of a neuron and an epidermal cell.	Merkel's Receptor Merkel Receptor	600	173	1965
Dentate Gyrus	Gray matter situated above the gyrus hippocampi. It is composed of three layers. The molecular layer is continuous with the HIPPOCAMPUS in the hippocampal fissure. The granular layer consists of closely arranged spherical or oval neurons, called granule cells, whose AXONS pass through the polymorphic layer ending on the DENDRITES of pyramidal cells in the hippocampus.	Fascia Dentata Gyrus Dentatus Dentate Fascia	10 882	3 499	1958
Breast Cyst	A fluid-filled closed cavity or sac that is lined by an EPITHELIUM and found in the BREAST. It may appear as a single large cyst in one breast, multifocal, or bilateral in FIBROCYSTIC BREAST DISEASE.	Breast Cysts	425	140	1947
T-Lymphocyte Subsets	A classification of T-lymphocytes, especially into helper/inducer, suppressor/effecter, and cytotoxic subsets, based on structurally or functionally different populations of cells.	T-Cell Subset T-Lymphocyte Subset	22 032	18 586	1951
Mycobacterium avium-intracellulare Infection	A nontuberculous infection when occurring in humans. It is characterized by pulmonary disease, lymphadenitis in children, and systemic disease in AIDS patients. Mycobacterium avium-intracellulare infection of birds and swine results in tuberculosis.	Mycobacterium avium-intracellulare Infection Mycobacterium intracellulare Infection	2 443	2 343	1971
Coccyx	No definition available	-	1 005	744	1934
omega-Agatoxin IVA	A neuropeptide toxin from the venom of the funnel web spider, Agelenopsis aperta. It inhibits CALCIUM CHANNELS, P-TYPE by altering the voltage-dependent gating so that very large depolarizations are needed for channel opening. It also inhibits CALCIUM CHANNELS, Q-TYPE.	omega Agatoxin IVA omega Aga IVA	761	389	1992
Hermaphroditism	A state of intersex or sexual ambiguity, involving the GENOTYPE, the GONADS, the reproductive tract, and/or the external GENITALIA or PHENOTYPE. This concept covers TRUE HERMAPHRODITISM and PSEUDOHERMAPHRODITISM. True hermaphrodites are rare and they possess gonadal tissues of both SEXES, tissues from the OVARY and the TESTIS. Pseudohermaphrodites possess gonadal tissue of one sex but exhibit external phenotype of the opposite sex.	Intersexuality	4 687	4 331	1859

Table 3.1: MeSH Terms

3.3.2 Classification Delta

Question 2. What is the optimal accuracy threshold?

Experiment. The experiment involves the application of 10-fold cross-validation (see Section 2.5) on 10 randomly selected MeSH terms using different validation thresholds. The concept recognition algorithm uses threshold that is equal to $0.5 + \delta$, where δ has values between 0.0 and 0.4. The documents are classified as true positive or as false positive according to the coherency between the prediction probability of the model and the defined threshold. The aim of this experiment is to determine the appropriate accuracy threshold that achieves the best prediction accuracy of the model.

Results. The average precision, recall and f-score values were calculated for the thresholds with δ values between 0.0 and 0.4 (Table 3.2c). These results are visualized in the Figure 3.2a according to various threshold levels. The increase of the threshold values in Table 3.2c shows a double effect. It leads to an increase of the average precision and to a decrease of the average recall and average f-score. In theory, a high threshold implies high requirements in correct document classification. A high precision indicates that more relevant (indeed true positives) documents will be retrieved, while low recall indicates that most of them will be recognized as false negatives. As a result, less documents will be recognized as true positives by the model. The classification process becomes more strict with the increasing threshold.

The statistical data in the Table 3.2c indicates the abnormal behavior of the algorithm. The algorithm reaches its best f-score value (97%) at the threshold 0.5 with $\delta = 0.0$, which indicates a perfect prediction ability of the model. However, the growth of delta contributes to the accuracy growth of the classification model that is confirmed by the drop of recall. Nevertheless, the precision level remains very high for all deltas' values. Thus, we decided to use delta equal to 0.1 because even a small delta provides a high precision in classification. Also, the threshold value defines the minimal accuracy value for the algorithm to classify the documents, i.e. the amount of correctly classified documents percentagewise. For example, the threshold value 0.5 ($\delta = 0.0$) implies 50% of correctly classified documents among the total number of documents that were processed by the algorithm. Thus, applying $\delta = 0.1$ (threshold value equals to 0.6) will conduce to get results with confidence not less than 60% while keeping high algorithm efficiency.

In order to recheck the unexpected results in the above experiment, we ran the experiment once again, but this time we checked manually, whether the analyzed terms occur literally in the true positive and true negative documents (in the title and abstract of the document). A minor increase of the precision is expected, while the recall and f-score are expected to drop their values with the increase of delta. Meanwhile, the recall should drop faster than f-score does. The results, as presented in Table 3.2d, show that the precision basically remains at the same level but with insignificant increase. However, the recall value drops as it was expected, because the classification model becomes more precise in the selection of the relevant documents while the delta is increasing. The f-score curve is located between the precision and recall that is rationally explained by being a harmonic mean of precision and recall. These findings prove that the algorithm is working properly. The results of this experiment differ from the algorithm's results (Table 3.2c), because we checked literal occurrence by application of simple matching, while the algorithm applied tokenization and stemming procedures to the PubMed abstracts.

Conclusion. By analyzing the results of the experiment, we determined that the threshold value 0.6 ($\delta = 0.1$) is sufficient for the given classification task. Therefore, at least 60% of the overall documents are correctly classified while the algorithm keeps high prediction ability. This threshold value will be used in the rest of our experiments.

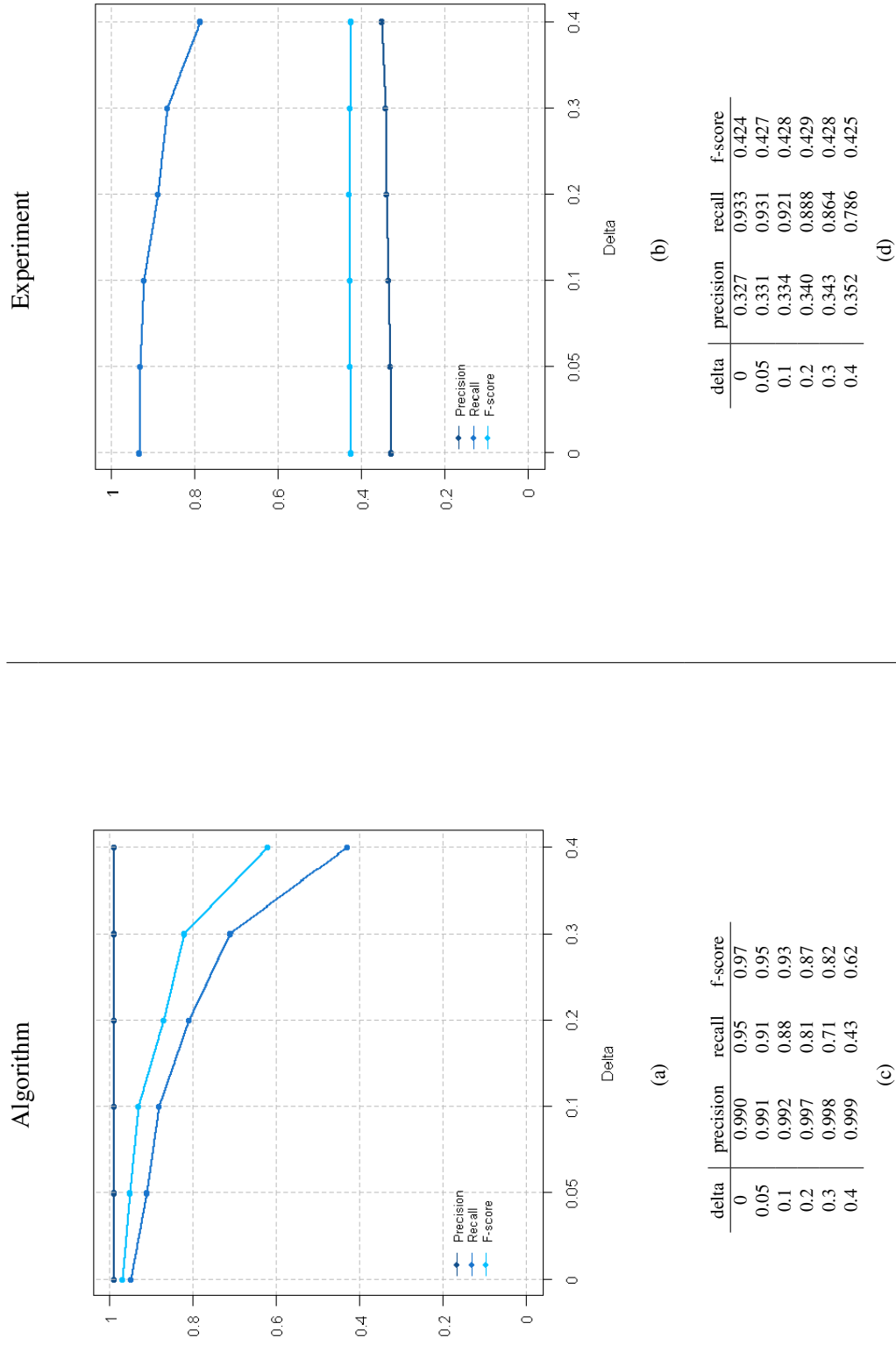


Figure 3.2: Algorithm behavior according to different thresholds ($\delta = [0, 0.05, 0.1, 0.2, 0.3, 0.4]$). The performance is shown with averaged precision, recall and f-score. (a) - visualization of the algorithm behavior, (b) - visualization of the manually checked algorithm behavior, (c) - statistical information for the algorithm performance, (d) - statistical information for the manually checked algorithm performance.

3.3.3 Training data size

The algorithm requires two different datasets: one for training and one for testing the classification model. The following experiment was performed in order to determine the optimal size of the training dataset that is sufficient for the algorithm to make correct predictions.

Question 3. How many documents are sufficient for the training?

Experiment. The experiment involves the iterative run of the algorithm with different step sizes that correspond to the number of training documents. The following step sizes were used: 100, 500, 750, 1000, 1500, 2000, 4000, 6000, 8000, 10000 where the lower and upper bounds are equal to 100 and 10000 documents respectively. The PubMed documents of 10 MeSH terms were used as the training data. The 10-fold cross-validation was applied to evaluate the algorithm performance on each iteration step. The algorithms' accuracy threshold value was set to 0.6 ($\delta = 0.1$) as discussed in the previous section.

Results. Figure 3.3a visualizes the results of the experiment. The error bars indicate the standard deviation of the f-score which varies between 0.82 and 1.0. Note that the f-score value increases with the increase of the training dataset size. Figure 3.3a shows high performance of the algorithm when the number of training dataset is equal to or larger than 5000 documents. Thus, we can assume that the upper bound of 5000 documents is the required amount of data that is needed for the algorithm to make a correct classification.

Since the algorithm was tested only on 10 MeSH terms, this number of terms is insufficient for making ultimate decision about the training dataset size. In order to check the previous assumption, we run the same experiment for 4078 MeSH terms that were randomly selected similar to Section 3.3.1. The results are visualized in Figure 3.3b. The f-score trend line is a logarithmic curve on the interval [100;1000] of documents. The interval [1000;10000] documents reflects the linear trend that increases insignificantly on interval [1000;5000] and turns into a straight horizontal line on interval [5000;10000]. Due to the presence of noise points, the f-score value varies in the range between 0.85 and 0.98 on interval [5000;10000]. The straight horizontal trend line affirms that the algorithm has the same effectiveness operating with 5000 training documents as with 10000 documents. In confirmation, the following experiment was applied on the interval [5000;10000]. We removed the minimum f-score values for the terms with more than 5000 documents while running the algorithm again. The graphical results (Figure 3.3c) look similar to the Figure 3.3b. The f-score value didn't improve significantly on the interval [5000;10000] documents that is confirmed by the percent of the noise points that remained to be nearly the same as before. After this experiment the f-score value kept to vary from 0.85 till 0.98. The analysis shows that there are 26 terms (5%) out of 470 that have f-score value less than 0.9 in the examined interval. Due to the stability of f-score value on the interval [5000;10000] documents, we can conclude that the limit of 5000 documents for training process is sufficient for classification procedure.

It can be noticed that the cut-off with 1000 documents is also possible to be set. The percentage of noise is decreasing with a small pace and the f-score range becomes narrower starting from 1000 documents. If there is a lack in computational resources, 1000 documents will be sufficient for good prediction level. However, in order to be on the safe side in making predictions and have a stability in the noise level and f-score value, we will remain with the cut-off of 5000 documents.

Conclusion. In this experiment we defined that the algorithm has good prediction ability using the training dataset of size in range between 5000 and 10000 (f-score $\in [0.85; 0.98]$). The set of experiments shows the stability in the classification on the interval [5000;10000] documents with the lowest percentage of errors (5%). As a result, in our further experiments we will consider the training dataset of size 5000 documents.

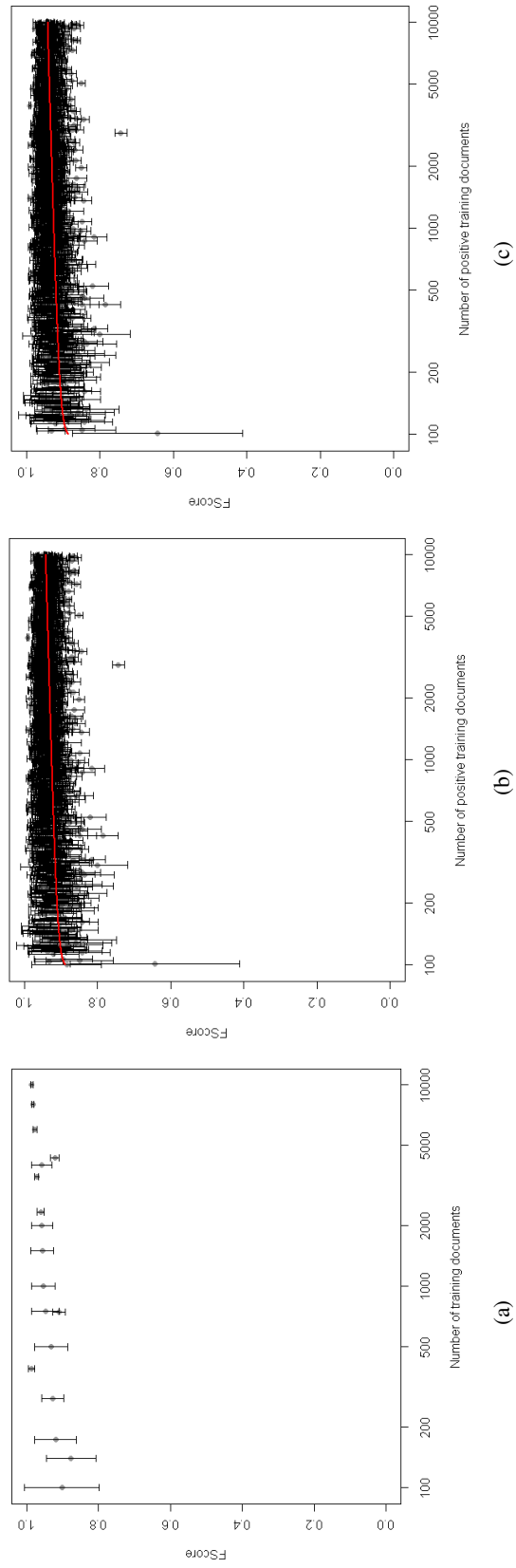


Figure 3.3: Performance of the algorithm on (a) 10 MeSH terms, (b) 4078 MeSH terms and (c) 4078 MeSH terms without min f-score values for terms with more than 5000 documents.

