CONCEPT BASED KNOWLEDGE DISCOVERY FROM BIOMEDICAL LITERATURE

Aleksandar Radovanovic

Thesis presented in fulfillment of the requirements for the

Degree of Doctor Philosophiae

at the South African National Bioinformatics Institute

Faculty of Natural Sciences

University of the Western Cape

Advisor: Prof. Vladimir Bajic

May 2009

Declaration

I declare that "Concept Based Knowledge Discovery From Biomedical Literature" is my own work, that it has not been submitted before for any degree or examination at any other university, and that all the resources I have used or quoted, and all work which was the result of joint effort, have been indicated and acknowledged by complete references.

Aleksandar Radovanovic

May 2009

CONCEPT BASED KNOWLEDGE DISCOVERY FROM BIOMEDICAL LITERATURE

Aleksandar Radovanovic

KEYWORDS

Bioinformatics

Text mining

PubMed

Entity recognition

Information extraction

Relation Extraction

Levenshtein distance

Supervised classification

Natural Language Processing

Machine learning

Publications arising from this work

DDEC: Dragon database of genes implicated in esophageal cancer
Magbubah Essack, Aleksandar Radovanovic, Ulf Schaefer, Sebastian Schmeier, Sundararajan V. Seshadri, Alan Christoffels, Mandeep Kaur & Vladimir B. Bajic, *BMC Cancer 2009, 9:219*

Database for exploration of functional context of genes implicated in ovarian cancer
Mandeep Kaur, Aleksandar Radovanovic, Magbubah Essack, Ulf Schaefer, Monique Maqungo, Tracey Kibler, Sebastian Schmeier, Alan Christoffels, Kothandaraman Narasimhan, Mahesh Choolani and Vladimir B. Bajic, *Nucleic Acids Research, 2009, Vol. 37, Database issue D820-D823*

DDESC: Dragon database for exploration of sodium channels in human
Sunil Sagar, Mandeep Kaur, Adam Dawe, Sundararajan Vijayaraghava Seshadri, Alan Christoffels, Ulf Schaefer, Aleksandar Radovanovic and Vladimir B. Bajic, *BMC Genomics 2008, 9:622*

Table of Contents

Abstract

Advancement in biomedical research and continuous growth of scientific literature available in electronic form, calls for innovative methods and tools for information management, knowledge discovery, and data integration.

Many biomedical fields such as genomics, proteomics, metabolomics, genetics, and emerging disciplines like systems biology and conceptual biology require synergy between experimental, computational, data mining and text mining technologies. A large amount of biomedical information available in various repositories, such as the US National Library of Medicine Bibliographic Database, emerge as a potential source of textual data for knowledge discovery. Text mining and its application of natural language processing and machine learning technologies to problems of knowledge discovery, is one of the most challenging fields in bioinformatics.

This thesis describes and introduces novel methods for knowledge discovery and presents a software system that is able to extract information from biomedical literature, review interesting connections between various biomedical concepts and in so doing, generates new hypotheses.

The experimental results obtained by using methods described in this thesis, are compared to currently published results obtained by other methods and a number of case studies are described. This thesis shows how the technology presented can be integrated with the researchers' own knowledge, experimentation and observations for optimal progression of scientific research.

Chapter 1. Introduction

Advancement in biomedical research and the continuous growth of scientific literature presents a challenge for scientists to keep up to date with new information, theories and progress even in their own field of research. To help researchers cope with information overload, a significant effort has been made to organize and make this literature available in electronic form. The United States of America National Library of Medicine (NLM) has established itself as a primary source of bibliographic information from the biomedical field and it will be used as a source of bibliographical information in this study.

1.1. National Library of Medicine as biomedical information source

The history of the National Library of Medicine starts in a bookshelf of a Surgeon General's Office in 1818 when Joseph Lovell, the U.S. army surgeon general started purchasing medical journals and reference books (Blake, 1986) . A small hand written book in 1840 by Lovell's successor Thomas Lawson became the first library catalog with only 134 titles. Years later in 1880, a tireless work effort by the library director Dr. John Shaw Billings resulted in publishing sixteen volumes entitled "*Index-Catalogue of the Library of the Surgeon General's Office, United States Army*". During the years to come, the catalogue evolved from first printed, then photographic in the 1950s, to computerized "Medical Literature Analysis and Retrieval System" (MEDLARS) (Dee, 2007) system in the 1960s. The request had to be written on a special form and mailed to the library. For each abstract the entire set of magnetic tapes had to be searched sequentially which in 1964 took about 40 minutes, and a summary mailed back to the library member (Miles et al., 1982). With advancement of information technology in 1971 the library started providing online access and in 1993 through the NLM website (Hansen, 1998).

Today, via the National Center for Biotechnology Information (NCBI) the NLM houses and provides access not only to bibliography but also to a number of biomedical databases (Wheeler et al., 2008). The access to biomedical literature citations is provided by NCBI's PubMed (Edhlund, 2005) service which utilises the cross-database search and retrieval system Entrez (Baxevanis, 2008). The Entrez query system is also used for services including NLM catalog, nucleotide and protein database, genome sequences, and many others.

The main PubMed component is Medical Literature Analysis and Retrieval System Online (MEDLINE)(Miller et al., 2000). This is the NLM bibliographic database that contains citations and abstracts of journal articles in life sciences with a focus on biomedicine. In addition to MEDLINE resources, the PubMed

contains citations and links to full text articles from other biomedical related life science journals. In the year 2000 a new service, the PubMed Central (PMC) was established as a digital archive of full text journal articles (Roberts, 2001). Since then the amount of indexed publications (Figure 1.1.1) and related services increases dramatically every year making it necessary to use new, innovative text mining methods.

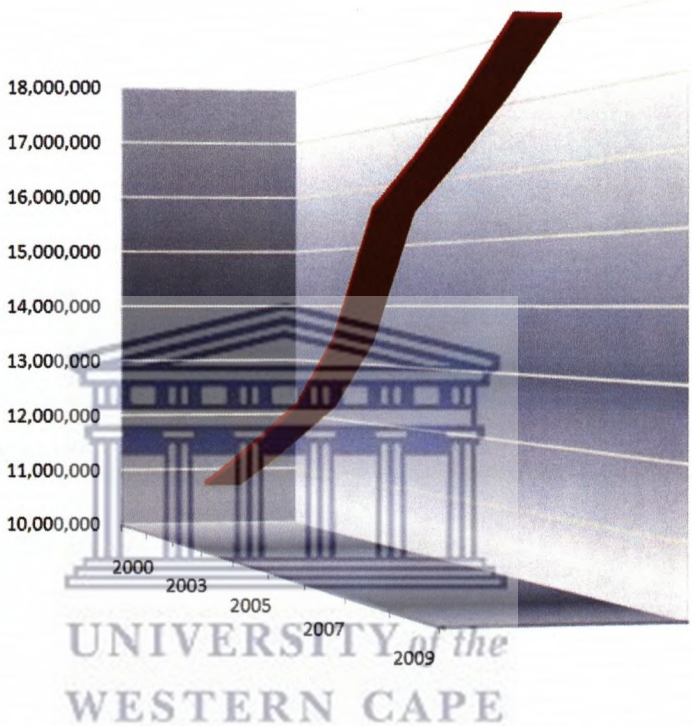


Figure 1.1.1 Number of citations in MEDLINE/PubMed since year 2000
(data source: NLM statistic http://www.nlm.nih.gov/bsd/licensee/baselinestats.html)

1.2. From automatic abstract creation to conceptual science

Decades before the information explosion, Vannevar Bush noticed that with any scientific investigation, "*the investigator is staggered by the findings and conclusions of thousands of other workers - conclusions which he cannot find time to grasp, much less to remember*". In his famous 1945 article "As We May Think" he envisioned the Memex machine that would help the researcher deal with the "*growing mountain of research*" (Bush, 1996). About a decade later, in 1958, for the first time an intellectual task in the field of literature evaluation was performed by a machine. In his paper "The Automatic Creation of Literature Abstracts" H.P. Luhn described a method of using computational statistics for the "*automatic creation of literature abstracts*" (Luhn, 1958). Swanson, a pioneer in bibliographic knowledge discovery in the 1980s, stated that even more important than the increase of literature volume is the combinatorial growth of unnoticed

logical connections (Swanson, 1990). In his work he described the idea of using biomedical literature as a potential source of new knowledge.