3.3.4 Feature Vectors

The aim of this experiment is to analyze the influence of different term features to the final result. The received results are based on context models that the algorithm creates for each MeSH term. A context model represents a feature vector that characterizes the term.

Question 4. What is the influence of term features namely the title, the abstract, the journal and the publication year on the algorithm efficacy?

Experiment. The impact of term features on the algorithm performance was analyzed in this experiment. The term features are extracted from PubMed documents and represent words from the title, abstract, name of the journal and year. The words are combined into context models that characterize the term. There are two types of the context model - positive and negative. While the positive context model contains features that characterize the term and are connected with it by the sense, the negative context model represents the set of features that have nothing in common with the term and contains only general or insignificant information. For each MeSH term the algorithm creates the positive and negative context models. The following feature notation is used in this experiment:

t - words that occur in the title
p - words that occur in the abstract
j - journal name
year

The context model can be describes as feature vector of the term. Table 3.2 contains the example of a positive context model that characterizes the MeSH term "Gallstones" (meshID: 42882). The total number of features contained in one context model is 1000 units.

#Gallstones
p=stones
year before 1999
t=bile
p=choledocholithiasis
year before 2000
p=gallbladder
t=bile_duct
t=choledocholithiasis
t=biliary
p=bile_common_duct
year before 2001
p=cholecystectomy
j=Gastrointest Endosc
t=duct
...

Table 3.2: Example of the positive feature vector for term "Gallstones"

Feature vectors are created with the help of tokenization and stemming procedures. During tokenization, the title and abstract of the article are split into tokens. Token represents a basic unit of the sentence, for example, a word or a bracket. Next, a stemming procedure is applied,

which transforms tokens into their morphological root. For example, words "fishing", "fished", "fish" and "fisher" will be stemmed to the root word "fish". Afterwards, stop words are removed that represent words with low information value, for example, words like I, both, down, are, etc.. The labels "t=" and "p=" are concatenated to the beginning of each word stem according to the part of the article they were extracted from. Journal features are constructed by concatenation of the label "j=" and the name of the journal. Usually, a journal name consists of one word. The year represents the year when the article was published. For the transformation of the year into a feature-mode, a featurizing procedure is applied. During this procedure, the words "after" and "before" are concatenated with the years from 1950 till publication year and from publication year till 2020 respectively. For example, consider that the article was published in 1990. The procedure adds "after" to the years starting from 1950 till 1990 and adds "before" to the years starting from 1990 till 2020. As a result, we receive the following year-features:

```

year after 1950,
year after 1951,
...
year after 1989,
year before 1991,
year before 1992,
...
year before 2020

```

In this experiment we applied 10-fold cross validation on 10 randomly selected MeSH terms using the accuracy threshold value 0.6 ($\delta = 0.1$). We run the algorithm several times with different combination of features that are extracted from PubMed documents. The following combinations of features were used:

```

TAYJ = title / abstract / year / journal
TAJ  = title / abstract / journal
TAY  = title / abstract / year

T     = title
TA    = title / abstract
TY    = title / year
TJ    = title / journal

A     = abstract
AY    = abstract / year
AJ    = abstract / journal

Y     = year
YJ    = year / journal

J     = journal

```

The combination TAYJ is used in the standard classification procedure of the algorithm, when all features are extracted from PubMed articles. The other combinations consist of all other possible feature combinations. We hypothesized that the features title and abstract must have big impact on the algorithms work. Beside this, we assume that features year and journal name are insufficient and could possibly be omitted in the model creation.

Results. A short experiment was done to identify the ratio of each feature in the context model. As raw data we used the context models of 10 randomly selected MeSH terms (Section 3.3.1). These context models were created by the algorithm that explored the title, abstract, journal name and publication date of the PubMed documents. The results are averaged and given percentage-wise. From Table 3.3, it can be noticed that the main source of features are the abstract and title of the article because their share composes 95,8% of the context model size.

	t	p	j	year
%	13.6	82.2	2,1	2,1

Table 3.3: Proportion of term features in context model.

Above we assumed that features from the title and abstract were most relevant to the algorithm efficiency. More than that, we marked out 13 different possible combinations of features. Further, we will analyze the influence of each combination to the algorithm's results. Figure 3.4 contains the graphical visualization of the algorithm's performance according to the extraction of different features.

First, we checked the influence of the journal and year in combination with the title and abstract. From Figure 3.4a, it can be noticed that the f-score has equally high trend lines while testing on combinations with all features (blue line) and features title-abstract-year (black line). Thus, the journal does not affect the f-score while using features title, abstract and year. The maximum f-score value decreases abruptly down to 80%, when we exclude the feature year (red line) in the next step. The f-score decreases even more when the feature journal is also eliminated (green line). As a result, the feature journal is not as essential as year and can be considered as unimportant. Still it can provide some valuable information when the feature year is not presented, otherwise it has no effect. The improvement of performance with the journal is lower than performance with the year.

Second, we analyzed the trend lines of each feature separately (Figure 3.4b). It is interesting to notice that the trend line for the abstract is located lower than the trend line for the title and has a negative slope. The abstract of the article, in its sense, comprises more informative sense than the title does. One assumption would be about the noise inside the abstract. Although the abstract is the most informative feature, it can still mislead the classifier, meaning it contains noise that leads to lower trend line. Another assumption was about empty/null abstracts in PubMed documents. We analyzed the percentage of null abstracts in the overall explored documents. The number of articles without abstract is around 20% of the total amount of articles. This leads only to a small influence for the classifier due to the fact that this is a quite small share. Beside this, the features year and journal are not very informative. The journal trend line is the lowest on the plot, i.e. using only the journal as a core characteristic implies low algorithm efficiency. The year trend is surprisingly getting better after 4000 documents and reaches the highest point among all trend lines at 10000 documents.

In the next step, we tested the algorithm using combinations of two different features. To each feature we added additional feature in order to show how features individually affect the algorithm's performance. We expect that by adding some additional features to the classifier, it will perform better. The results are visualized in Figures 3.4c- 3.4f corresponding to the experiments with the title, abstract, year and journal. The title trend line has less improvement in combination with the journal than with the year (Figure 3.4c). The combination with the abstract has the lowest trend line with a negative slope because the abstract of the article may contain noise. The abstract trend line has no improvement in combination with the journal and is located lower than the title-abstract trend line (Figure 3.4d). The combination of the abstract and year gives the highest performance of the classifier. The last combination has a larger improvement than the

combination year-title if applied to a small data sample, but later on they exhibit in the same maximum (Figure 3.4e). The less successful combinations are the year-journal and single year; they have the lowest trend lines that coincide. As mentioned above, the journal in combination with the other features brings less progress in performance than the year (Figure 3.4f). As a result, we can conclude that the feature year is important in the classification task.

The importance of the year attracts attention. There are several possible explanations for this. Firstly, it might be the case that a particular MeSH term was widely discussed after/before some year of publication. The term has a trend in publications during particular years. Therefore, the model might learn that a certain term is very important during a particular period of time. As a result, the model captures the connections between the term and years of publications. However, this assumption does not fit for the disambiguation problem for the MeSH terms. A lot of other terms could be highly discussed during the specified years. Secondly, it can happen that the used dataset sample is biased for the MeSH terms and for some years. A term could be equally popular along the period of its existence, but the training and testing datasets will have only annotations for some parts of this period. The first explanation is more probable in our case, because during the analyses we covered almost all PubMed publications for each term. Exceptions are "Sepsis" and "T-Lymphocyte Subsets" that have more than 10000 hand annotated documents. The second explanation is suitable for them.

Since the combination of all features has the best performance, we used this combination in the credibility analysis of context models. As mentioned above, the algorithm creates the positive and negative context models, which characterize the term in a positive and negative sense. We analyzed the top 10 of the features from positive and negative context models. The results are presented in the Tables 3.4 and 3.5 respectively. The cell of the table is colored in gray if the feature makes sense for the term, otherwise the cell has white background. Yellow cell represents a wrong association of the feature with the term. Let us consider the MeSH term "Sepsis". The positive features like *bacteremia*, *septicemia* and *septicaemia* make sense for the term. Septicemia and septicaemia have the same meaning and are synonyms with the term "Sepsis". Bacteremia is the synonym for infection and blood poisoning that occur in the definition and synonym list of the term "Sepsis". There are six year features that are not related to the term in its sense, but indicate the publishing year of the term's articles. These features are marked in white color. Among the negative features are features like *Here*, *PRC*, *social*, *neurons*, *domain*, *plant* that have general sense and are not connected to the "Sepsis". Note that the feature PRC is an abbreviation and can be decrypted in different ways and, thus, it was marked as a good negative example. For example, in relation to sepsis PRC can mean a member of PGC family of transcriptional co-activators that coordinates the upregulation of mitochondrial biogenesis (Sweeney et al., 2010). PRC level increases with the presence of inflammation caused by sepsis. In medicine PRC can also be interpreted as "plasma renin concentration" or "phase response curve" that shows biological responses to light and to exogenous melatonin in animals. Also, the abbreviation PRC can be related to many other domains like geography (PRC is People's Republic of China commonly known as China), botany (PRC is photosynthetic reaction centre), education, etc..

Please note, that the feature year appears many times in the top features in the positive context models. For example, for the terms "Sepsis", "Tricuspid Atresia", "T-Lymphocyte Subsets", "Mycobacterium Aviumintracellulare Infection", "Coccyx", "Hermaphroditism". Apparently, for the mentioned terms, the year is the most important criteria for classification. Probably, during these years, the terms were highly discussed in biomedical domain. We analyzed the trend line of publications over the time period for terms with number of hand annotated documents less than 10000, thus we are sure that we cover whole time period of publications. For these terms, the feature year appears more than 3 times in top 10 of positive features. The trends of the terms are provided by GoPubMed. The trend lines for terms "Tricuspid Atresia" and "omega-Agatoxin IVA" are presented in Figure 3.5. The smoothed trend line (dark gray line) shows the relative growth of publications in comparison to the growth of the whole PubMed. As it is indicated in

Sepsis	Tricuspid Atresia	Merkel Cells	Dentate Gyryus	Breast Cyst	T-Lymphocyte Subsets	Mycobacterium Avium-intracellulare Infection	Coccyx	omega-Agatoxin IVA	Hermaphroditism
year before 1998	year before 2003	year before 2006	p=dentate	p=lesion	year before 1999	year before 2000	year before 2005	year before 2005	year before 2002
year before 1999	year before 2002	p=sensory	p=dentate_gyryus	t=cystic	p=CD4+_CD8+	year before 1999	year before 2004	year before 2004	year before 2001
p=bacteremia	year before 2001	p=20	p=granule	year after 2006	year before 2001	year before 2004	year before 2003	year before 2003	year before 2003
year before 2000	year before 2000	p=pattern	year before 2001	p=carcinoma	year before 2000	year before 2003	year before 2002	p=channel	p=gonadal
t=septicemia	year before 2004	p=mechanical	t=hippocampal	p=malignant	year before 2002	year before 2002	year before 2001	p=Ca2+	year before 2004
year before 2001	p=connection	p=sinus	year before 2000	p=ultrasound	p=CD4+_cells	year before 2001	year before 2006	p=calcium	year before 2005
year before 2002	p=shunt	p=express	p=gyryus	t=report	t=subsets	t=Mycobacterium	p=anomalies	p=micromM	t=elegans
year before 2006	p=like	p=perception	year before 2002	p=lesions	p=thymic	year before 2005	p=vertebral	t=calcium	p=Caenorhabditis
p=septicaemia	year before 1999	p=stimulation	p=hippocampal	p=classification	year before 2003	p=Mycobacterium	p=bovine	t=channels	p=Caenorhabditis_elegans
year before 2003	p=venous		p=cell_granule	p=cystic	p=T-lymphocyte	t=AIDS	p=spine	p=blocked	year before 2006

Table 3.4: Top 10 positive features of the terms

Sepsis	Tricuspid Atresia	Merkel Cells	Dentate Gyryus	Breast Cyst	T-Lymphocyte Subsets	Mycobacterium Avium-intracellulare Infection	Coccyx	omega-Agatoxin IVA	Hermaphroditism
p=Here	p=cells	p=potential	p=lung	p=function	p=cognitive	p=brain	p=health	p=patient	p=energy
p=PRC	p=cell	p=model	p=pulmonary	p=receptor	p=motor	p=pressure	p=diseases	p=approach	p=inflammatory
p=social	p=effects	p=binding	p=bacteria	p=structures	p=pH	p=stress	p=C	p=therapy	p=improved
p=neurons	p=mechanism	p=genes	p=cancer	p=protein	p=plants	p=transfer	p=species	p=years	p=pain
p=signaling	p=differences	p=cancer	p=health	p=low	p=oxidation	p=injury	p=model	p=research	p=injury
p=domain	p=formation	p=subjects	p=plant	p=order	p=spectroscopy	p=apoptosis	p=models	year after 2004	p=improve
p=strategy	p=evaluated	p=treatment	p=kidney	p=These	p=pressure	p=pregnancy	p=compounds	p=risk	p=drug
p=synthesized	p=organ	p=acid	p=selection	p=evidence	p=materials	p=nucler	p=receptor	p=clinical	p=oxygen
p=plant	p=does	p=genetic	p=hospital	p=model	p=channels	p=expression_gene	p=individuals	p=molecules	p=synthesized
p=compounds	p=60	p=scale	p=metal	p=best	p=oxygen	p=energy	p=elderly	p=tumor	p=inhibitor