Today, automated literature analysis that aims towards conceptually driven knowledge discovery, coupled with vast amounts of information, creates a new paradigm: conceptual research i.e. a non-empirical literature based science that can accompany any research. Recently 'conceptual biology' has emerged as being complementary to empirical biology (Blagosklonny and Pardee, 2002, Bekhuis, 2006). P. Srinivasan defines it as "*text mining applied to the domain of biomedicine*" (Srinivasan, 2003).

1.3. Terminology

The following section describes and defines some of the terminology and concepts later used in this thesis. It briefly explains it from the angle related to this study without going into the details.

- **Biomedical text mining (TM)** can be defined as the use of automated methods for:
  - finding relevant documents;
  - finding hidden patterns, relationships and trends;
  - recognizing key concepts;
  - producing a summary of information found and
  - discovering new information.
- **Document**: a unit of information retrieval. In this thesis this will be a title of a biomedical publication followed by an abstract retrieved from the PubMed in PubMed XML tagged format. If document does not have an abstract (e.g. in case of older or some copyrighted publications), only title will be used to represent the publication.
- **Document collection**: also referred to as document corpus, corpora, or simply a collection, is a group of documents. In this study this will be a group of publication titles each followed by abstract (if such exists).
- **Abstract**: a brief summary of a publication. It describes the most important findings of the publication.
- **Biomedical entity**, or term defined in the biomedical domain, is something that has a distinct separate existence in an objective, material, or theoretical reality. It can stand alone, it has no dependences. An example would be the general use of the word 'gene' as well as a specific type of gene name such as 'abcd1', or the term 'blood pressure'.
- **Biomedical concept** is an idea or thought described in biomedical terms. If the biomedical concept is not verified, it is a hypothesis, theory or possibility. A biomedical concept whose truth has been proven is a fact.

1.4. Components of a text mining system

A text mining system can be divided into a number of components and related methodologies as shown in the figure below (Jensen et al., 2006). Information Retrieval (IR) and Entity Recognition (ER) are well established methodologies. Most research is currently focusing on the remaining three components: Information Extraction (IE), Information Integration and new knowledge discovery. However, the boundary between IR, ER and IE is not exact and all three tasks can be integrated in one application.

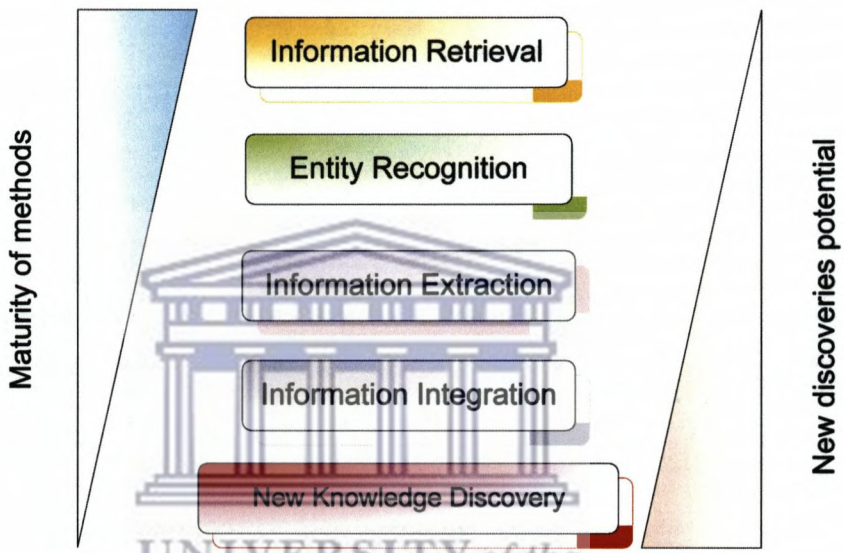


Figure 1.4.1 Biomedical text mining components. (adopted from (Jensen et al., 2006))

1.4.1. Information retrieval (IR)

Information retrieval (IR) is a process of identifying relevant documents from what is usually a large collection of documents, based on a search query commonly entered into the system as a list of keywords. The IR is expected to return a list of documents that will not only match the user's query but also his information needs. IR methodologies are usually based on Boolean and vector models (Jensen et al., 2006). The Boolean model uses keywords and Boolean operators between the keywords. All documents that satisfy the Boolean expression are retrieved. In the vector model (Lin et al., 2007) documents are represented as vectors of weighted terms. Weighting can be term frequency, inverse document frequency (logarithm of ratio between number of all documents over documents containing the term), or a more complex function. The query itself is transformed to a vector and some similarity metric determines how well this vector matches the document vectors (Salton and Buckley, 1987).

The best known biomedical IR system is PubMed. Every bibliographic record is linked to other NCBI databases as well as to each other within the Entrez system (Sayers et al., 2009). The bibliographic database is indexed by using controlled vocabulary called 'Medical Subject Headings' (MeSH) (Neveol et al., 2007). The MeSH thesaurus contains descriptors or 'main headings' representing biomedical concepts and subheadings referring to some aspect of the concept. Terms are arranged into a hierarchical tree structure, for example: Body Regions -> Face -> Nose. Terms can occur in more than one place in the hierarchy, for example Nose appears under Body Regions, Respiratory system and Sense organs[1]. Each article is assigned the most specific MeSH heading and subheadings that will cover the topic of the article. When processing a query, PubMed uses an Automatic Term Mapping (ATM) technique that applies the following translation tables sequentially: MeSH Translation Table, Journals Translation Table, Phrase List, and Author Index. If no match is found, the query phrase is broken down and the search process is repeated (Smith, 2004). Common, non-specific words, i.e. 'stop-words' are ignored both in indexing and in the search process (Appendix C).

There are attempts to find better, more precise methods for IR. Some researchers have suggested methodology that would restructure information returned from a PubMed search (Tanabe et al., 1999). Other researchers introduced an ontology based search over full text articles (Muller et al., 2004). Ontology is created by taking biological entities (e.g. gene, allele) and classes that describe the relationship between two entities (e.g. association). The ontology is then used to structure document collections in their entirety. Since the ontology defines the meaning of the terms, this system allows semantic queries. However, limitations of this system is that the user must test his keywords against the ontology before writing his query. (WormBase - (Rogers et al., 2008).

1.5. Entity recognition

Named Entity Recognition (NER or ER) or Automatic Term Recognition (ATR), is a process of recognizing and highlighting the biological entities found in retrieved documents (for example, genes and proteins). The function of ER is to identify, tag and map biological entities inside the text. As shown in Figure 1.5.1 this task involves three steps: term recognition, term classification, and term mapping (Krauthammer and Nenadic, 2004):

[11] Example is generated by using NCBI MeSH browser

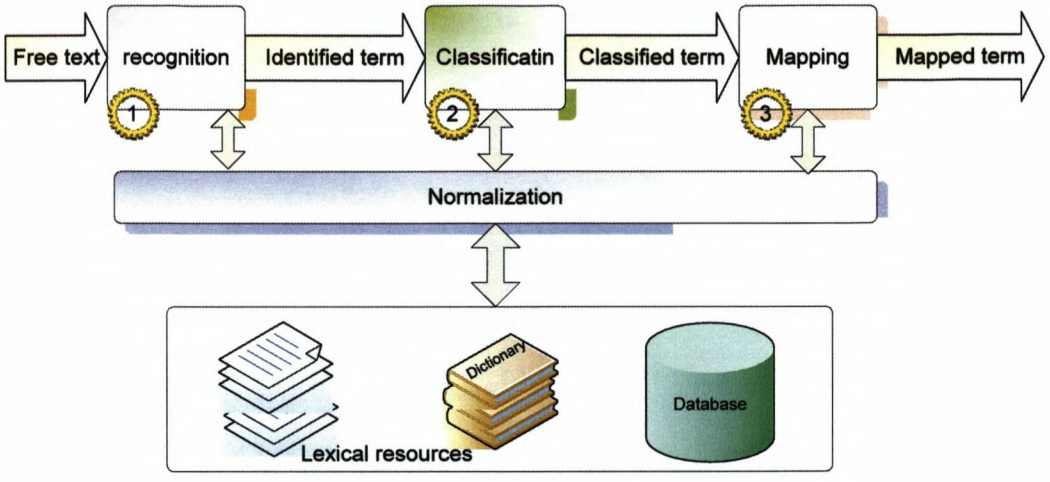


Figure 1.5.1 Three steps in term identification (adopted form (Krauthammer and Nenadic, 2004)).

Term recognitions identify the lexical units in text that correspond to biomedical entities and concepts. It uses binary classification to separate them into 'terms' and 'no terms'.

Term classification classifies terms into biomedical classes such as genes, proteins, chemicals and other categories.