Table 3.5: Top 10 negative features of the terms

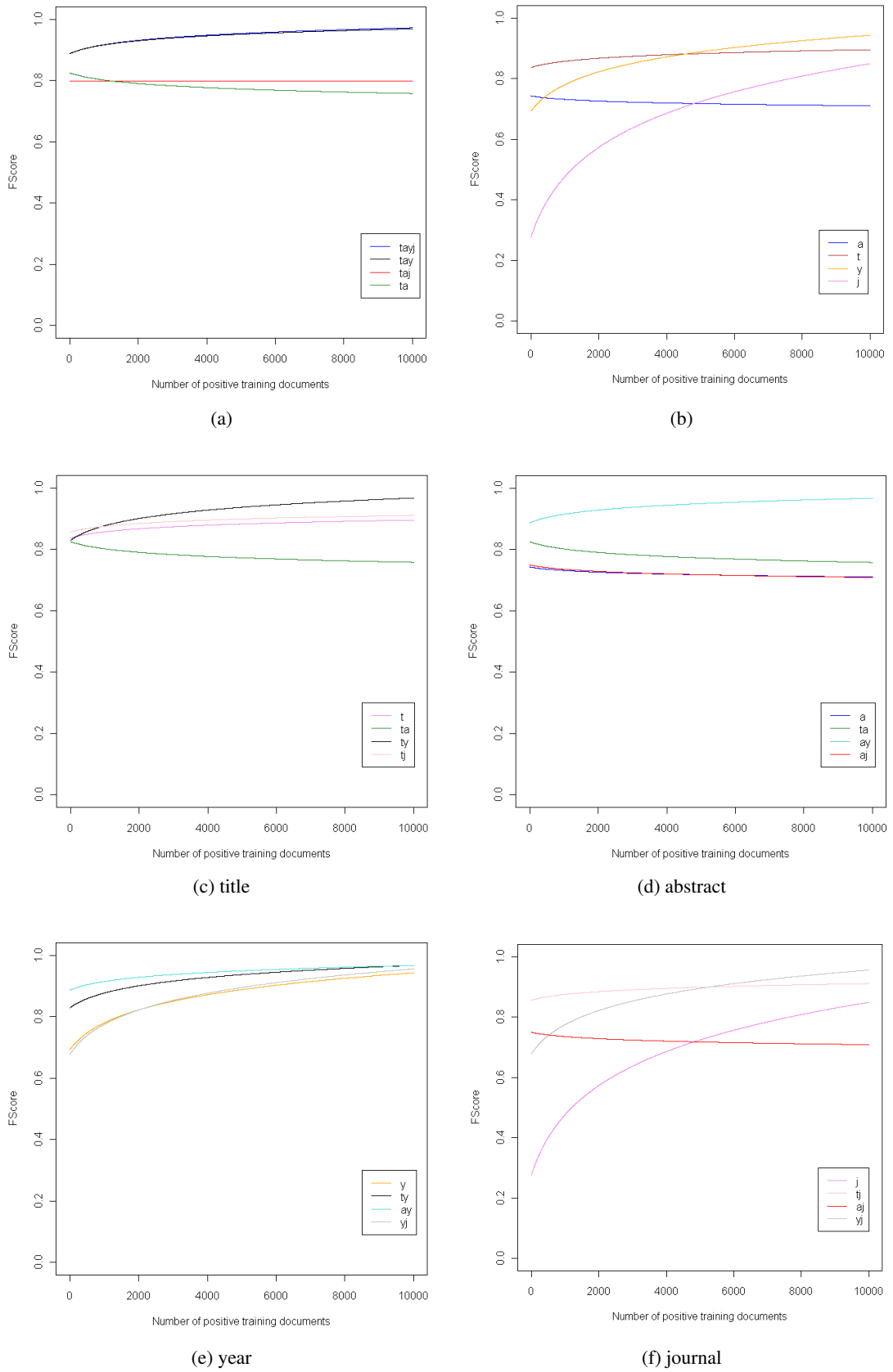


Figure 3.4: Features combinations

positive features, the most researches for term "Tricuspid Atresia" were done before 1999. The term "omega-Agatoxin IVA" had high interest before year 2004, but after 2004 till nowadays the publication number decreases. This can explain the fact that the feature "year after 2004" is in negative list. Thus, the proposed assumption about high interest in a term during specified years is possible to be a true assumption.

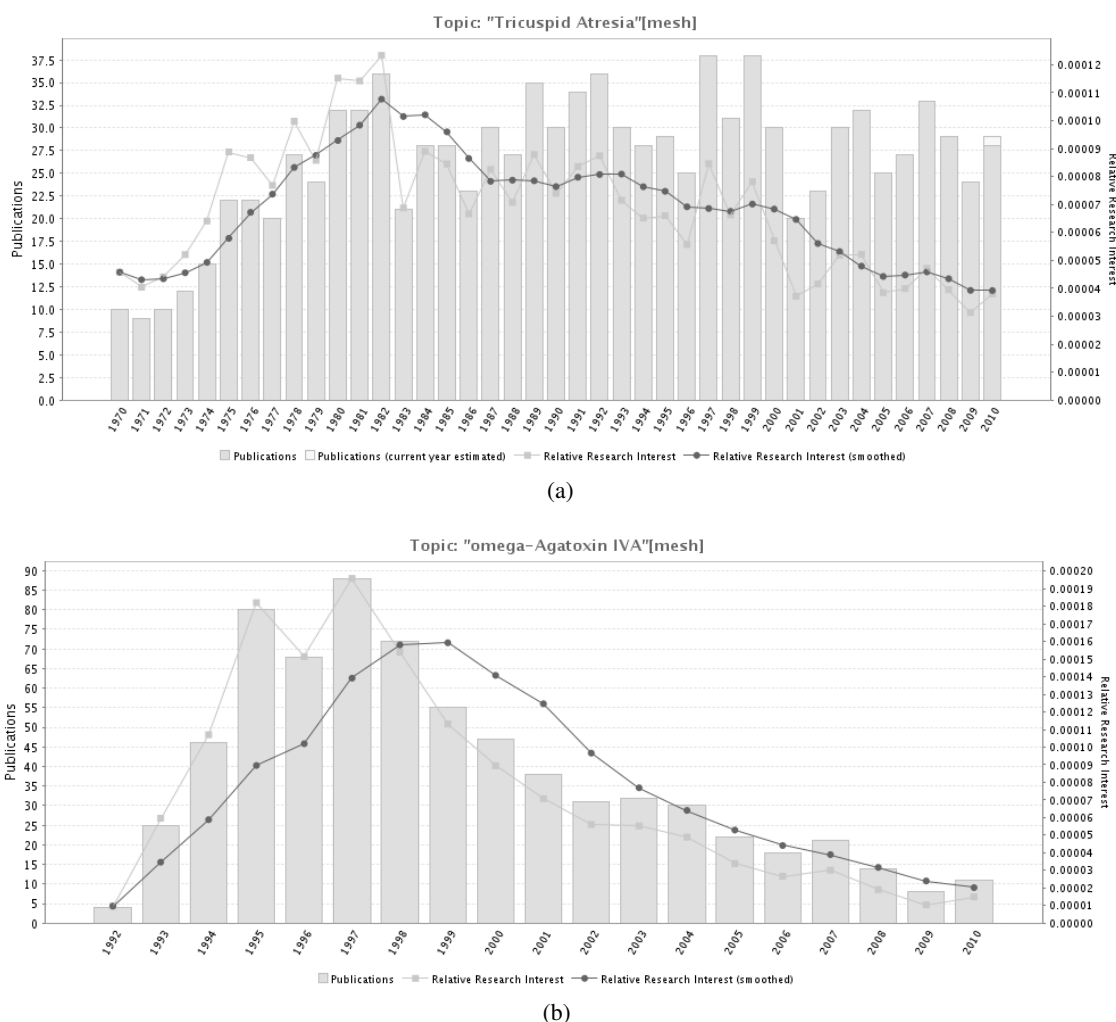


Figure 3.5: Trend line of publications over the time for (a) Tricuspid Atresia and (b) omega-Agatoxin IVA.

Beside the fact that the algorithm assigns high probability to the year feature, there are examples that indicate wrong feature classification. For example, MeSH term "Merkel Cells". The features *pattern*, *20*, *sinus*, *express*, *like* were assigned to positive vector while they contain general sense and must be used as negative features.

The negative feature *cancer* might be considered as a classification error, because Merkel cells can turn malignant and form skin tumor, which leads to cancer. However, cancer is a very common word in PubMed, using it as positive feature may lead to many false positives.

Conclusion. In this experiment we analyzed the influence of the term features on the algorithm performance. Different combinations of features were applied in order to examine the classification process. The features title and abstract are the most informative and, thus, were assumed to be more relevant for the algorithm efficiency. Moreover, the feature year is essential to achieve the best results; it is needed for the context models construction.

The analysis showed that using only the abstract corresponds to the lowest performance due

to the presence of noise inside abstracts. The abstract in combination with the title gives a little improvement, but still the trend line remains with a negative slope. The combination abstract-year gives the highest performance of the classifier. The title trend line has less improvement in combination with journal than with year. Thus, the year gives a significantly larger improvement in the performance than the journal. Furthermore, the feature journal is essential if the feature year is not used, otherwise it can be considered as less important. As a result, we consider the features title, abstract, year as important features for the classification process, while the journal is considered to be insufficient. However, the experiment was run on 10 MeSH terms, which is insufficient for making final conclusion. For more precise results we need to test the algorithm on more terms.

Possible explanations for the high influence of the year are the following. Firstly, the term can have a trend in publications during particular years. We showed this for the terms, where the training set covered all publications for a MeSH term over time. Secondly, the data sample is biased for MeSH terms and for some years. This is the case for terms that have more than 10000 hand annotated documents (terms: Sepsis, Tricuspid Atresia).

The analysis of context models for the used 10 MeSH terms shows partial correctness in the classification. The feature year appears in the top 10 positive features that is affirmed by the highest probability assigned to the year. This fits to the first explanation mentioned above.

The correctness of the classification procedure depends on the positive and negative examples. The positive examples are hand annotated PubMed documents that were manually annotated by human indexers and, thus, they correspond to highly reliable data. The negative examples are randomly chosen by the algorithm among 20 millions of PubMed articles. Since we are sure of the correctness of positive examples and the accurate algorithm classification (Section 3.3.2), the negative examples must be analyzed thoroughly. The research of the negative examples is done in the next section.

3.3.5 Negative Training Data

The aim of this experiment is to analyze the influence of the negative training dataset to the algorithm performance. As mentioned in the previous section, the algorithm uses the positive and negative sets of documents to create context models for each MeSH term. The positive documents are the MeSH hand annotated documents. The set of negative documents were randomly extracted out of all PubMed documents. The results of such combination can be seen in Section 3.3.2. The algorithm performs well, even though it makes mistakes in the creation of feature vectors that characterize the terms (Tables 3.4 and 3.5). The reason for this might be the choice of the negative documents.

Question 5. How should the negative examples be selected?

Experiment. The experiment involves the application of 10-fold cross-validation on 10 randomly selected MeSH terms with accuracy threshold 0.6. In order to analyze the influence of negative training dataset on the behavior of the algorithm, we specify the four options of negative dataset can be selected.

- Option 1. Random selection.
- Option 2. Selection from the positive examples.
- Option 3. Selection from the random abstracts that contain term literally.
- Option 4. Selection from the random abstracts that contain term literally, but are semantically distant from the term.

In Option 1, the negative examples are extracted randomly from the PubMed documents. Due

to the fact that currently PubMed comprises more than 20 million documents, the probability of choosing a negative example as positive is quite small. This option was considered in the work above (Sections 3.3.2 – 3.3.4). In Option 2, the negative examples for the term are extracted from the positive examples of this term that are the hand annotated documents. In Option 3, the negative examples for the term are extracted randomly from PubMed abstracts that contain this term literally. These documents were extracted from PubMed with the query *"term" NOT "term"[mh:noexp]*, such queries retrieve documents that contain the term literally but removes documents with this MeSH term. The Option 4 is similar to the Option 3, but with the requirement that the selected abstracts are semantically distant to the explored term. The obtained set of negative training documents has similar syntax, but different semantics. For measuring the semantic distance between MeSH terms, we applied our own metric as described in Section 2.8.

Results. For each option, we applied 10-fold cross-validation with classification delta 0.1. In order to evaluate the success of the algorithm on each case, we calculated precision, recall and F-score measures (Section 2.4). Table 3.6 shows the average F-score results for the four options of choice of the negative examples.

	Option 1	Option 2	Option 3	Option 4
Average F-score	0.94	0.16	0.74	0.72

Table 3.6: Average F-score measured for 4 options of selection negative examples.

The highest f-score (94%) is achieved with random selection of the negative examples. The performance of the algorithm using random set of the negative examples can be seen in the previous sections. This option is considered to be the easiest one for choosing the negative documents, since the number of negative documents that contain term literally must be significantly small and the similarity between the positive and negative sets will be low. The classifier will give high probability to the features that occur in the positive set and low probability to those in the negative. The analysis of negative documents showed that 60 documents (0.6%) out of 10000 contain literally the MeSH terms (Section 3.3.1).

Option 2 is considered to be the hardest option (16%) for creation the classification model. It is hard to determine negative features from a positive set of documents. This test case is designed to measure the case, where there is no distinction possible and successful distinction happens by chance.

Options 3 and 4 are supposed to create more robust models than in option 1, due to the fact that it will be harder to classify the negative features in the set that contains term names and their synonyms. The semantic distance, in option 4, indicates the exclusion of the words that are highly connected to the examined terms. These words can be referred as synonyms of the terms.

An interesting experiment was done in order to analyze the robustness of the models from options 1, 3, 4. The algorithm was rerun for these options. The 10-fold cross-validation was applied for the evaluation of the models. The positive data is a set of MeSH hand annotated documents that is stable for each MeSH term. The negative data is one of the specified options. The experiment for option 1 involved the following: the negative examples from option 1 were used as a training dataset for the model, and the negative examples from option 3 or 4 were used as a testing data. The same experiments were done with options 3 and 4.

Due to the fact that option 1 showed the highest performance (94%) in training and testing on the same dataset, we expect to get f-score above 90% in the case of testing the trained model from option 1 on the datasets from option 3 and 4. We have similar expectations about option 3 and 4 (f-score above 70%). The results of the experiments are collected in the Table 3.7 percentagewise. The combination of options 3 and 4 are out of our interest due to the similarity of the results with

		Test		
		Option 1	Option 3	Option 4
Train	Option 1	94	67	67
	Option 3	64	74	-
	Option 4	64	-	72

Table 3.7: Training and testing model with different combination of negative options. The results are given percentage-wise.

the option 3 and option 4 separately.

In the diagonal cases there is an explosion of the performance, especially with option 1-1, which is rational due to the training and testing of same datasets. In these cases, the algorithm tries to model the domain better by learning all the details of the dataset. As a result, the model is overtrained with the data and cannot be generalized on another test data.

Let us have a view on the combinations for options 1 and 3. The performance of the model from option 1-3 (trained with option 1 and tested with option 3) drops down to 67%. This makes around 30% of variance from the optimal f-score. The high discrepancy in the averaged f-scores indicates the overfitting of the initial model (option 1-1) with the used dataset. In this case the model cannot be used as a general model for the biomedical domain. The model of option 3-3 gives 74%. The result drops to 64% if the model is tested on the dataset of option 1. The variance of 10% allows us to consider that the model of option 3 is robust. Analogous results are obtained with option 4.

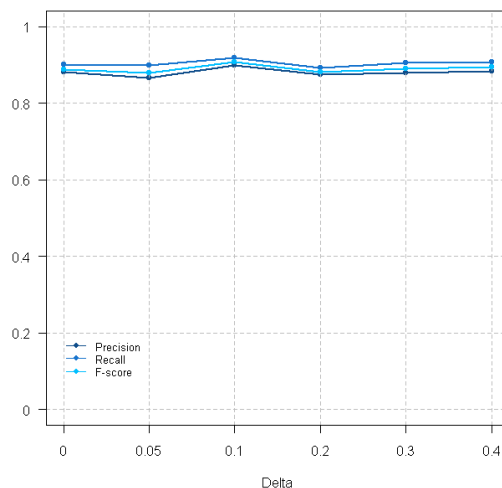
From the Table 3.7, it can be seen that whenever the option 1 is used as training or as testing data, there is a large variance in the results (train set : from 94% to 67%, test set: from 94% to 64%). Option 3 and 4 produce more robust models for training than option 1 and, thus, should be used as a negative datasets.

Conclusion. The choice of negative examples and its influence on the algorithm performance were analyzed in this section. The testing of four options for selection of the negative examples showed that optimal f-score (94%) is reached while using random set of the negative examples (option 1). High performance and big variance in combination with other options indicates the model overfitting with used dataset. The models of options 3 and 4 behave as expected and are considered to be robust. For the future classification we should use negative examples that were selected by options 3 or 4.

3.3.6 Re-Examination of Negative Training Data

In the previous experiments, the negative examples for context models were extracted randomly. Because of the significant improvement of the algorithm trend line after adding the year feature, we analyzed the publication years in the negative dataset. The analysis showed that 68% of the articles were published in the year 2007, 23% in 2006, 6% in 2008, and 3% in other years. Thus, negative documents were not equally spread among the whole period of publications and were biased to some period of time. Apparently, this is one of possible reasons for the great influence of the term feature year to the f-score value.

We compiled the new set of negative data that was extracted randomly from 20 million PubMed documents. We selected documents published in between 1970 and 2009. The documents were equally spread with respect to the growth of the whole PubMed. We rerun the experiment for the second question (Section 3.3.2) and evaluated the algorithm on different validation thresholds. Figure 3.6a shows the measured performance of the algorithm for the different threshold levels. The optimal threshold value is 0.6 ($\delta = 0.1$). With this threshold the algorithm achieves the maximum f-score value of 90,7% with precision of 89,8% and recall of 91,8% (Table 3.6b). Thereby,



(a)

delta	precision	recall	f-score
0	0.880	0.900	0.888
0.05	0.865	0.898	0.879
0.1	0.898	0.918	0.907
0.2	0.873	0.892	0.880
0.3	0.879	0.904	0.889
0.4	0.883	0.908	0.893

(b)

Figure 3.6: Analysis of precision, recall and f-score for classification delta in range [0, 0.05, 0.1, 0.2, 0.3, 0.4] for new set of random selected negative documents: (a) - visualization of average values, (b) - statistical information for average values

the decision to use the threshold value of 0.6 was a good choice in the previous work.

We checked the correctness of the context model using the new negative dataset. As before, the combination of all term features was applied. We explored the top 10 of the features from positive and negative feature vectors. The results are provided in the Tables 3.8 and 3.9 respectively. A gray cell indicates the correct feature in relation to the term, a yellow cell - incorrect and a white cell - feature year. Let us consider the MeSH term "Sepsis". Among the positive features for this term are *sepsis*, *septicaemia*, *bacteremia* which are synonyms for sepsis. The features *staphylococcal*, *meningococcal* and *yersinia* characterize the infectious diseases that are subkinds of sepsis. Furthermore, the feature *neonatal* is correctly chosen by the algorithm, because there exists a subkind of sepsis called neonatal sepsis. The features *blood isolated* and *patients sepsis* can be classified as correct because the phrase *blood isolated* contains *blood* and the presence of bacteria in the blood is bacteremia that is synonym for sepsis, meanwhile the phrase *patients sepsis* includes term sepsis. Negative features are words with general sense that could be applied to other terms as well.

Generally, the context models became more precise according to features that characterize the term. As before, the algorithm makes around two mistakes in top 10 positive features. Feature year rarely appears in the positive list. Terms Breast Cyst and Coccyx have 'year before 1971' positive feature. The publication of articles for the Coccyx started in 1902 year and for Breast Cyst it started in 1947. There were 137 published articles out of 744 for Coccyx and 43 articles out of 140 for Breast Cyst till 1970. Apparently, these articles are indeed connected to the terms and describe their characteristics. The articles published after 1970 year can also contain describing

information, but also use terms as reference information in the experiments or the conclusions. Feature year appears often in the negative list. The relation between the year and the publication trend of the terms were checked with GoPubMed analogously with Section 3.3.4.

From Figure 3.7, it can be seen that the influence of year became lower than in Section 3.3.4 due to the normal distribution of publication year in the negative dataset. The trend lines of combinations for all features and title-abstract-year coincide, making the feature journal unessential with the presence of year. Still the improvement of performance with journal is much less than performance with a year. The features abstract and title have the same behavior as before.

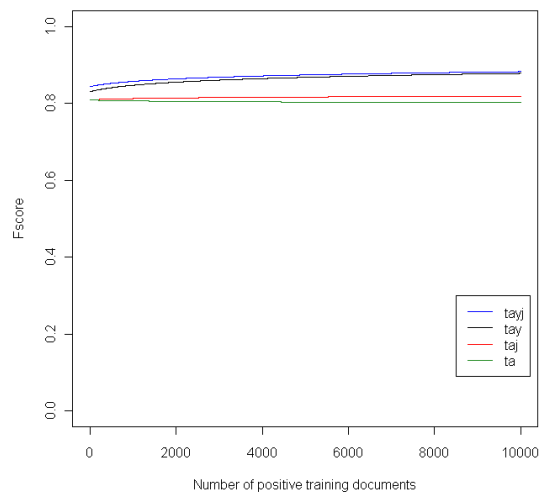


Figure 3.7: Features combinations

Conclusion. The abnormal behavior of the algorithm and the high influence of the feature year in the previous sections is caused by the unequal distribution of publication year in the used negative dataset. The experiment with new sample of negative examples showed that the algorithm achieves the optimal f-score of 90% with the validation threshold 0.6. This result corroborates the previously made decision to use the validation threshold of 0.6 in all the experiments. The context models turned to be more precise in terms of the features that characterize the MeSH terms. The influence of the feature year became lower. However, the feature year still gives the higher improvement in the performance than the feature journal. The features title, abstract, year can be considered the important features for the classification process, while the journal can be considered to be less important.

The general behavior of the Concept Recognition algorithm was evaluated and the correctness of its performance was analyzed in the Section 3.3. The analysis of the algorithm performance with respect to the ambiguity of biomedical concepts is done in the next section.

3.4 Application for disambiguation

Ambiguity is a serious problem in the semantic interpretation. Natural language contains a lot of ambiguous words. A word can be defined as ambiguous if it can be well interpreted in more than one way (Section 2.9). The meaning of the sentence depends on the defined senses of the used words. For example, the term "Light" (meshID: 8027) can mean (i) an electromagnetic radiation of a wavelength that is visible for the human eye, (ii) a source of light (a celestial body, device, candle), (iii) truth, (iv) not heavy (in relation to the volume), (v) not dark, etc.. The words like "Light" induce lexical ambiguity in the phrases or sentences in which they occur. In most of the cases, the correct sense of the ambiguous word can be defined from the domain of the context that

Sepsis	Tricuspid Atresia	Merkel Cells	Dentate Gyryus	Breast Cyst	T-Lymphocyte Subsets	Mycobacterium Avium-intracellulare Infection	Coccyx	omega-Agatoxin IVA	Hermaphroditism
p=bacteremia t=septicaemia t=sepsis p=septicaemia t=staphylococcal t=Yersinia t=meningococcal p=blood_isolated p=patients_sepsis t=Neonatal	p=pulmonary p=artery p=atrial p=flow p=valve p=right p=connection p=rare p=ventricular p=late	t=cells p=skin p=receptors p=microscopy p=innervation p=peripheral p=carcinoma p=channel p=mice p=receptor	p=dentate p=dentate_gyrus p=gyrus p=cells_granule p=hippocampal t=hippocampus p=granule p=cell_layer p=sprouting p=hippocampus	p=breast t=breast year before 1971 p=cystic year before 1974 year before 1973 year before 1972 p=ultrasound p=contrast p=optical	t=subsets p=T_lymphocytes p=CD4+_cells t=CD4+ t=T_lymphocytes p=f-cell_subsets p=CD4+ p=CD8+ p=MHC_class t=subset	p=mycobacterial t=complex t=mycobacterial t=AIDS t=disseminated p=tuberculosis p=azithromycin t=AIDS_patient p=baacilli p=ciprofloxacin	p=rectal p=disc year before 1971 p=excision p=vertebral p=bladder p=tail p=vertebrae t=tail p=congenital	p=channel p=voltage p=nM p=presynaptic p=blockers p=calcium p=release p=synaptic t=calcium p=current	p=gonad p=Caenorhabditis t=sexual p=Caenorhabditis_elegans p=XX t=Caenorhabditis t=Caenorhabditis_elegans t=gonadal p=androgen t=elegans

Table 3.8: Top 10 positive features of the terms for new random selected negative documents

Sepsis	Tricuspid Atresia	Merkel Cells	Dentate Gyryus	Breast Cyst	T-Lymphocyte Subsets	Mycobacterium Avium-intracellulare Infection	Coccyx	omega-Agatoxin IVA	Hermaphroditism
t=health p=chromosome t=work p=plants p=compond p=peptides p=componds p=nuclous p=spectra t=breast	year before 1994 year before 1993 year before 1992 year before 1991 year before 1990 year before 1989 year before 1988 year before 1987 year before 1986 year before 1985	p=common p=discussed p=events p=clinical p=potential p=agents p=negative year before 1996 p=baseline p=gestation	year before 1994 year before 1993 year before 1992 year before 1991 year before 1990 year before 1989 year before 1995 year before 1988 year before 1987 year before 1986	p=protein t=patients p=symptoms p=7 p=componds p=sensitive p=phase p=controls p=paper p=children	year before 1987 year before 1988 year before 1989 year before 1990 p=CD27(-) p=Ag_transfer p=CMV_T_specific p=energy p=SCCHN p=NED	p=rats p=tumors year before 1989 p=efficiency p=social p=cardiovascular p=pressure p=heart p=chromosome p=III	p=active p=vascular year after 2007 p=energy p=h p=strains p=sequence p=exposure p=fat p=health	year before 1993 p=years p=syndrome p=24 p=factors p=months t=study p=cancer p=patients t=analysis	p=rats p=rat t=acute p=administered p=insulin t=acid p=lipid p=cancer p=mg p=hybrid

Table 3.9: Top 10 negative features of the terms for new random selected negative documents

contains it. In the biomedical domain, a term is ambiguous if it appears in at least two places in the MeSH tree. For example, the term "Light" is located both in the "Electromagnetic Phenomena" and in the "Optical Phenomena" branches of the "Physical Phenomena" MeSH category.

In Section 3.3 we evaluated the general behavior of the Concept Recognition algorithm in the context of specified MeSH terms. The set of hypotheses were checked and the results were stated.

The second group of the experiments will now contain the analysis of the algorithm regarding ambiguous MeSH terms. The remaining hypotheses will be analyzed and we will use lexical WordNet and Wikipedia to distinguish "likely" ambiguous terms.

3.4.1 PubMed and GoPubMed vs Yahoo

As mentioned before (Section 3.2), PubMed is a biomedical literatures database that grows exponentially in its size. The submitted scientific articles to PubMed and the annotation of these with the MeSH terms have a similar pattern, meaning the identical trend of progress (Figure 3.1a). GoPubMed is a semantic search engine that explores PubMed for answering biomedical questions (Doms, 2008). The statistical information about the frequency of documents presence obtained from biomedical search engines PubMed and GoPubMed and a generic search engine *Yahoo! Search*¹ was compared and analyzed. By October, 2010 Yahoo was indicated to be the second largest web search engine on the web by query volume (App, 2010).

Hypothesis 6. The terms that are more general in its sense will have a large number of documents in PubMed, GoPubMed and Yahoo.

Experiment. For the examination we used the MeSH terms from the following MeSH categories: Anatomy, Diseases, Psychiatry and Psychology. The total number of used MeSH terms is 6654. The number of term occurrences in PubMed, GoPubMed and Yahoo were calculated for each MeSH term.

Results. The experiment involves the analysis of the results received by querying the MeSH terms in PubMed, GoPubMed and Yahoo. The results are represented in a graphical and statistical interpretation in Figure 3.8. Each dot on the scatter plots represents one MeSH term. Both axes of the scatter plots are logarithmic scaled allowing a better visualization of the results. The majority of the terms are concentrated in the center of the plot. However, note that the scale of biomedical domain and Yahoo have big difference (Tables 3.8c - 3.8d). This indicates the fact that the capacity of Yahoo considerably exceeds the capacity of PubMed. Each MeSH term has more results in Yahoo than GoPubMed can find for this term in the PubMed database. The presence of the "garbage" pages in the final result set of Yahoo is the explanation of this fact. "Garbage" pages correspond to (i) the documents that are partially connected to the queried MeSH term, (ii) the documents that contain the modified name of the term, (iii) the documents that contain the modified sense of the term as part of the speech. Option (i) refers to the MeSH terms that have a complex name like terms "Chromosomes, Human, Pair 10" or "Central Nervous System Parasitic Infections". Yahoo expands the user's query for term "Chromosomes, Human, Pair 10" into "Chromosomes OR Human OR Pair OR 10", so the returned documents contain at least one of the words from term's name. Option (ii) refers to the MeSH terms with a specific name. For example, the name of the term "Dourine" (meshID: 4313) is transformed into "During". In this case Yahoo provides the combined results for both queries "Dourine" and "During". Option (iii) refers to the MeSH terms that are homonymous. For example, term "Bear" (meshID: 1503) can be interpreted as a noun or verb.