Term mapping is a fine-grain classification that establishes the term 'knowledge space'. It assigns an identity to the term by linking it to well-defined concepts or reference data sources such as database (e.g. EntrezGeneID or UniProt protein identifier). In a more complex form, when integration is required, this task can be assigned to an Information Integration module.

Each of the tasks uses a normalization module on its own level, for example - to distinguish common words from biomedical entity names, to select the right term for the entity from the dictionary or to link the term to correct database resources.

However, irregularities and ambiguities associated with biological entity names make the identification task complex. Essentially the origin of the problem is a lack of naming conventions in the biomedical field. The most common ambiguities are caused by terms representing synonyms, homonyms, acronyms and abbreviations, or even a combination of those as summarized in Table 1.5.1.

	Synonym	Homonym	Initialism/Acronym	Abbreviation
Definition	different name for the same entity	same name for different entities	formation of a new word for an entity by combining initial letters of that entity (initialism) or the formation of a new word by using parts of a series of words and initials	contracted form of an entity
Example	ABCD1 gene is also known as: ABC42; ALD; ALDP; AMN	'CAT' can be a protein, animal or medical device.	AIDS is an initialism: Acquired Immune Deficiency Syndrome CPR is an acronym: cardiopulmonary resuscitation	sct is an abbreviated form of Secretin

Table 1.5.1 Synonym, homonym, acronym, abbreviation - definitions as used in this study.

In addition, a single term can be associated with different entities as well as other concepts. These ambiguities can be classified into three types:

- General cross ambiguity: common English words like 'bad' 'cat', or 'for' can stand for biomedical terms or abbreviations. For example, the term 'CAT' can stand for a protein, 'cat' as an animal or medical imaging (computerized axial tomography).
- Systematic or class ambiguity (Hatzivassiloglou et al., 2001): one name represents closely related entities. For example: gene products are referred with the same name as the gene; homologous genes can be represented by the same name (gene RARA in human, mouse, rat and chicken share the same name).
- Entity-specific ambiguity (Liu et al., 2006): multiple unrelated entities are represented with a same name. For example, the term 'CAP' refers to biological entities, such as capsid protein, cystine aminopeptidase, catabolite gene-activator protein, cyclase-associated protein, and calcium activated protease.

Entity name identification methods can be separated into four groups:

- Rule based methods (Fukuda et al., 1998) typically use orthographic or lexical rules, or morpho-syntactic features. Rule construction and their practical implementation can be difficult and results are applicable only to very specific areas.

- Dictionary or lexicon based methods use predefined terminological resources i.e. dictionaries, in order to locate term occurrences in text. If used alone this method can encounter a number of problems: the same term occurring in different dictionaries; a recognized term from one dictionary could be part of a longer term specified in another one; problems with spelling variations and the existence of compound terms etc. For these reasons the dictionary method is commonly combined with rule based methods (Gaizauskas et al., 2000).
- Statistical and machine learning methods are usually designed for a specific class of entities. Machine learning methods use training data to 'learn' features useful for term recognition. Widely used techniques are the Hidden Markov Model (HMM) (Zhou and Su, 2004, Rabiner, 1989), the Maximum Entropy Markov Model (MEMM) (McCallum et al., 2000), the Conditional Random Field (CRF) (Lafferty et al., 2001, McCallum and Li, 2003), Support Vector Machines (SVM) (Shi and Campagne, 2005) and others.
- Most of the methods used today are hybrid methods. An example of this approach would be to combine dictionary and rule based methods with a machine learning technique (Mika and Rost, 2004).

1.6. Information extraction

Information Extraction or (IE) is a process of extracting useful information from documents. Because of the complexity of the task at hand, most of the methodologies developed are limited to a specific field.

In the field of discovery and extraction of relations and associations between biomedical entities, much attention has been given to the extraction of protein-protein information from the early work of Blaschke (Blaschke et al., 1999) to the most recent of Chowdhary (Chowdhary et al., 2009) and He (He et al., 2009). Protein-gene and gene-gene (Kim et al., 2007) relationship extraction as well as some other types of interactions have been researched, but to a smaller extent. Usually the type of entity is well defined (e.g. genes, drugs, proteins) and the type of relationship can be either specific (e.g. regulatory relationship) or general (e.g. any biochemical association). A coherent 'all-in one' solution that will cover these and other biological and biochemical interactions still does not exist (Skusa et al., 2005). In general, approaches to this problem have been categorized into a number of groups (Zhou and He, 2008):

- **Co-occurrence and statistical methods** are based on a premise which assumes that if two biological entities have a related function it is more likely they will co-occur in the same abstract or a sentence. Statistical methods are based on detecting patterns from texts and use these for classification.

- **Rule based methods** use manually or automatically generated rules or patterns that could match potential textual relationships.
- **Natural language processing methods** use computational models of language to analyze syntactic structure and semantic meaning of the sentences.
- **Machine learning methods** use algorithms capable of learning from examples and applying the acquired experience/recognition model to extract knowledge from documents.
- **Hybrid methods**. Most of the text knowledge mining systems today combine two or more methods mentioned above.

1.6.1.1. Co-occurrence and statistical methods

In 1958 computers were used for the first time for literature evaluation. Luhn developed an algorithm for scoring sentences based on weight, derived from word frequency of occurrences. The highest scoring sentences were used to create 'auto abstracts' (Luhn, 1958).

Andrade and Valencia (1998) (Andrade and Valencia, 1998) conducted research based on word distribution statistics by comparing word frequency with background distributions in broad sets of protein families. They concluded that words with higher frequency can be good indicators of different protein functions and can be used as a guide for database annotations. Stapley and Benoit showed that if biomedical entities such as genes occur together in the same document with statistically significant frequency they are likely to have some functional relationship (Stapley and Benoit, 2000).

Jenssen and co-workers (Jenssen et al., 2001) created a gene-to-gene co-citation network for 13,712 named human genes. They validated extracted networks by three large-scale experiments showing that co-occurrence reflects biologically meaningful relationships.

Ding and colleagues (Ding et al., 2002) measured co-occurrences of biomedical entities in various text units (abstracts, sentences, and phrases) as indicators of interactions. Abstracts were found to be most effective when recall is of primary concern, phrases as most effective when precision is of overriding concern, and sentences as most effective over balanced precision/recall values.

1.6.1.2. Rule based methods

To overcome some of the shortcomings of co-occurrence methods, research has been done in developing rule (or pattern, template) based techniques. These methods use various techniques for extracting sentences that match manually or computationally specified templates.

Early work in this area used manually defined rules combined with restricted problem domains. Some of the limitations include pre-specified entity names and restricted sets of verbs that describe relations and interactions between biological entities. Blaschke and co-workers (Blaschke et al., 1999) describe a system in which a user can specify two protein names. Using a number of simple rules, PubMed text is parsed into fragments of a form: 'protein A – action – protein B' indicating that protein A interacts with protein B. The action verb specifies the type of interaction. They investigated 14 verbs, including their spelling variants. For example bind (-ing, –s, to, /bound).

Advancement in this methodology was made by introducing some statistical components into rule evaluation. In their work, Thomas et al. (Thomas et al., 2000) used 30 different words for interaction and developed several patterns around each verb. The patterns were used as filters on a tagged and parsed text. Results were ranked according to context, relation frequency and existence of entity names in the relation. They reported an average precision and recall of about 73% and 50% respectively.

Huang and colleagues (Huang et al., 2004) developed a method for automatic discovery of distinctive patterns that could point to the interaction between proteins. As an input the system used only a dictionary of protein names. Their method achieved precision of 80.5% and recall of 80%.

Pan and co-workers (Pan et al., 2006) developed the first system for transcription factors interaction extraction. The method uses a combination of automatic learning for the generation of rules and manual rule tuning. They obtained precision of 93% and recall of 88%.

1.6.1.3. Natural language processing methods

Natural language processing (NLP) is a well established discipline and now commonly used in text mining. NLP is characterized by syntactic and semantic analysis of sentences. Generally syntactic parsers are used to convert a variety of sentences to canonical form: subject, object and verb. Some ER methods, dictionary-based for example, can be used to semantically tag entities (e.g. protein names) and their relationships. Relationships are usually extracted by a rule that combines syntactical and semantic information (Jensen et al., 2006). For the syntactic analysis, a variety of parser technologies can be used. Each of them can produce different types of structures providing different information about the analyzed sentence.

- Part of speech tagging (POS) is a process of marking up the words that are part of speech like nouns, verbs, adjectives, adverbs, etc.

- Shallow parsing identifies syntactically related groups of words such as noun and verb phrases inside the sentence.
- Phrase structure parser breaks a sentence into a sequence of words and constructs a dependency tree consisting of dependency links between them.
- A dependency parser is similar to a phrase structure parser, but it does not make reference to a phrasal type and its span. Some authors make no differences between the two.
- Deep parsing is built on formal mathematical models and uses grammars that precisely describe a mechanism by which sentences and phrases are built from smaller blocks, like words or word elements.

Yakushiji and co-workers (Yakushiji et al., 2001) developed an experimental system for full parsing and tested it by trying to extract 133 structures from 97 sentences. 23% of the structures were obtained uniquely and 24% with ambiguity. A further 20% were extractable from not complete but partial results of full parsing.

Friedman and colleagues developed a general NLP based extraction system (Friedman et al., 1994) which they modified in an attempt to capture a chain of nested interactions. In 2001 they announced the GENIES (Friedman et al., 2001), a system that extracts and structures information about cellular pathways based on pre-defined knowledge model. The system had shown high precision of 96% and satisfactory recall of 63%.

Temkin and Gilder (Temkin and Gilder, 2003) showed that it is possible to reduce the complexity of NLP by focusing on domain specific structure instead of analyzing the language semantic. They developed a technique for extracting protein-gene-small molecule (PGSM) interactions using context-free grammar (CFG). This technique achieved a recall rate of 83.5% and a precision rate of 93.1% for recognizing PGSM names and a recall rate of 63.9% and a precision rate of 70.2% for extracting interactions between these entities.

Some researchers pointed out that the task of information extraction does not require full text understanding capabilities and complicated parsers and grammars. Park et al. (Park et al., 2001) proposed focusing on verbs of interest and start looking for their semantic arguments using combinatory categorical grammar (CCG). With this grammar, verbs are expected to be surrounded by a particular sentence structure. This approach tested on protein-protein relations achieved precision of 80% and recall of 48%.

Other studies suggested finding a balance between the complexity of NLP methods and rules based methods as linguistic analysis alone could not provide a satisfactory semantic interpretation. Experimental results confirmed that simple

part-of speech rules coupled with dictionaries could produce high precision and recall rates. Using this technique Ono et al. (Ono et al., 2001) reported average precision over 90% and recall over 80%.

Leroy and Chen (Leroy and Chen, 2002) suggested a somewhat different approach. They focused on prepositions in trying to capture the underlying sentence logic. They used prepositions 'by' and 'of' as the sentence entry point and retrieved phrases surrounding them according to the pre-defined template. This method achieved precision of 70%.

Huang and co-workers (Huang et al., 2004) suggested using a hybrid between shallow parsing and pattern matching. In the described method, they used a rule based shallow parser for coordinative structures analysis with both syntactic and semantic constraints. In addition, long sentences were segmented and relations extracted from the segments using pattern matching algorithms. In the full text analysis for protein-protein interactions they reported an average F-score of 80% on individual verbs, and 66% on all verbs.

Šaric et al. (Saric et al., 2006) focused their approach on the rule based, organism-specific relation extraction. At the same time they paid special attention to the semantic correctness. The system was tested on number of model organisms reaching accuracy of 83-90%.

Miyao and colleagues (Miyao et al., 2008) evaluated eight parsers based on dependency parsing, phrase structure parsing and deep parsing. The parser outputs were used as statistical features for a machine learning classifier for protein-protein interaction. They concluded that the level of accuracy obtained with different parsers are similar, but that accuracy improvements may vary when the parsers are retrained with domain-specific data.

1.6.1.4. Machine learning and hybrid methods

In the field of biomedical entity relationship discovery, it is difficult to make a clear cut distinction between machine learning and hybrid methods. Machine learning models are usually combined with other techniques such as co-occurrence, rule based or NLP. These methods produce a training set which the machine learning algorithm uses to construct a model. The model is then applied to predict desired patterns from a biomedical text.

Bunescu and Mooney (Bunescu et al., 2006) noticed a number of interesting properties of natural language statements. When a sentence describes the relationship between two entities it is usually by one of the three patterns:

(a) 'In fore-between pattern words' (FB) that stand before and between entities are used to express relationships. For example: 'activation of [p1] by [p2]'.

(b) 'In between pattern only words' (B) that stand between entities are important for expressing the relationship. For example: '[p1] activates [p2]'.

(c) 'In between-after pattern words' (BA) that stand between and after entities are used to express relationships. For example: '[p1] and [p2] interact'.

In addition, they observed that commonly only four words, apart from the entity names, are used to describe the relationship. For the classification they used the kernel based machine learning. Kernel is a similarity function for features derived from a pair of objects, in this case sentences reduced in length by using patterns they observed. Precision of their method is in the region of 70% and recall in the region of 40% depending of the evaluation environment.

Malik and co-workers (Malik et al., 2006) showed that combinations of different text mining algorithms increases system performance. They used three different protein tagging methods to compile a list of protein names found in the text. As an extra classifier, a machine learning meta-algorithm was used to check the outcome of the protein tagging. In searching for interaction they used a pattern matching method based on a set of regular expression. The essential part of the system was the data integrator, a module that combined data and produced output: an abstract with annotated protein names, protein-protein interactions and protein-mutation data. The method was tested using a number of challenge datasets. The results were dependant on data but the system achieved the highest precision/recall/F-measure in the region above 80%.

Van Landeghem et al. (Van Landeghem et al., 2008) used a linear support vector machines classifier for protein-protein interaction discovery. They deployed the Stanford dependency parser (Marneffe et al., 2006) to create a dependency tree. Their feature extraction method was based on syntactic and lexical patterns derived from the shortest path between two proteins in the dependency graph. To evaluate the method various benchmark data sets were used. The best performance achieved was 62% for precision, 52% for recall and 57% for F-measure.

Miwa and colleagues (Makoto et al., 2008) combined existing NLP and machine learning methods in a new way. They used dependency and deep parsers in combination with three kernels to produce input for the machine learning algorithm. Knowing that dependency parsers ignore some deep information, and at the same time that deep parsers do not find certain shallow relations, they used a combination of kernels to cover each parsers' loss of information. They achieved 62% F-measure using a Support Vector Machines as a classifier for protein-protein interaction.

Koussounadis and co-workers (Koussounadis et al., 2009) developed a method for protein classification based on the hypothesis that textual similarity between documents reflects similarity in biological function. They were processing abstracts using a machine learning approach (Support Vector Machines) to discriminate sentences containing information that are relevant to protein classification tasks. Feature vectors were constructed by using words representing protein names, description of their function and protein classification. Performance was measured by using Area Under Curve (AUC) method and for text classification it achieved 0.789 and for protein structural similarity 0.908.

Chowdhary et al. (Chowdhary et al., 2009) investigated methods of extracting triplets that consists of two protein names and an interaction word. They used dictionaries of protein names and interaction keywords to find sentences containing the triplets. The sentences were manually annotated to obtain true and false triplets/interactions needed for the machine learning algorithm. Finally, the Bayesian networks were used to learn the rules from this training set. The authors reported an overall accuracy of 87%.

In summary, the modern methodologies for biomedical entity relation extraction are hybrids of different techniques typically including combination of rule based, natural language processing and machine learning methods. Still, reliable extraction remains a difficult, unsolved problem. However, this is a dynamic, developing area of research with promising results.

1.7. Information integration

Information integration that combines literature with various biological data sets like gene and protein databases is a standard task performed by text mining systems. There are various ways this integration can be implemented - as a part of ER process, to support and complement empirical results, and as association networks, to name a few.

Andrade and Valencia (Andrade and Valencia, 1998) used cross-database referencing to retrieve sentences related to protein functions. They selected protein families by intersecting PDBSELECT (Hobohm and Sander, 1994) and HSSP (Schneider et al., 1997) databases on criteria of sequence similarity to the master sequence of the family. For each of the proteins they queried the SwissProt (Bairoch and Apweiler, 1996) protein sequence database to retrieve pointers to related MEDLINE abstracts. Finally, relevant sentences were selected by calculating word frequency in abstract in relation to word frequency in protein families.

Glenisson and coworkers (Glenisson et al., 2004) developed a framework called TXTGate that combines literature information with biological annotation

databases for the purpose of gene profiling. TXTGate summarizes groups of genes based on text indices. Visualization, clustering and links to external resources allow for in-depth analysis of the resulting gene profiles.