The correlation coefficients are weak (PubMed vs Yahoo - 0.21, GoPubMed vs Yahoo - 0.28) that indicates a weak linear relation between biomedical engines and Yahoo concerning the literal

¹<http://search.yahoo.com/>

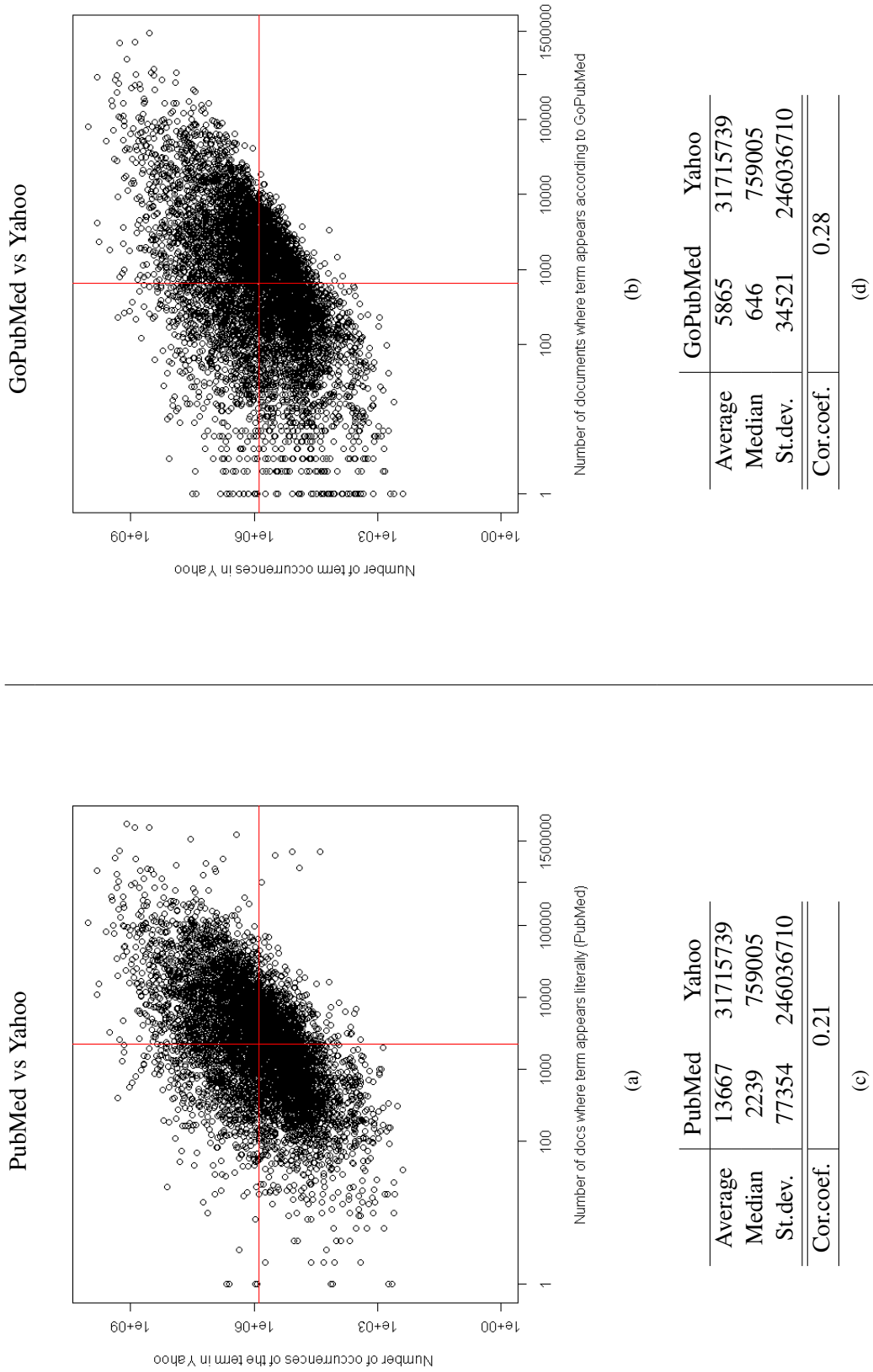


Figure 3.8: Correlation between the number of term occurrences in Yahoo and the number of documents where the term appears according to (a) PubMed and (c) GoPubMed. Axis X corresponds to the PubMed / GoPubMed extracted documents, axis Y - to the Yahoo. Both axes are log scaled. Medians are shown as red lines. Statistical measurements for (b) PubMed vs Yahoo and (d) GoPubMed vs Yahoo are calculated in the number of documents.

occurrence of MeSH terms. The presence of the big number of extreme points, so called outliers, strongly influences the correlation coefficient.

As stated above, most of the MeSH terms are concentrated in the middle of the scatter plots. The range of concentration in Yahoo is [2 000, 40 mio] and [50, 20 000] in PubMed and GoPubMed. For the further observations we split the scatter plot (Figure 3.8b) into 9 approximate regions in order to analyze the distribution of the terms in accordance with the number of senses in the WordNet and Wikipedia. According to the stated hypothesis, we expected that general terms will be located in the area that is closer to the right upper corner. Also, the MeSH terms were divided into several groups according to the values in WordNet and Wikipedia. The following groups were defined:

- (0,0) - terms with no entry in WordNet and Wikipedia,
- (0,1) - terms with one entry only in Wikipedia,
- (1,0) - terms with one entry only in WordNet,
- (1,1) - terms with one entry in both lexica,
- (x,y) - terms that have more than one meaning in WordNet and/or Wikipedia.

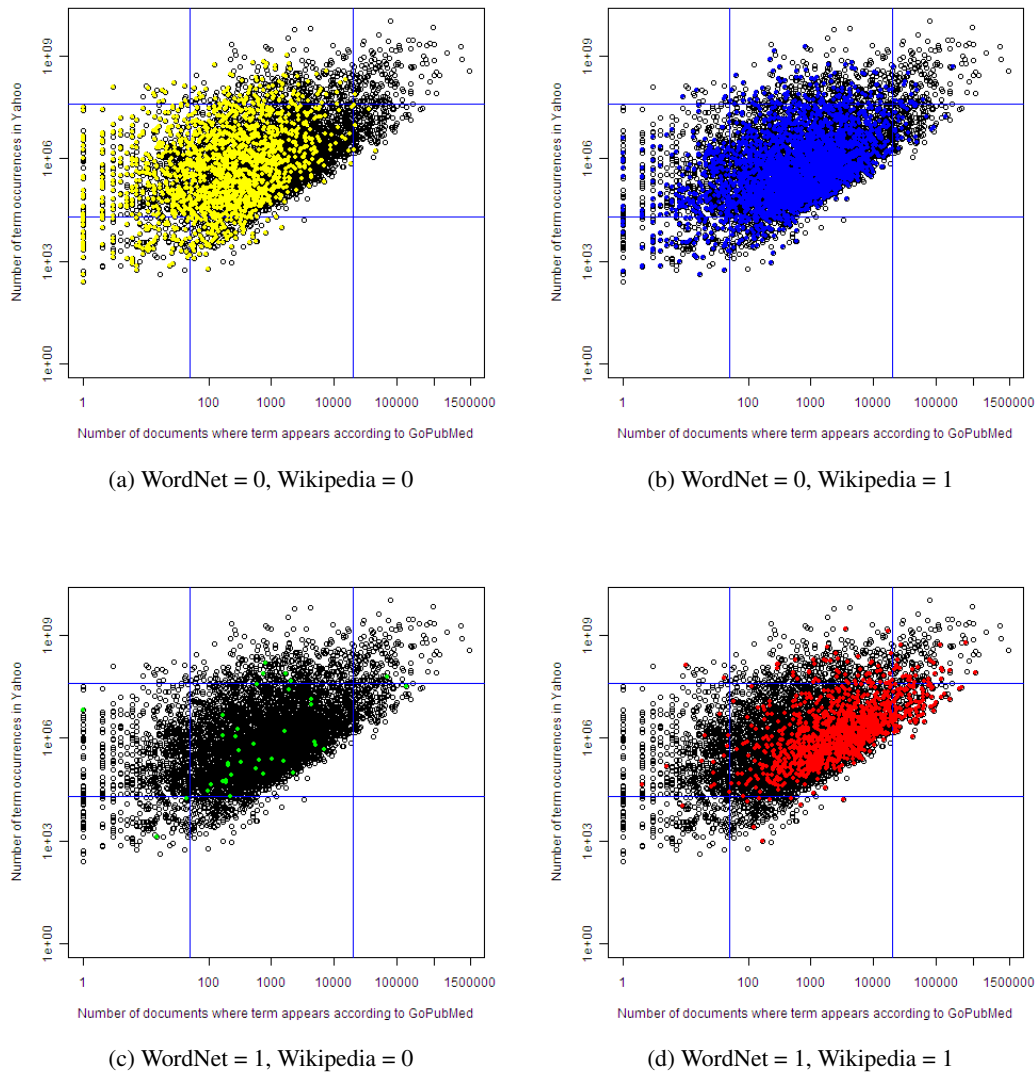


Figure 3.9: Combination of number of senses in WordNet and Wikipedia in range 0, 1.

Figure 3.9 shows the scatter plots for different combinations of senses number in the WordNet

and Wikipedia in the range $\{0,1\}$. The terms with no sense in WordNet and Wikipedia are closer to the left side of the scatter plot (Figure 3.9a). The terms with a single sense in both lexica are closer to the right side of the scatter plot (Figure 3.9d). The rest of the terms that have a single sense in WordNet or Wikipedia are concentrated in the center (Figures 3.9b- 3.9c). It is interesting to notice that half of the examined terms has the combination of senses number of (0,1). The total number of such terms is 3093. Only few terms have opposite combination.

Despite the fact that WordNet and Wikipedia are not biomedical oriented lexica, we infer that most the terms with none or single sense in the used lexica are not ambiguous. The number of senses indicates the specific biomedical interpretation of the terms. Besides, the probability of having a complex name among these terms is high.

The MeSH terms with more than one sense in WordNet and/or Wikipedia are visualized in the Figure 3.10. These terms are colored in black color. The total number of these terms is 481. The majority of these terms lies in the right upper corner of the scatter plot. This indicates that terms have large number of documents in GoPubMed and Yahoo. The examples of these terms are provided in the Table 3.10. The given terms have a large number of senses in both lexica. More precisely, these terms have meanings in different domains and, thus, can be referred to be general terms in their sense. The obtained results validate the stated hypothesis.

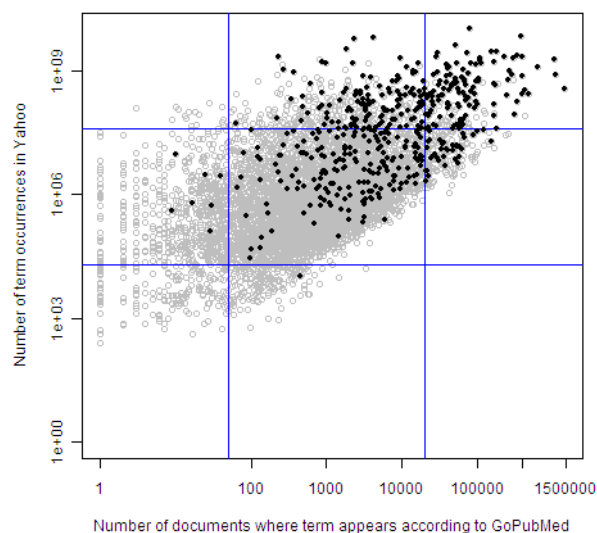


Figure 3.10: Analysis of terms with more than one sense in WordNet and Wikipedia. These terms are colored in black color.

Notice that there are also terms that are not together with the majority of the terms in the Figure 3.10. The position of these terms is closer to the lower left corner which implies fewer results in GoPubMed. Table 3.11 contains the examples of these terms. The explanations for this fact are the following:

- The MeSH term does not appear literally in the abstract or title of the PubMed article.
- The MeSH term has a complex name that contains several words and punctuation marks. For example, term "Dominance-Subordination" (meshID: 4291). This term can appear in forms of "Dominance-Subordination" and "Dominance Subordination" in the text of the article. After stemming, these forms are transformed into "dominancesubordin" and two stems "domin" and "subordin" respectively. The received transformations are not equal to

TermId	Term Name	PM	GPM	Yahoo (millions)	WordNet	Wikipedia
6257	Head	226 045	151 449	2 800	32	25
5145	Face	100 973	72 138	2 700	14	10
6225	Hand	251 626	177 501	2 900	14	6
4328	Drive	78 061	29 791	2 500	12	11
5528	Foot	73 081	39 898	1 100	11	6
6321	Heart	871 019	342 217	2 300	10	15
3863	Depression	214 382	88 292	396	10	10
20521	Stroke	138 395	92 956	265	10	6
1415	Back	109 140	79 813	10 000	9	9
5190	Family	594 980	374 201	6 700	8	22

Table 3.10: An example of terms with more than one sense in WordNet and/or Wikipedia. Information about the number of documents in PubMed, GoPubMed, Yahoo and the number of senses in WordNet and Wikipedia is also provided.

each other. As a result, the MeSH terms with complex name structure may require a special treatment.

- GoPubMed does not support different languages recognition. For example, term "Deja Vu" (meshID: 3690) is not recognized in majority of the PubMed articles if it is written in French manner in the text.
- The existence of PubMed articles with empty abstract. For example, the term "Callosities" (meshID: 2145).
- The MeSH term name coincides with the author's name. For example, the terms "Pinta" (meshID: 10874), "Koro" (meshID: 16911).
- GoPubMed does not support abbreviation recognition. For example, the term "Bundle-Branch Block" (meshID: 2037) has abbreviation "BBB".

Conclusion. In this experiment we analyzed the number of senses in the lexica WordNet and Wikipedia with respect to the documents retrieved by Yahoo, PubMed and GoPubMed. The total number of the analyzed MeSH terms is 6654. As a result, the majority of the MeSH terms have no sense in WordNet and one entry in Wikipedia. Usually, these terms have median number of the documents in PubMed and Yahoo. The analysis of these terms showed that they cannot be ambiguous due to the presence of specific biomedical interpretation and complex name structure. The small number of senses in the used lexica indicates the stated facts.

The rest of the MeSH terms (481 terms) have a large number of documents in PubMed and Yahoo. These terms have more than one sense in WordNet and/or Wikipedia. The terms with large number of senses in both lexica are located closer to the upper right corner of the scatter plot. Despite the fact that WordNet and Wikipedia are not biomedical oriented lexica, we can assume that these terms might be ambiguous. These terms have large number of meanings in different domains and, thus, can be referred to be general terms in their sense. The validity of the stated hypothesis is proved.

Notice that inaccuracy in the obtained results is caused by many reasons that must be taken into account in further experiments. The reasons are the difference in the scale and the domain specification of Yahoo and PubMed, "garbage" pages in Yahoo, complex names of MeSH terms and empty PubMed abstracts, the occurrence of MeSH terms in the full text of the article, the existence of abbreviations and terminology in the languages other than English.

TermId	Term Name	PM	GPM	Yahoo (millions)	WordNet	Wikipedia
21184	Nut Hypersensitivity	140	9	0.4	0	2
18602	Milk Sickness	18	10	9	2	1
4291	Dominance-Subordination	1 635	17	0.6	3	0
18614	Sweating Sickness	36	25	3	2	1
3690	Deja Vu	427	64	53	1	5
44905	beta-Mannosidosis	103	98	0.03	2	1
10874	Pinta	283	103	36	0	2
16911	Koro	216	124	13	0	2
850	Anomie	400	134	2	2	1
5535	Foot Rot	412	134	7	2	1
20960	omega-Conotoxins	750	450	0.01	2	0
6181	H-Reflex	2 574	1 434	0.09	5	1
2145	Callosities	575	188	0.13	1	2
2037	Bundle-Branch Block	3 134	612	2	0	1

Table 3.11: An example of the terms-outliers. Information about the number of documents in PubMed, GoPubMed, Yahoo and the number of senses in WordNet and Wikipedia is provided.

3.4.2 Term occurrence in MeSH hand annotated documents

Hypothesis 7. The MeSH hand annotations are often taken from the full text and don't appear literally in the title and abstract of the article.

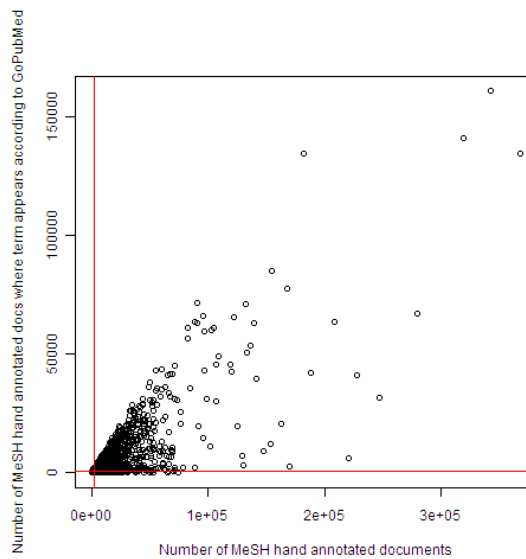
Experiment. For the examination we used the MeSH terms from the following MeSH categories: Anatomy, Diseases, Psychiatry and Psychology. The total number of retrieved MeSH terms is 6654. The number of the MeSH hand annotated documents in PubMed and GoPubMed were calculated and compared.

The PubMed database contains the manually MeSH annotated biomedical articles. The annotation corresponds to the manual annotation of the title, abstract and full text of the article. GoPubMed explores the PubMed as background knowledge. As a result, it retrieves the title and abstract of the article for the presence of the term in them.

Results. The experiment involves the comparison of the results obtained from the extraction of MeSH hand annotated documents from PubMed and GoPubMed. The results are visualized in Figure 3.11. Each dot on the scatter plot represents one MeSH term. The results show that almost all MeSH terms (97%) have more MeSH hand annotated documents in PubMed, than the number of hand annotated documents that were retrieved by GoPubMed. These results can be explained with the fact that PubMed articles have MeSH indexed title, abstract and full text, while GoPubMed explores only the title and abstract of the PubMed article, where the term does not appear often. These results proof the stated hypothesis.

The correlation coefficient between GoPubMed and PubMed is 0.82 that indicates the strong dependence among them. The dataset is normally distributed. The presence of outliers influences the correlation coefficient and lowers it.

Conclusion. The results of the experiment were based on the comparison of PubMed and GoPubMed according to the ability to retrieve the MeSH hand annotated documents. The obtained results showed that the manual annotation of the PubMed articles was done not only with the title and abstract, but also with the full text of the article. GoPubMed explores only the title and abstract



(a)

	PubMed	GoPubMed
Average	6124	1983
Median	1682	373
St.dev.	16159	6313
Cor.coef.	0.82	

(b)

Figure 3.11: Correlation between number of MeSH hand annotated documents in PubMed and GoPubMed: (a) graphical visualization of the experiment. Axis X corresponds to the PubMed extracted documents, axis Y - to the GoPubMed. Medians are shown as red lines. (b) statistical measurements of PubMed and GoPubMed calculated in number of documents.

of the PubMed article while searching the relevant knowledge. In 97% of MeSH terms the full text is required to be analyzed. Thus, the validity of stated hypothesis is proved.

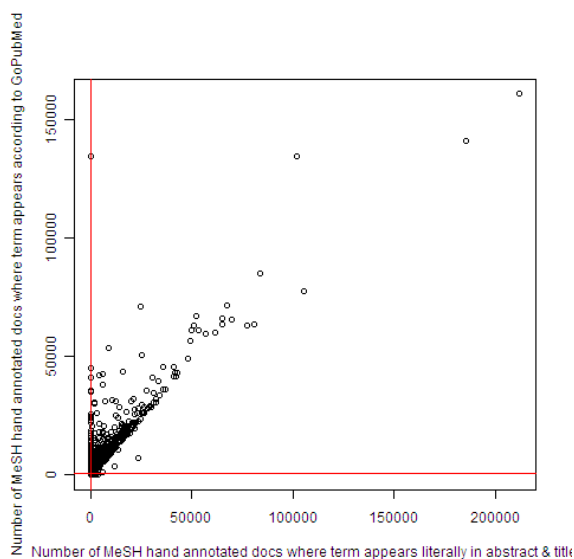
3.4.3 Term recognition by GoPubMed

Hypothesis 8. The MeSH hand annotations that appear literally in the title and abstract, don't appear according to GoPubMed.

Experiment. For the experiment we analyzed the performance of GoPubMed applied to the 6654 MeSH terms. For each MeSH term we calculated the number of annotated documents returned by GoPubMed. Additionally, the number of MeSH hand annotated documents, where the term appears in the title and abstract, was extracted for each term. This was done by querying the PubMed with "*term name*" [*tiab*]. The obtained results were compared.

Results. The results of the experiment are visualized in the Figure 3.12a. Each dot on the scatter plot represents one MeSH term. The results show that documents of almost all terms were returned by GoPubMed. Owing to the procedures of tokenization and stemming that are applied to the title and abstract of the article, GoPubMed algorithm recognizes more articles related to the given Mesh terms. The low performance of PubMed in providing the documents where terms appear literally in the title and abstract can be explained by the fact that PubMed does simple matching of the term's name in the title and abstract. Such search is less effective if the term

has a composite name or a name with punctuation symbols. As a result, Figure 3.12a shows the contradiction to the stated hypothesis.



(a) Correlation between the number of MeSH hand annotated documents where a term appears in the title and abstract in PubMed and the number of MeSH hand annotated documents recognized by the GoPubMed algorithm. Axis X corresponds to the PubMed extracted documents, axis Y - to the GoPubMed. Medians are shown as red lines.

	PubMed	GoPubMed
Average	1379	1983
Median	107	373
St.dev.	5837	6313
Cor.coef.	0.88	

(b) Statistical measurements of PubMed and GoPubMed calculated in number of documents

Figure 3.12: Correlation between PubMed and GoPubMed in MeSH hand annotated documents, and statistical measurements for it.

The correlation coefficient between the GoPubMed and PubMed in this experiment is 0.88 that shows a high dependence of two datasets. The data is normally distributed. The presence of outliers influences the correlation coefficient and lowers it. The outliers are terms that have more results in PubMed than in GoPubMed searches (Figure 3.13). The total number of these terms is 38. These terms were analyzed with a restriction: a term can be considered as outlier if the difference between the number of MeSH documents in PubMed and GoPubMed is more than 50% of the minuend, i.e.:

$$(\#PubMed - \#GoPubMed) > \frac{\#PubMed}{2}$$

The example of terms-outliers is given in the Table 3.12. Most of the reasons for the presence of these terms were mentioned in the Section 3.4.1. As before, these are the existence of abbreviations (term "Bundle-Branch Block" meshID:2037 is abbreviated with "BBB" in most of the PubMed articles), the existence of terminology in the languages other than English (term "Deja Vu" meshID: 3690), different variations of complex names of the MeSH terms (term "Self Efficacy" meshID: 20377 can appear in forms "Self Efficacy" or "Self-Efficacy" that have different results after stemming).

However, we should also take into account that in this experiment PubMed gave low performance in relation to the terms with complex names. The majority of the analyzed MeSH terms has

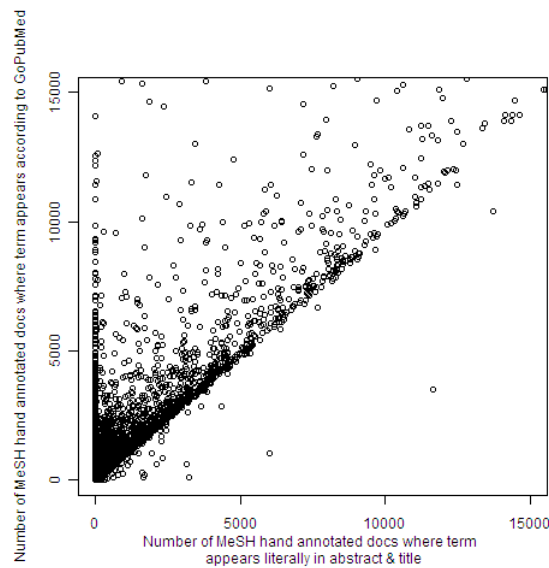


Figure 3.13: The correlation between MeSH hand annotated documents where term appears in title and abstract and MeSH hand annotated documents recognized by GPM algorithm (Figure 3.12) in range [0,15000].

composite name with punctuation marks in it. As a result, most of the PubMed documents were not retrieved back.

TermId	Name	PM	GPM	Reason
13601	T-Lymphocytes	23172	6953	Stem
1402	B-Lymphocytes	11659	3490	Stem
20377	Self Efficacy	3218	90	Stem
2037	Bundle-Branch Block	3134	612	Abbr. "BBB"
6463	Hemolytic-Uremic Syndrome	2325	849	Stem / Abbr. "HUS"
20275	Guillain-Barre Syndrome	1614	305	Language / Stem
6086	Graft vs Host Disease	971	249	Stem
3410	Cri-du-Chat Syndrome	223	105	Stem / Abbr. "CCS"
18813	Multiple Endocrine Neoplasia Type 2a	195	31	Abbr. "MEN2A"/"MEN type 2a"
16111	Sjogren-Larsson Syndrome	133	15	Language / Abbr. "SLS"
10381	Pelger-Huet Anomaly	130	65	Language / Abbr. "PHA"
16098	Gerstmann-Straussler-Scheinker Disease	97	22	Language / Abbr. "GSS"
19080	Cafe-au-Lait Spots	90	14	Language / Stem / Abbr. "CAL"
20331	Mobius Syndrome	69	19	Language / Stem
3690	Deja Vu	68	9	Language
20232	Kluver-Bucy Syndrome	44	15	Stem / Abbr. "KBS"
18370	Leukocyte-Adhesion Deficiency Syndrome	12	1	Stem / Abbr. "LADS"

Table 3.12: An example of terms that are outliers from Figure 3.13 and stated reason why they are considered to be outliers. Information about the number of documents found in PubMed and GoPubMed is provided.

Conclusion. The results of the experiment contradict the stated hypothesis. GoPubMed provides a higher performance in recognizing relevant articles that are related to the given MeSH term than PubMed does. The low performance of PubMed can be explained by the fact that PubMed does simple matching of the term's name in the title and abstract of the article. The matching process is less effective if the term has a complex name. GoPubMed, in turn, shows high performance

due to the application of the tokenization and stemming procedures.

3.4.4 WordNet and Wikipedia

In Section 3.4.1 the hypothesis about the ambiguity of MeSH terms with multiple numbers of documents in PubMed and Yahoo was proved. Such MeSH terms have a large number of meanings in different domains and, thus, can be referred to be general terms in their sense. The lexical thesauri WordNet and Wikipedia were used to analyze the ambiguity of these MeSH terms.

This section will analyze the used thesauri in relation to 6654 MeSH terms and check if WordNet and Wikipedia are correlated and agree in their number of senses for MeSH terms.

Hypothesis 9. There is a correlation among thesauri WordNet and Wikipedia.

Experiment. The analysis of the 6654 MeSH terms was done according to the number of senses that they have in WordNet and Wikipedia. The senses for MeSH terms with multiple senses were analyzed manually in order to combine similar definitions. At the end, the correlation analysis between a number of senses in WordNet and Wikipedia was done.

Results. The experiment involves the analysis of the number of terms' senses in the lexical thesauri WordNet and Wikipedia. The results are based on the experiments done by Tsatsaronis and Torge (2010). In the analysis of senses for the given 6654 MeSH terms, the authors divided the MeSH terms into several groups according to the number of senses they have in WordNet and Wikipedia. The following groups were defined:

- terms with no entry in WordNet and Wikipedia (27%)
- terms with an entry only in WordNet (1%)
- terms with an entry only in Wikipedia (47%)
- terms with entries in WordNet and Wikipedia (25%)

The visualization of the defined groups is given in the Figure 3.14. The total number of MeSH terms that don't have any senses in WordNet and Wikipedia is 1799 (27%). The example of such terms can be terms "Pulmonary Infarction" (meshID: 54060), "Foot Joints" (meshID: 33023).

The total number of terms that have an entry only in WordNet is 49 (1%). Among them 41 terms have only one sense in WordNet (for example, terms "Blastodisc" (meshID: 54239), "Tooth Root" (meshID: 14092)) and 8 terms have more than one sense (terms "Rupture" (meshID: 12421), "Toxemia" (meshID: 14115)).

The majority of terms have an entry only in Wikipedia (3137 terms), that indicates that the capacity of Wikipedia exceeds the capacity of WordNet. These terms can be further split into the following groups:

- terms with exactly one sense in Wikipedia (3093 terms). For example, term "T-Lymphocytes, Cytotoxic" meshID: 13602.
- terms with more than one sense (44 terms). For example, terms "Facies" (meshID: 19066) and "Skin Diseases" (meshID: 12871).