Another approach is to integrate empirical results with text mining methods. Kramer-Hammerle and co-workers (Kramer-Hammerle et al., 2005) combined microarray, promoters database and literature analysis to investigate the effect of long-term expression of HIV-1 Nef viral protein on astrocytes (cells that provide support and capillaries connections for neurons (Parri and Crunelli, 2003)). Two types of studies were performed: data driven and knowledge driven analysis. In data driven analysis the microarray-based experiment identified genes involved in astrocytes response to expression of the HIV-1 Nef protein. In parallel, the PubMed query 'HIV-1 Nef transcription factor' produced a list of genes to be used in knowledge investigation. Both streams were subject to further analysis (statistical, pathways, promoters) and finally combined to produce the resulting gene signaling and regulatory network.

Because of their ability to combine many types of data, networks are commonly used for information integration task. Dragon Plant Biology Explorer (DPBE) (Bajic et al., 2005) uses interactive networks to present associations between Arabidopsis thaliana (a model plant for genome analysis (Rensink and Buell, 2004)) genes and their functions. The associations presented were based on various information including genes ontology information, metabolic pathways, enzymes, metabolites databases such GO (Harris et al., 2004), AraCyc (Mueller et al., 2003), BRENDA (Pharkya et al., 2003), and user specified PubMed abstracts.

1.8. New knowledge discovery

New knowledge discovery is a process of discovering new, previously unknown information (Hearst, 1999). It combines information from various sources suggesting new links or generates hypotheses about new possible connections between biomedical entities.

In 1986 Swanson suggested searching for "*undiscovered public knowledge*" in complementary but disjoint literature. He used a model today known as the Swanson ABC model. According to this model, if some literature reports an association between variables A and B (denoted as AB) and different literature reports a connection BC, there is a possibility that A might be linked to C (denoted as AC) through the interconnection concept B. However, if those two literatures have never been cited together, and neither cites the other, scientists might not be aware of a potential A-C connection (Swanson, 1990).

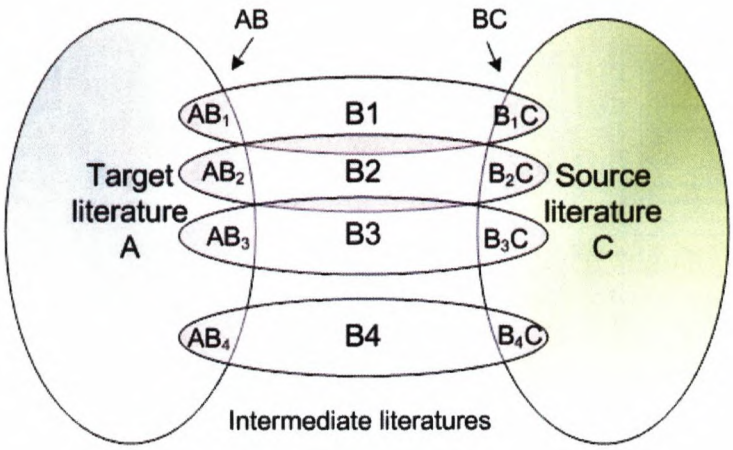


Figure 1.8.1 Venn diagram of the Swanson model adopted from (Swanson and Smalheiser, 1997). Sets of articles or "literatures" A and C have no articles in common, but they are linked through intermediate articles, Bi (i = 1, 2.. .). This structure may contain unoticed information that can be obtained by combining pairs of intersections ABi and BiC.

In 1996, Swanson and Smalheiser announced the Arrowsmith, an interactive system for finding hypotheses from complementary literature sets (Swanson and Smalheiser, 1997). The Arrowsmith uses two node search (A and C as input) that finds title terms (also B-term) that are shared across two literatures sets within PubMed. During more than two decades Swanson and associate researchers published a number of discoveries using this method. Starting from 1996's "Fish oil, Raynaud's syndrome, and undiscovered public knowledge" (Swanson, 1986) through a series of suggested associations summarized in ten years review (Swanson and Smalheiser, 1996) to "Running, esophageal acid reflux, and atrial fibrillation: A chain of events linked by evidence from separate medical literatures" in 2008 (Swanson, 2008).

Other authors have reviewed and extended this work. Chen (Chen, 1993) suggested a general model for creating knowledge by connecting related documents. Weeber and colleagues (Weeber et al., 2000) developed a Drug-Adverse Drug Reaction-Disease (DAD) system. Instead working with the 'raw' text, the system used words mapped to predefined concepts from the Unified Medical Language System (UMLS) Metathesaurus. The concept was tested on a well known link between Fish oil and Raynaud's syndrome. Stegmann and Grohmann (Stegmann and Grohmann, 2003) coined a new term: Swanson Linking (SL) defined as: "*finding disjoint literature partners by establishing meaningful links between them using information retrieval from bibliographic databases*". They used statistical analysis of word co-occurrences. Sorted pairs were clustered and graphically displayed in so-called strategic diagrams according

to their cluster centrality (abscissa) and density (ordinate). Promising terms linking complementary but disjoint literatures tend to appear in regions of low centrality and density.

Atkinson and Rivas (Atkinson and Rivas, 2008) suggested discovery of cause-effect hypotheses such as *"substance A producing effect B may be useful for treating disease C"* using the Bayesian net inference method. In their approach, the cause-effect relationships are extracted by using a previously trained algorithm. In the next step, the extracted concepts and relationships are processed by Bayesian inference methods to generate new patterns. Finally, those patterns are assessed by field experts. Although research in this area continues, most of the models are still based on Swanson and Arrowsmith concepts. (Torvik and Smalheiser, 2007, Smalheiser et al., 2009b).

1.9. Integrated web based frameworks

Since the World Wide Web existence, numerous web based text mining applications appeared and disappeared. This review will focus only on applications that are functional, related to the topics discussed and publicly available.

One of the earliest integrated text mining tools is the Arrowsmith that exists since 1996 (Swanson and Smalheiser, 1997, Smalheiser et al., 2009a). This is an application based on the Swanson's ABC methodology for finding possible meaningful links between two seemingly unrelated sets of articles in PubMed. Two versions are available: Swanson's and University of Illinois version that contains the 'Author-ity' tool for disambiguating authors on scientific papers, and the 'Anne O'Tate' tool for summarizing results of a PubMed query. Another tool for hidden relations discovery that exists since 2006 is Hristovski and Peterlin's Biomedical Discovery Support System (BITOLA) (Hristovski et al., 2006).

Chen and Sharp coauthored the Chilibot (Chen and Sharp, 2004), a tool for identifying relationships between biomedical terms. The system searches for sentences containing user supplied keywords and organizes them into different relationship types based on the linguistic analysis of the text. The relationships are summarized into networks and displayed as a graph. Based on this network the system has the ability to suggest new hypotheses.

An online prediction system for protein-protein interactions – PIE (Kim et al., 2008) allows manual paper uploads or downloads from PubMed. The PIE system uses natural language processing techniques and machine learning methodologies to find and display sentences that contains protein-protein interactions.

In summary, most of the research stops at the proof of concept and fails to deliver a working model of the theory presented. Very few integrated frameworks for biomedical text mining exist. They are usually linked for a specific project and short-lived.

1.10. Research questions

The primary objective of this research is to give a contribution towards developing new methodologies for knowledge discovery from the biomedical literature. The following research questions are the main focus of this study:

- *How to develop a method for an automatic knowledge extraction from biomedical literature by using a hybrid approach that combines advantages of the natural language processing, rule based and supervised machine learning techniques?*
- *Can the proposed methodology be effectively used for extracting information about a specific relationship between transcription factors and promoters of genes? In this case we are interested in whether a transcription factor does bind to the promoter of a specific gene.*
- *How to develop an integrated biomedical text mining software framework that combines named entity recognition, knowledge extraction and information integration?*
- *How to develop a method for generating potential new knowledge based on relational networks of biomedical entities extracted from disparate biomedical articles?*

1.11. Thesis outline

This thesis consists of seven chapters:

Chapter 1, *Introduction,* (as outlined till this point in the thesis) gives a brief introduction to biomedical text mining and its components, literature overview and highlighted the main research questions.

Chapter 2, *Background,* gives a brief summary of theoretical foundation this study lies on.

Chapter 3, *Automated extraction of information: sentence based approach*, presents the concept based knowledge discovery (CobKD), a proposed knowledge extraction methodology.

Chapter 4: *Extracting information about transcription factor binding to gene's promoter*, presents an application for the proposed methodology and critically discusses results.

Chapter 5, *Implementation methodology*, describes the integrated biomedical text mining system architecture: Dragon Exploration System. It also presents a methodology used for software development: the Academic Biomedical Extreme Programming (ABXP).

Chapter 6, *Integrated text mining framework: Case studies*, presents a number of research case studies that arise from this study.

Chapter 7, *Conclusions*, summarizes the main findings, contributions and limitations of this study. Discuss on future research is followed.

Chapter 2. Background

The chapter gives a brief summary of theoretical foundation this study is based on. The related theories include selected topics in string matching, supervised machine learning and methods for evaluating classification algorithms and biomedical text mining systems as a whole.

2.1. Levenshtein edit distance

Commonly used algorithms in text mining, for example to solve problems in spelling variants, are those for approximate string matching. They are based on an 'edit distance' metric that measures similarity between two strings. In this research one of those algorithms, the 'Levenshtein edit distance' (Levenshtein, 1966) will be used to measure similarity between sentences. Generally, the edit distance is a metric defined as a minimum number of operations, called costs, needed to transform one string into the other. In other words, the distance $d(x,y)$ between two strings x and y is the minimal amount of operations needed to transform x into$y$, where an operation is an insertion, deletion, or substitution of a single character.

To calculate the distance $d(x,y)$ between the string x and the string y,a matrix $L_{0..|x|,0..|y|}$ is used. The dimensions of this matrix are a $(n+1)*(m+1)$ where n is the length of string x and m the length of string y. Each cell $L_{i,j}$keeps the minimum number of operations needed to match $x_{1..i}$ to $y_{1..j}$ and its value can be calculated as a simple function of the surrounding cells (Navarro, 2001):

$$L_{i,0} = i,$$
$$L_{0,j} = j,$$
$$L_{i,j} = if\ (x_i = y_j) then\ L_{i-1,j-1}$$
$$else\ 1 + \min(L_{i-1,j}, L_{i,j-1}, L_{i-1,j-1})$$

At the end of the calculations: $d(x,y) = L_{|x|,|y|}$.

Insertion and deletion are assigned the cost of 1, substitution the cost of 0 if the next characters are equal, else 1. To remove discrimination of the longer terms the distance is normalized by dividing it with the term length (Yang et al., 2008):

$$Cost_{norm}(x,y) = \frac{d(x,y)}{lenght(x)}$$

For example the thyroid gland swelling can be spelled as *goiter* or *goitre*. If the term *goitre* is not recognized as a disease name the algorithm finds the closest match and calculates the distance between the known term and the term found in the text. If the cost is below some predefined threshold, the term in text is marked as a disease concept (Table 2.1.1).

		G	O	I	T	R	E
	7	6	5	4	3	2	2	2
R	6	5	4	3	2	1	2	2
E	5	4	3	2	1	1	1	2
T	4	3	2	1	0	1	2	3
I	3	2	1	0	1	2	3	4
O	2	1	0	1	2	3	4	5
G	1	0	1	2	3	4	5	6
	0	1	2	3	4	5	6	7

Yellow lines show the editing steps. The number in a cell shows the actual costs to get to the specific field. At the end, a red line shows the optimal path and top right cell the final cost 2.

Table 2.1.1 Calculating edit distance between words goitre and goiter

2.2. Supervised machine learning

Supervised machine learning (ML) is a technology related to the problems of classification or regression. Both classes of problem use learning of an unknown function from training instances (examples) as shown in Figure 2.2.1.

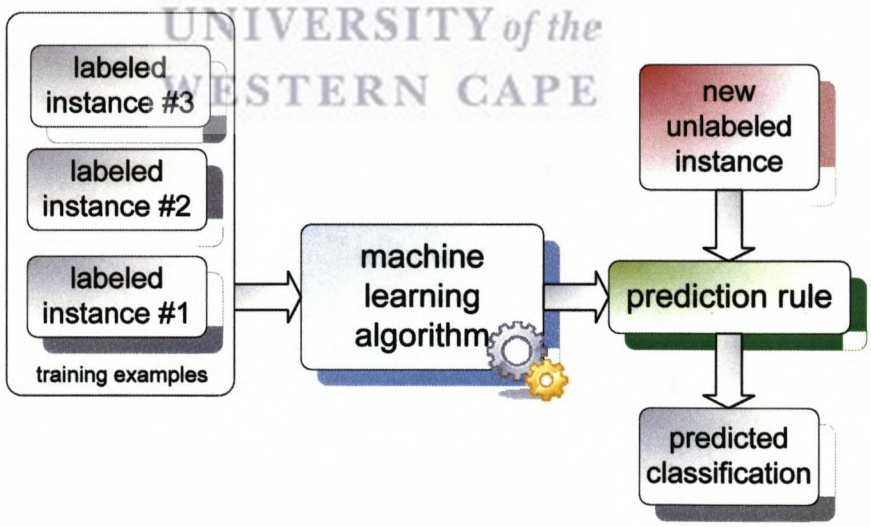


Figure 2.2.1Machine learning model.

Each instance is represented with a label and a number of attributes called a feature vector. The label is a class the instance belongs to. Some target function

maps the feature vector to one or more output classes. When supplied with a new unlabeled instance, the function learned is used to predict this instance class.

A labeled instance can be described as assignment of values $f = (f_1, \ldots f_n)$ to a set of features $F = (F_1, \ldots F_n)$ and one of m possible classes $c_1, \ldots c_m$ to the class label C. For binary classification m =2 when a class can have one of two values, usually (0, 1) or (yes, no) (Yu and Liu, 2004).

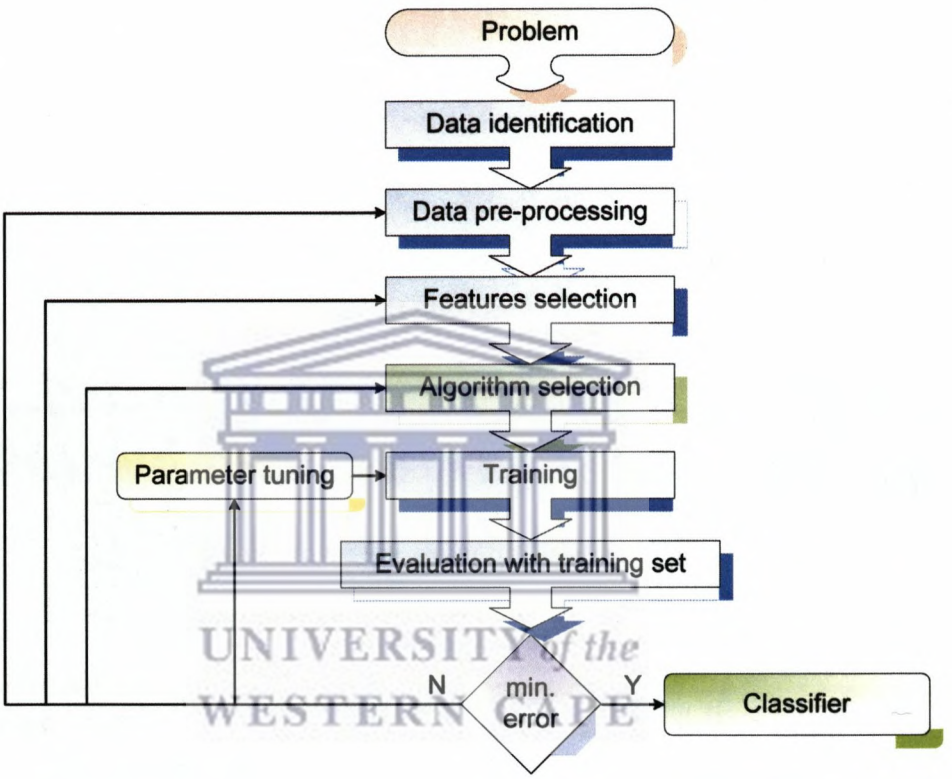


Figure 2.2.2 Supervised machine learning flowchart (adopted from (Kotsiantis, 2007)).

A typical process of selecting the algorithm that would perform the classification task is shown in Figure 2.2.2. (Kotsiantis, 2007). The first step is a data identification process. In the second step, pre-processing, this data is narrowed down. In this step the irrelevant and redundant features are removed from instances that will be used as training data. The learning and performance of the classifier is essentially influenced by the features selection, so it is important to have a methodology for their evaluation. Methods of evaluation mainly adopt two approaches: individual feature evaluation and feature subset evaluation. In the individual feature evaluation, features are ranked according to their importance in differentiating instances of different classes. This method removes irrelevant features, but redundant features might be unnoticed as they are likely to have

similar ranking. Methods of subset evaluation, search for a minimum subset of features that satisfies some goodness measure. It can remove irrelevant features as well as redundant ones. In general this is an optimization problem of competing goals: (1) the goodness (to be maximized), and (2) the number of features (to be minimized). (Guyon and Elisseeff, 2003).