The last group of terms represents the terms with entries in WordNet and Wikipedia (1669 terms). Among them, 1240 terms have only one sense in both lexica. For example, terms "Sepsis" (meshID: 18805), "Aorta" (meshID: 1011). The remaining 429 terms have more than one sense in WordNet and/or Wikipedia. For example, terms "Face" (meshID: 5145) and "Back" (meshID: 1415). According to the ambiguity criteria (Section 2.9), we will refer to these terms as ambiguous. They compose 6.4% of the total of analyzed terms. Due to the fact that WordNet and Wikipedia

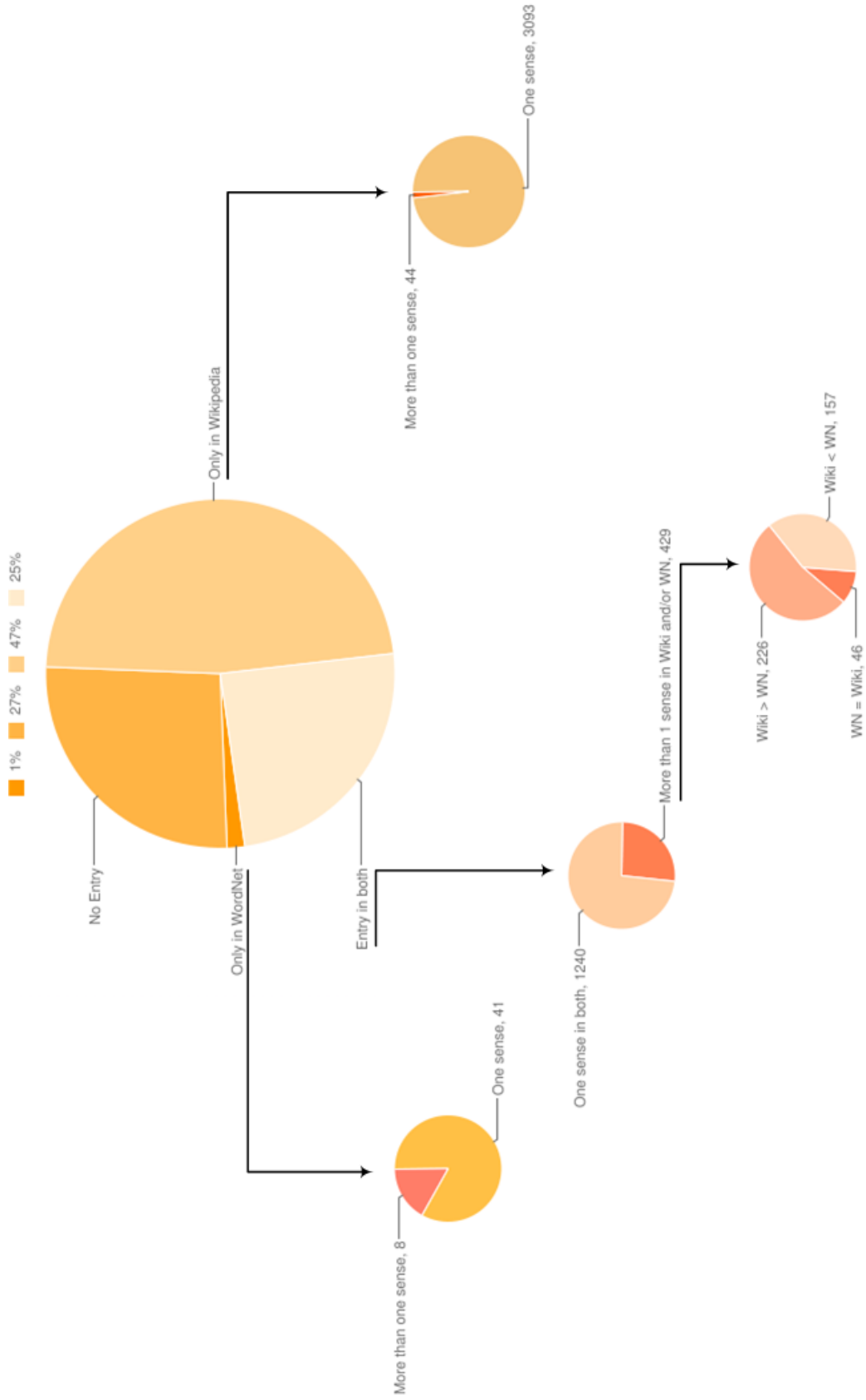


Figure 3.14: Visualization of number of MeSH term-meanings in WordNet and Wikipedia.

are not biomedical oriented thesauri, the senses of MeSH terms with multiple meanings were manually corrected. Especially this concern the senses returned by Wikipedia because it is based on volunteer work of users without scientific background. The Wikipedia's disambiguation page for a given biomedical term returns a list of the possible meanings the term can have related to the biology, medicine, linguistics, entertainment (music, films, books, and games), etc. Similar meanings were combined in groups that reduced the number of senses and made the correlation results more objective.

The number of terms in relation to the number of senses returned by both thesauri is visualized in Figure 3.15. The histogram demonstrates that WordNet and Wikipedia are almost at the same level of the returned results. If a term has several meanings in WordNet, than it also has almost the same number of senses in Wikipedia. The correlation coefficient is 0.7, which indicates a strong correlation between the thesauri. This statement validates the stated hypothesis. Thus, WordNet and Wikipedia agree between each other in the number of senses.

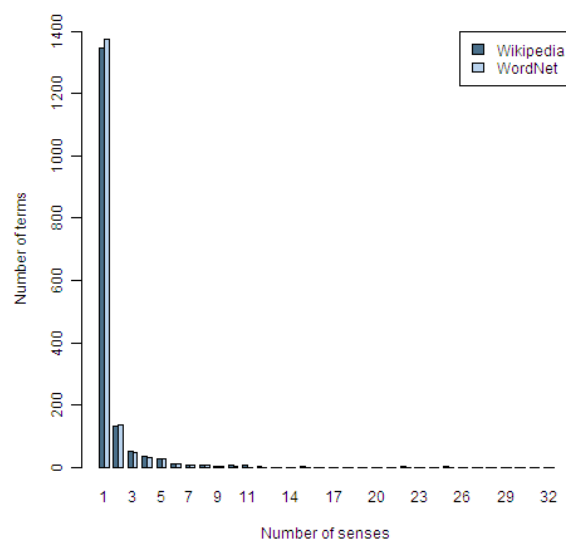


Figure 3.15: Agreement between Wikipedia and WordNet in the number of senses extracted for each MeSH term.

The analysis of the correctness of WordNet and Wikipedia was done by Tsatsaronis and Torge (2010) in order to determine if we can trust to the thesauri in the detection of ambiguous terms. The authors manually examined 200 MeSH terms such that 50 terms had no entry in both lexica, 50 terms had exactly one entry in WordNet and/or Wikipedia and 100 terms had more than one entry in WordNet and/or Wikipedia. The results for the first group showed that 16 terms had entries in other dictionaries (for example, The Free Dictionary²), 14 terms had articles in Wikipedia and WikiMiner didn't find it, 12 terms were ambiguous and 20 terms had no entry in any explored dictionaries. In the second group, only 2 terms were ambiguous. The results for third group showed that 6 terms were found to be non-ambiguous among the 100 ambiguous terms. In overall, WordNet and Wikipedia return nearly the correct number of senses. They can be referred as informative knowledge bases for the terms with one or few senses. However, in the case of no or multiple senses, an additional thesaurus that specializes in biomedical and health related concepts should be used. The metathesaurus Unified Medical Language System³ (UMLS) should be used for managing the ambiguity problem.

According to Wagner (2009) the number of words in a MeSH term affects the f-score. The

²<http://www.thefreedictionary.com/>

³<http://www.nlm.nih.gov/research/umls/>

more specific a term is, the more words it contains. The average number of words that a term has in 6654 MeSH terms is 2.1, that makes two words in a name. The averaged f-score is 0.92. The terms with 2 words in the name can be also considered as specific. Only 2% of these terms has multiple senses in WordNet or Wikipedia. Due to the fact that the majority of terms has 2 or more words, the average f-score value is high. Almost all of the possible ambiguous MeSH terms have one word-name. However, Tsatsaronis and Torge (2010) researched the variance of f-score in relation to the number of senses in WordNet or Wikipedia. The results showed that there is no correlation between the algorithm performance and the number of senses of a MeSH term. Around 75% of possibly ambiguous terms has f-score more than 90%. The f-score for the rest 25% varies between 75% - 90%. The general specification of used thesauri needs to be combined with additional thesaurus that specializes in biomedical domain, like UMLS, for detecting the ambiguous MeSH terms.

Conclusion. In this experiment we explored the strong correlation between the WordNet and Wikipedia. The analysis done by Tsatsaronis and Torge (2010) showed the correctness of the returned results from the thesauri. Despite this, an additional thesaurus that specializes in biomedical domain needs to be used in order to lighten the detection of ambiguous biomedical concepts. An example of such thesaurus is the metathesaurus UMLS.

The number of senses of a MeSH terms does not correlate with the algorithm performance. The variation of the f-score values or possible ambiguous terms is between 75% - 98%. The usage of UMLS will decrease the range of variation by eliminating the unambiguous MeSH terms among the ambiguous.

Chapter 4

Conclusion

The main contribution of this thesis lies in the evaluation of the concept recognition approach intended for managing the ambiguity problem of search engines. It is based on Maximum Entropy method that is widely used for solving disambiguation tasks. The approach identifies the ontology (MeSH) concepts in the PubMed abstracts and generates context models that characterize these concepts. The evaluation consists of two subtasks: the general analysis of the algorithm's performance and the analysis of the algorithm performance in relation to the ambiguous terms.

Improving the work done in Macari (2010), the algorithm obtained maximum f-score of 91% (precision 90%, recall 92%) with validation threshold value of 0.6. The optimal size of the training data was set to 5000 documents which is sufficient for good prediction ability of the classifier. The analysis of the term features showed that the features title, abstract and year are important to achieve high results, especially the feature year. It brings a significant improvement in the algorithm's performance and it is an essential feature in the context models construction. The choice of the negative examples plays an important role in the classification process. In order to avoid overfitting and overestimation of the model, the negative training set should consist of: (i) random abstracts that contain explored term literally or (ii) random abstracts that contain explored term literally, but are semantically distant from the term. Both options create robust model, due to the fact that it is harder for the classifier to recognize the negative features in the set that contains the term name and its synonyms.

The second part of the analysis shows that 6.4% of the explored MeSH terms are ambiguous terms which imply having more than one sense in WordNet and Wikipedia. In fact, such terms have a large number of documents in PubMed and Yahoo. The annotation process of the biomedical articles consists in the annotation of the title, the abstract and the full text of the article. This disables GoPubMed to return as many results as PubMed does. However, GoPubMed shows high performance in exploring the title and abstract while searching the articles that contain term literally. The analysis of the concept recognition algorithm shows that its performance does not correlate to the number of senses of a MeSH term. The f-score variation for ambiguous terms is between 75% - 98%. Due to the fact that WordNet and Wikipedia are not biomedical oriented thesauri, the additional metathesaurus Unified Medical Language System should be used for the recognition of the ambiguous MeSH terms.

4.1 Future Work

Though the general analysis of the algorithm performance showed the high results, the recognition of the synonyms must be taken into account for future research. In the MeSH thesaurus, the synonyms are grouped together with the terms and are not stored as separate concepts. For this reason, an additional tool that discovers the semantic relations between the terms and synonyms needs to be applied for building a high quality classification model. As mentioned in Section 3.4.4, the

ambiguity level of the biomedical terms should be measured by applying the additional thesaurus UMLS together with the WordNet and Wikipedia.

By now, the biomedical domain comprises many resources, and the majority of those have not been used yet. Currently, we are using resources that are integrated into the UMLS thesaurus. The methods that rely on the usage of background knowledge (for example, ontologies) have several disadvantages:

- the coverage always depends on the information awareness and the size of the ontology
- the granularity of the ontologies, for example, ontologies can be fat or tall.
- the compatibility of the resources, i.e. not all resources contain the same information on the same level.

For the further research, we are planning to improve the algorithm performance by increasing the background knowledge with additional ontologies or by applying additional methods based on the statistical information extracted from the text (for example, the PubMed corpus). Both options will augment the background knowledge with new information, but the second option will add statistical information that is not included in the ontologies (for example, frequency of ontological terms in text, number of senses for non-ontological concepts).

For the improvement of the terms' ambiguity recognition in the biomedical domain, the concept recognition algorithm needs to be applied in combination with the knowledge bases and the clustering indices done by Tsatsaronis and Torge (2010). Using knowledge bases (WordNet, Wikipedia and UMLS), we can evaluate the terms ambiguity level according to the definition, while the usage of the clustering indices gives an estimation of the terms ambiguity according to their appearance in the PubMed documents. This combination is being used as a part of the approach that automatically assesses and quantifies the ambiguity of ontological terms and evaluates the influence of the terms' ambiguity on the text classification procedure.

Bibliography

- Search engine market share. Net Applications, October 2010. URL <http://marketshare.hitslink.com/search-engine-market-share.aspx?qprid=4>.
- Cross-validation report (analysis services - data mining). Microsoft TechNet Library, 2010. URL <http://technet.microsoft.com/en-us/library/bb895177.aspx>.
- S. Abe. *Support Vector Machines for Pattern Classification*. Springer, 2 edition, March 2010.
- S. Abu-Nimeh, X. Wang, and S. Nair. Abstract a comparison of machine learning techniques for phishing detection.
- D. Altman and J. Bland. Statistics notes: Diagnostic tests 1: sensitivity and specificity. In *British Medical Journal*, June 1994. URL <http://www.bmj.com/content/308/6943/1552.full.pdf>.
- A. Aronson, J. Mork, F. Lang, W. Rogers, and A. Neveol. Nlm medical text indexer: A tool for automatic and assisted indexing. Technical report, US National Library of Medicine, April 2008.
- 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, M. A. Harris, D. P. Hill, L. Issel-Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock. Gene ontology: tool for the unification of biology. the gene ontology consortium. *Nature genetics*, 25(1):25–29, May 2000. URL <http://dx.doi.org/10.1038/75556>.
- D. Barbella, S. Benzaid, J. Christensen, B. Jackson, X. V. Qin, and D. Musicant. Understanding support vector machine classifications via a recommender system-like approach.
- Z. N. Baskin I.I., Palyulin V.A. Application of artificial neural networks in chemical and biochemical research. pages 323–326, 1999.
- K. P. Bennett, L. Auslender, D. Wu, and S. Ave. On support vector decision trees for database marketing. Technical report, Department of Mathematical Sciences Math Report No. 98-100, Rensselaer Polytechnic Institute, 1998.
- A. L. Berger, V. J. D. Pietra, and S. A. D. Pietra. A maximum entropy approach to natural language processing. *Comput. Linguist.*, 22(1):39–71, 1996.
- G. Beri. *Business Statistics*. TATA MC Graw HILL, 3 edition, 2009.
- M. H. Biglu. The editorial policy of languages is being changed in medline, 2007. URL <http://scielo.sld.cu/pdf/aci/v16n3/aci06907.pdf>.
- R. C. Blattberg, B.-D. Kim, P. do Kim, and S. A. Neslin. *Database marketing: analyzing and managing customers*. 2008.