The machine learning algorithm selection is another critical step. There is no universal classifier that can give best results on all given problems and all data sets. The classifier performance depends on characteristics of the data and it might be selected to best suit the problem. For a given problem more than one classifier should be tested and their accuracy compared.

The simplest method to test the accuracy is to compute the error on the sample data itself (Larranaga et al., 2006). However, this method usually predicts errors too optimistically. In k-fold cross-validation the training set is randomly partitioned into k folds that are usually mutually independent. Each of the folds is left out and used as a testing set. The error rate of the classifier is the average of error rates of the folds. A special case of the k-fold method is the 'leave-one-out' where all test subsets consists of a single instance.

Finding the classifier most suited to the particular classification problem is a dynamic process that include finding the best combination of a data set, feature vectors and ML algorithm that produces the minimal error rate. The following section presents a brief overview of classifiers used in this study.

2.3. Support Vector Machines

Cortes and Vapnik (Cortes and Vapnik, 1995) introduced the support-vector machines (SVM) as a method for two-groups classification problem. The SVM view input data as two sets of vectors, maps them into some high dimensional feature space and define a decision surface a hyperplane, that optimally separates them into two categories. The vectors that define the margin of largest separation between the two classes are called support-vectors. Joachims (Joachims, 1998) provided theoretical and experimental evidence of suitability of SVM for use for in text mining applications.

For some text mining applications, particularly when the feature space is relatively small, it is difficult to find the hyperplane that separates objects of two classes with sufficient accuracy. To overcome this problem some additional features can be derived from the original ones by using so called kernel functions. By doing so feature vectors are moved into higher dimensional space where separation might be easier.

2.4. Decision Tree

The decision tree algorithm (Mitchell, 1997) constructs tree data structure consisting of decision nodes, branches and leaves. A node represents a feature, a branch represents a value of that feature and a leaf represents a class.

The decision tree starts as a single node representing the whole training set. Algorithm identifies the attribute that provides maximum discrimination among instances. Such an attribute is said to have the maximum information gain. A child node is created for each value of that attribute and all instances with the same attribute are attached to it. If all instances attached can be uniquely classified the child node is classified and marked as a leaf node. The process of creating child nodes continues recursively until all attributes are used. If there are unclassified instances left, they are all assigned the same class as the majority of the instances under that branch.

The best known decision trees algorithms are J.R. Quinlan C-series algorithms: C4.5, C5.0 (Quinlan, 1993) and their various adaptations.

2.5. Random Forest

In his 2001 paper, L. Breiman described a classifier he named the random forest (Breiman, 2001). The random forest is a combination of decision tree classifiers in such a way that "*each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest*".

2.6. Naïve Bayes

Naive Bayesian classifiers are probabilistic classifiers based on Bayes theorem (Irina, 2001). The classifiers assume that features are independent for a given class of example and assigns the most likely class to instances. This probability is calculated by using the Bayes formula. Despite this assumption of independence the classifier is very successful and often used in text mining (Sohn et al., 2008, Chowdhary et al., 2009).

2.7. K*

K* or Kstar is an instance based classifier developed by J.G. Cleary and L.E. Trigg (Cleary and Trigg, 1995). The classifier learns from previously classified instances using the assumption that similar instances have similar classification. The first component of the classifier is a distance function that determines the similarity of instances using entropy as a measure. The second one is the classification function which specifies how this similarity determines the classification of an unclassified instance. This algorithm is capable of handling symbolic attributes, real values as well as missing attributes.

2.8. Neural Networks

Although there are many types of neural networks the term itself frequently refers to Multilayer Perception Network (MLP) (Raju et al., 2002). The network includes three or more layers each consisting of certain number of nodes. Each node in one layer connects with the certain weight to every other node in the subsequent layer. The goal of the training process is to adjust the weights so the output from the network will match the desired values. In the learning phase different learning algorithms for changing weights are proposed and the back-propagation is one of the best known. In back-propagation, the network propagate weights from the output towards input, and during the classification process only propagation occurs (Rumelhart et al., 1986). Although the work in application of neural networks to text mining is smaller than for other techniques they have been successfully applied in the text classification tasks (Govindarajan and Chandrasekaran, 2007).

2.9. Biomedical text mining systems evaluation measures

Most commonly used metrics to evaluate quality of the IR, ER or classification system are the measures of: precision (P) or positive predictive value (PPV), recall (R) or Sensitivity (Se), Specificity (Sp), F-measure (F) and Receiver Operating Characteristic (ROC) curves (Hersh, 2005).

One of the basic tasks that text mining systems and classifiers perform is one of binary classification in which an entity can belong to one of two classes. In other words, the entity can be either correctly or incorrectly recognized or classified. True positive(tp), true negative(tn), false positive (fp) and false negative (fn) are all possible outcomes of a single instance, classification or recognition. For classification of case, sample, instance, or recognition of en entity the following definitions applies:

- tp case was positive and classified positive;
- fp case was negative but classified as positive.
- tn case was negative and classified as negative.
- fn case was positive but classified negative.

An outcome of classification of a set of instances can be summarized in a contingency table (Table 2.9.1)

		actual classes	
		positives	negatives
classification	positives	true positives (tp)	false positives (fp)
	negatives	false negatives (fn)	true negatives (tn)

Table 2.9.1 Contingency table (also called confusion matrix).

The green diagonal elements represent correctly classified entities while the red diagonal elements represent misclassified entities. Table 2.9.2 shows definition of more complex measures derived from these basic measures.

Measure	Formula	Meaning
Precision (P) or positive predictive value (PPV),	$P = \frac{tp}{tp + fp}$	exactness
Recall or Sensitivity (R or Se)	$R = \frac{tp}{tp + fn}$	true positive rate
Specificity (Sp)	$Sp = \frac{tn}{tn + fp}$	true negative rate
Accuracy (A)	$A = \frac{tp + tn}{tp + fp + fn + tn}$	proportion of true results
F-measure (F)	$F = \frac{2PR}{P + R}$	overall correctness

Table 2.9.2 Measuring classifiers accuracy.

The F-measure is the weighted harmonic mean of precision and recall. It is based on van Rijsbergen's effectiveness measure (Van Rijsbergen, 1979):

$$F = \frac{1}{\frac{\alpha}{P} + \frac{1-\alpha}{R}} = \frac{(\beta^2 + 1)PR}{\beta^2 P + R} \quad \text{where } \beta^2 = \frac{1-\alpha}{\alpha}; \quad \alpha\epsilon[0,1]$$

Values $\beta < 1$ emphasize precision while $\beta > 1$ emphasize recall. Commonly used values for β are: 0.5 - weights precision twice as much as recall; 1 - balances precision and the recall; 2- weights recall twice as much as precision.

Some related measures are:

$$False\ positive\ rate = \frac{fp}{fp + tn} = 1 - Sp,$$

$$False\ negative\ rate = \frac{fn}{tp + fn} = 1 - Se$$

Another way to assess the classifier performance is by drawing the Receiver Operating Characteristic (ROC) graph (Spackman, 1989) and calculating the area below that graph. ROC curve is created by plotting the true positive rate against false positive rates or sensitivity versus 1-specificity as shown in Figure 2.9.1. It shows the tradeoff between sensitivity and specificity - any increase in sensitivity goes together with a decrease in specificity (Davis and Goadrich, 2006).

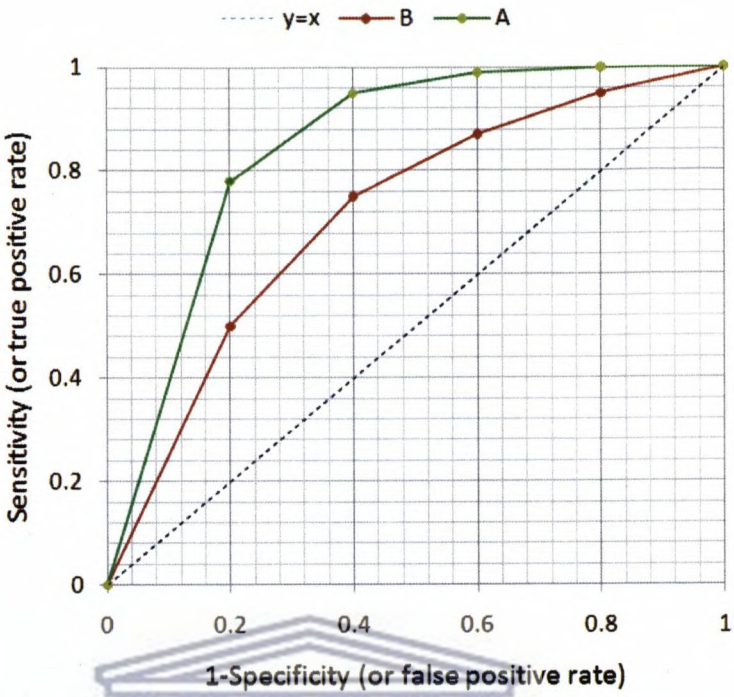


Figure 2.9.1. ROC diagram. Red and green lines are examples of ROC curve.