- O. Bodenreider, A. Burgun, and J. Mitchell. Evaluation of wordnet as a source of lay knowledge for molecular biology and genetic diseases: a feasibility study. *Stud Health Technol Inform*, 95: 379–384, 2003.
- A. Borthwick. *A Maximum Entropy Approach to Named Entity Recognition*. PhD thesis, New York University, September 1999.
- A. Bresell. Interpretation of microarray expression data using ontology browsing. Master’s thesis, Linköpings Universitet, Linköping, Sweden, 2002. URL https://people.ifm.liu.se/andersb/papers/LiTH-IDA-ex-02-76_oneside.pdf.
- C. J. Burges. A tutorial on support vector machines for pattern recognition, 1998.
- J. Cannady. Artificial neural networks for misuse detection. In *National Information Systems Security Conference*, pages 443–456, 1998.
- E. Charniak and R. Goldman. A semantics for probabilistic quantifier-free first-order languages, with particular application to story understanding. In *IJCAI’89: Proceedings of the 11th international joint conference on Artificial intelligence*, pages 1074–1079, San Francisco, CA, USA, 1989. Morgan Kaufmann Publishers Inc.
- X. Chen, M. M. Hoffman, J. A. Bilmes, J. R. Hesselberth, and W. S. Noble. A dynamic bayesian network for identifying protein-binding footprints from single molecule-based sequencing data. *Bioinformatics*, 26(12):334–342, 2010.
- C. Cherry. Review of hidden markov models in bioinformatics, 2001. URL <http://webdocs.cs.ualberta.ca/~colinc/cmput606/606FinalPres.ppt>.
- L. De Ferrari and S. Aitken. Mining housekeeping genes with a naive bayes classifier. *BMC genomics*, 7:277, October 2006. URL <http://dx.doi.org/10.1186/1471-2164-7-277>.
- D. Dementhon, D. Doermann, and M. V. Stükelberg. Hidden markov models for images. In *International Conference on Pattern Recognition*, 2000.
- A. Demiriz, K. P. Bennett, and J. Shawe-Taylor. Linear programming boosting via column generation. *Machine Learning*, 46(1-3):225–254, 2002.
- P. Domingos and M. Pazzani. Beyond independence: Conditions for the optimality of the simple bayesian classifier. In *Machine Learning*, pages 105–112. Morgan Kaufmann, 1996.
- A. Doms. *GoPubMed: Ontology-based literature search for the life sciences*. PhD thesis, Technical University of Dresden, October 2008.
- A. Doms and M. Schroeder. Gopubmed: exploring pubmed with the gene ontology. *Nucleic Acids Research*, 33:783–786, 2005.
- M. Dorrer, A. Gorban, and V. Zenkin. Neural networks in psychology: classical explicit diagnoses. In *Second International Symposium on Neuroinformatics and Neurocomputers*, pages 281–284, Rostov on Don, Russia, 1995.
- H. Drucker, S. Member, D. Wu, S. Member, and V. N. Vapnik. Support vector machines for spam categorization. *IEEE Transactions on Neural Networks*, 10:1048–1054, 1999.
- B. Efron. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7:1–26, 1979. URL <http://dx.doi.org/10.2307/2958830>.

- C. Elkan. *Introduction to Artificial Intelligence*. University of California, San Diego, 2006. URL <http://cseweb.ucsd.edu/users/elkan/151fall2006/>.
- N. Fenton, M. Neil, and D. Marquez. Using bayesian networks to predict software defects and reliability, 2007.
- T. Fomby. Naive bayes classifier. April 2008.
- Y. Freund. An adaptive version of the boost by majority algorithm. *Machine Learning*, 43(3): 293–318, 2001.
- Y. Freund. A more robust boosting algorithm. May 2009. URL <http://arxiv.org/abs/0905.2138>.
- T. S. Furey, N. Cristianini, N. Duffy, D. W, and D. Haussler. Support vector machine classification and validation of cancer tissue samples using microarray expression data, 2000.
- E. Gabrilovich and S. Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 1606–1611, 2007.
- V. Gant. *Clinical Applications of Artificial Neural Networks*. Cambridge University Press, New York, NY, USA, 2001.
- T. R. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 5:199–220, 1993.
- D. Heckerman. A tutorial on learning with bayesian networks. Technical Report MSR-TR-95-06, Microsoft Research, November 1996.
- M. Hilario and A. Kalousis. Building algorithm profiles for prior model selection in knowledge discovery systems. *Systems, Man, and Cybernetics, 1999. IEEE SMC '99 Conference Proceedings. 1999 IEEE International Conference on*, 3:956–961, October 1999.
- P. R. Hinton. *Statistics explained*. 2 edition, 2004.
- A. Hliaoutakis. Semantic similarity measures in mesh ontology and their application to information retrieval on medline. Master’s thesis, Technical University of Crete, November 2005.
- T. Hofweber. Logic and ontology. In E. N. Zalta, editor, *The Stanford Encyclopedia of Philosophy*. Fall 2010 edition, 2010.
- W. R. Hutchison and K. R. Stephens. The airline marketing tactician (amt): A commercial application of adaptive networking. In *In Proceedings of the 1st IEEE International Conference on Neural Networks, San Diego, USA*, volume 2, pages 753–756, 1987.
- O. Intrator and N. Intrator. Robust interpretation of neural-network models, 1997.
- M. J. Islam, Q. M. J. Wu, M. Ahmadi, and M. A. Sid-Ahmed. Investigating the performance of naive- bayes classifiers and k- nearest neighbor classifiers. In *ICCIT '07: Proceedings of the 2007 International Conference on Convergence Information Technology*, pages 1541–1546, Washington, DC, USA, 2007. IEEE Computer Society.
- O. Ivanciuc. Applications of support vector machines in chemistry. *Reviews in Computational Chemistry*, 2007.
- A. W. Jerome L. Myers. *Research design and statistical analysis*, volume 1. Routledge, 2, illustrated edition, 2003.

- J. Jiang and D. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proc. of the Int'l. Conf. on Research in Computational Linguistics*, pages 19–33, 1997. URL <http://www.cse.iitb.ac.in/~cs626-449/Papers/WordSimilarity/4.pdf>.
- T. Joachims, F. Informatik, F. Informatik, F. Informatik, F. Informatik, and L. Viii. Text categorization with support vector machines: Learning with many relevant features, 1997.
- C. M. Jones, L. A. Darzi, and T. Athanasiou. Diagnostic tests and diagnostic accuracy in surgery. In *Key Topics in Surgical Research and Methodology*. Springer, 2010.
- M. W. Kadous. *Temporal Classification: Extending the Classification Paradigm to Multivariate Time Series*. PhD thesis, The University of New South Wales, School of Computer Science and Engineering, October 2002.
- R. Karchin. Hidden markov models and protein sequence analysis.
- C. Karlof, D. Wagner, C. Karlof, and D. Wagner. Hidden markov model cryptanalysis. In *In Cryptographic Hardware and Embedded Systems – CHES '03*, pages 17–30. Springer-Verlag, 2003.
- M. Kearns. Thoughts on hypothesis boosting. Unpublished manuscript, Dec. 1988.
- Z. Kim and R. Nevatia. Learning bayesian networks for diverse and varying numbers of evidence sets. In *Proc. Int'l Conf. on Machine Learning*, pages 479–486. Morgan Kaufmann, 2000.
- R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection, 1995.
- D. Koller and M. Sahami. Hierarchically classifying documents using very few words, 1997.
- A. Krogh. Two methods for improving performance of an hmm and their application for gene finding, 1997.
- T. Kudo and Y. Matsumoto. A boosting algorithm for classification of semi-structured text. IEIC Technical Report (Institute of Electronics, Information and Communication Engineers), 2004.
- R. Lacher. Can neural network computers learn from experience?, October 1999. URL <http://www.scientificamerican.com/article.cfm?id=can-neural-network-comput>.
- C. T. Le. *Applied Categorical Data Analysis and Translational Research*. 2009.
- C. Leacock, G. A. Miller, and M. Chodorow. Using corpus statistics and wordnet relations for sense identification, 1998.
- T. S. Levitt, J. M. Agosta, and T. O. Binford. Model-based influence diagrams for machine vision. In *UAI '89: Proceedings of the Fifth Annual Conference on Uncertainty in Artificial Intelligence*, pages 371–388, Amsterdam, The Netherlands, The Netherlands, 1990. North-Holland Publishing Co.
- D. D. Lewis. An evaluation of phrasal and clustered representations on a text categorization task. In *SIGIR '92: Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 37–50, New York, NY, USA, 1992. ACM.
- M. Li. *Sequence and Text Classification: Features and Classifiers*. PhD thesis, Ming Li, July 2006.

- B. Liu. *Web data mining: exploring hyperlinks, contents, and usage data*. 1 edition, 2007.
- L. Liu and M. T. Zsu. *Encyclopedia of Database Systems*. Springer Publishing Company, Incorporated, 2009.
- P. Lucas. Bayesian networks in medicine: a model-based approach to medical decision making, 2001.
- N. Macari. Analysis of the concept recognition algorithm. Student project, Technical University of Dresden, May 2010.
- G. Mann, R. McDonald, M. Mohri, N. Silberman, and D. Walker. Efficient large-scale distributed training of conditional maximum entropy models. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 1231–1239. 2009.
- A. McCallum and K. Nigam. A comparison of event models for naive bayes text classification, 1998.
- S. Merler, C. Furlanello, B. Larcher, and A. Sboner. Tuning cost-sensitive boosting and its application to melanoma diagnosis. In *MCS '01: Proceedings of the Second International Workshop on Multiple Classifier Systems*, pages 32–42, London, UK, 2001. Springer-Verlag.
- V. Metsis, I. Androutsopoulos, and G. Paliouras. Spam filtering with naive bayes – which naive bayes? In *Third Conference on Email and Anti-Spam (CEAS)*, California, USA, July 2006.
- G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. Miller. Wordnet: An on-line lexical database. *International Journal of Lexicography*, 3:235–244, 1990.
- D. Milne and I. H. Witten. An effective, low-cost measure of semantic relatedness obtained from wikipedia links. In *In Proceedings of AAAI 2008*, 2008.
- A. O. Mohamed Addin. *Bayesian Network Classifiers for Damage Detection in Engineering Material*. PhD thesis, Universiti Putra Malaysia, 2007.
- R. Nelson. Kendall tau metric. In *Encyclopedia of Mathematics*. Springer, 2001.
- A. Névéol, J. Mork, A. Aronson, and S. Darmoni. Evaluation of french and english mesh indexing systems with a parallel corpus. American Medical Informatics Association, 2005.
- K. Nigam. Using maximum entropy for text classification. In *In IJCAI-99 Workshop on Machine Learning for Information Filtering*, pages 61–67, 1999.
- L. Ohno-Machado. *Medical applications of artificial neural networks: connectionist models of survival*. PhD thesis, Stanford, CA, USA, 1996. Adviser-Musen, Mark A.
- E. Osuna, R. Freund, and F. Girosi. Training support vector machines: an application to face detection. pages 130–136, 1997.
- K. Pelckmans, J. A. K. Suykens, and B. D. Moor. Handling missing values in support vector machine classifiers.
- M. S. Pepe. *The statistical evaluation of medical tests for classification and prediction*. Oxford University Press, 2004.
- P. Peursum, H. H. Bui, S. Venkatesh, and G. West. Robust recognition and segmentation of human actions using hmms with missing observations. *EURASIP J. Appl. Signal Process.*, pages 2110–2126, 2005. URL <http://dx.doi.org/10.1155/ASP.2005.2110>.

- L. Rabiner and B. Juang. An introduction to hidden markov models. *IEEE Magazine on Acoustics, Speech and Signal Processing*, 3(1):4–16, April 1986.
- R. Rada, H. Mili, E. Bicknell, and M. Blettner. Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(1):17–30, Jan 1989.
- A. Ratnaparkhi. A maximum entropy model for part-of-speech tagging, 1996.
- A. Ratnaparkhi, J. Reynar, and S. Roukos. A maximum entropy model for prepositional phrase attachment. In *In Proceedings of the ARPA Workshop on Human Language Technology*, pages 250–255, 1994.
- P. Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *In Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 448–453, 1995.
- B. D. Ripley. *Pattern Recognition and Neural Networks*. Cambridge University Press, 2008.
- S. E. Robertson and K. S. Jones. Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27(3):129–146, 1976.
- M. Rochery, R. Schapire, M. Rahim, and N. Gupta. Boostexter for text categorization in spoken language dialogue.
- M. Rochery, R. Schapire, M. Rahim, N. Gupta, G. Riccardi, S. Bangalore, H. Alshawi, and S. Douglas. Combining prior knowledge and boosting for call classification in spoken language dialogue, 2002.
- M. Sadeghi, M. A. Sadeghi, S. Nourizadeh, A. M. Ranjbar, and S. Azizi. Power system security assessment using adaboost algorithm.
- I. Sanches. Noise-compensated hidden markov models. In *Speech and Audio Processing*, volume 8, pages 533 – 540. IEEE Signal Processing Society, September 2000.
- R. E. Schapire. The strength of weak learnability, 1990.
- R. E. Schapire. The boosting approach to machine learning: An overview, 2002.
- R. E. Schapire, Y. Singer, and A. Singhal. Boosting and rocchio applied to text filtering. In *In Proceedings of ACM SIGIR*, pages 215–223. ACM Press, 1998.
- M. Smith. *Neural Networks for Statistical Modeling*. John Wiley & Sons, Inc., New York, NY, USA, 1993.
- E. L. L. Sonnhammer. A hidden markov model for predicting transmembrane helices in protein sequences. In *In Proceedings of the 6th International Conference on Intelligent Systems for Molecular Biology (ISMB)*, pages 175–182. AAAI Press, 1998.
- T. E. Sweeney, H. B. Suliman, J. W. Hollingsworth, and C. A. Piantadosi. Differential regulation of the *pgc* family of genes in a mouse model of staphylococcus aureus sepsis. *PLoS ONE*, 5(7), July 2010. URL <http://dx.doi.org/10.1371/journal.pone.0011606>.
- G. Tsatsaronis and S. Torge. Measuring the ambiguity of terms in the biomedical domain and its effect in text classification. Technical report, Technical University of Dresden, June 2010.
- G. Tsatsaronis, I. Varlamis, and M. Vazirgiannis. Text relatedness based on a word thesaurus. *J. Artif. Intell. Res. (JAIR)*, 37:1–39, 2010.

- E. Wagner. Evaluation of a machine learning approach for ontology concept recognition and synonym prediction. Master's thesis, Technical University of Dresden, November 2009.
- R. E. Walpole, R. H. Myers, S. L. Myers, and K. Ye. *Probability and Statistics for Engineers and Scientists*. 8 edition, 2007.
- A. S. Weigend. On overfitting and the effective number of hidden units. In M. C. Mozer, P. Smolensky, D. S. Touretzky, J. L. Elman, and A. S. Weigend, editors, *Proceedings of the 1993 Connectionist Models Summer School*, pages 335 – 342, Hillsdale, NJ, 1994. Lawrence Erlbaum Associates.
- R. L. White. Object classification in astronomical images, 1996.
- I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 2 edition, 2005.
- Z. Wu and Z. Wu. Verb semantics and lexical selection, 1994.
- Q. Yu, Y. Miche, A. Sorjamaa, A. Guillen, A. Lendasse, , and E. Séverin. Op-knn: Method and applications. *Advances in Artificial Neural Systems*, 2010.
- J. Zhao, X. long Wang, Y. Guan, and L. Lin. Analyzing the incomplete data based on the improved maximum entropy model analyzing the incomplete data based on the improved maximum entropy model.
- M. Zou and S. D. Conzen. A new dynamic bayesian network (dbn) approach for identifying gene regulatory networks from time course microarray data. *Bioinformatics*, 21(1):71–79, 2005.