A dotted diagonal line divides ROC space into two areas. The line itself (y=x) represent the case when the class is guessed randomly. The space above the line indicates good classification results. Point (0, 1) is the perfect classification point representing 100% sensitivity and 100% specificity because true positive rate is 1 (all) with no false positives (0). In contrast, in point (1, 0) the classifier is incorrect for all classifications. It is interesting to note that in this case if each of the predicted class is substituted with the opposite class an ideal classifier that predicts every instance correctly will be created. A good classifier would follow the upper left axis. The area under the ROC Curve (AUC) is a useful metric for the classifier performance. The AUC comparison can establish a dominance relationship between classifiers. Curve A dominates curve B if it is always to the left as wells as above curve B. If the ROC curves intersect, the total AUC is an average comparison between classifiers.

2.10.Chapter summary

Chapter 2 gave a summary of the main concepts used in this study: approximate string matching by using Levenshtein edit distance, a brief introduction to supervised machine learning systems with a number of ML algorithms and biomedical text mining systems evaluation measures.

Chapter 3. Automated extraction of information: sentence based approach

This study proposes a new and original generalized approach to automated extraction of information. This approach named "Concept based Knowledge Discovery" (CobKD) methodology makes use of NLP, rule based and other approaches in a supervised machine learning framework. It also uses a specific data pre-procesing in the process of model building. It is based on the analysis of sentences rather than abstracts and keywords to deliver the high quality information about various biomedical concepts relationships.

This chapter presents some of the possibilities of extracting targeted information based on the presumed (core) structure of the sentence in which the information is harbored. It will be shown how the sentence structure can be specified, extracted, stored in a small local database and further annotated by curators. Separately, the study explores what the quality of such extracted information is and how it can be additionally improved.

3.1. Data pre-processing

For the experimentation purposes the model of supervised ML described in Chapter 2 was implemented. In the classifier training process, the annotated biomedical text is broken down into sentences by a sentences parser (Figure 3.1.1) A user describes the concept using simple, intuitive queries and submits them into the system. In the sentences filtering step, the queries are used as templates to extract matching sentences. These sentences are then evaluated and labeled by curators and used for the derivation of features. A flexible approach is taken to select the features, that will in combination with the proper classifier, produce minimal classification errors. When this is achieved a classification model is created and recorded for future use.

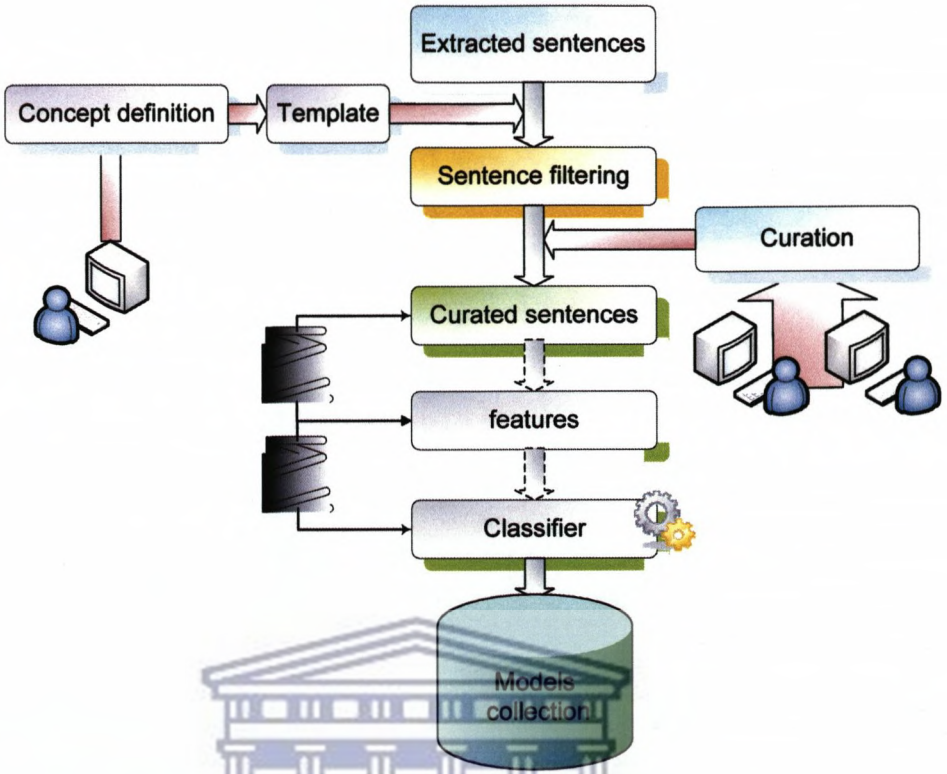


Figure 3.1.1 Classifier training.

When defining the biomedical concept, the most studied example in relationship extraction is protein-protein interaction. For example, a way to express interaction of two proteins would be:

protein A interacts with protein B

There could be variations of this expression, such as

(protein A) (some word that implicates an interaction) (protein B)

More generally one can write:

[protein] X [interaction] Y [protein]

where *[protein]* stands for any name or identifier that identifies a protein, and *[interaction]* stands for any term that implicates interaction/mutual binding of proteins. *X* and *Y* stand for any set of words that could appear in between. If a structure of such a type could be found in a sentence, then there is a high probability that the sentence could convey information about mutual binding of two proteins.

This consideration puts forward a possibility of automated recognition of the presence of information about mutual protein binding and a possibility that the

actual extraction of that information can be automated using for example, machine learning systems. However, problems to be solved are: (a) how to define in a convenient manner the desired sentence structure, and (b) how to extract such sentences from a set of documents.

A system that allows extraction of a variety of sentence structures has been developed herewith. The process starts with a sentence structure or concept description. The concept serves two purposes:

- to act as a search template for sentences filtering;
- coupled with the classifier defined in the machine learning process it is used for new knowledge discovery.

Concept description consists of references to biomedical entities and reference to relationship between entities and keywords. Elements are linked by implicit 'AND' and/or explicit 'OR' (written as '|') clauses (Figure 3.1.2).

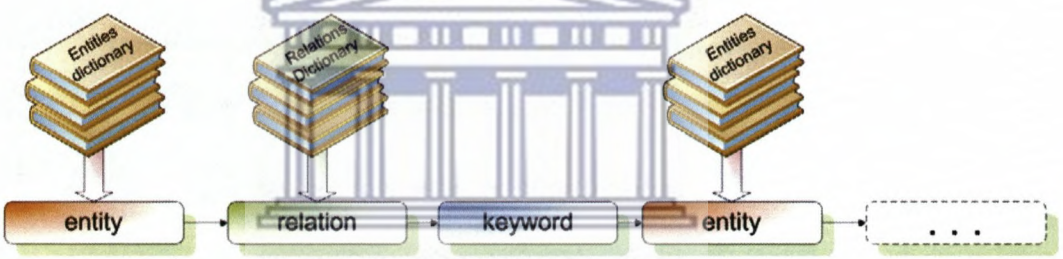


Figure 3.1.2 Concept description

A biomedical entity is presented as a keyword that stands for any entity from the selected dictionary of entities (Appendix A). For example, it can be a name for gene, protein, chemical, toxin… A reference for human genes and proteins stands for 289,240 actual entity names stored in a curated dictionary of human genes and proteins.

Similarly, a reference to a relation is presented as a keyword that stands for any word from the selected dictionary of relations. The relations are divided into a number of groups: direct interactions, indirect effects, modifications, 'other' types of interactions, as listed in Appendix B.

References can be mixed freely with English words. For example, the following query:

[gene or protein] [interaction] [gene or protein] promoter

describes a concept that that contains the name of a gene or a protein that interacts with another gene through the promoter of that other gene. An affix stemmer is used to handle various prefixes and suffixes (es, s, ing…).

Multiple lines, all forming the same query, can be used to describe more complex information needs. For example:

[gene or protein] interacts [gene or protein] promoter
[protein] inhibition [toxin] silence [gene]
[gene or protein] activates|induces|represses|reduces|increases [gene] transcription
. . .

Figure 3.1.3 shows an implemented dialog box that allows a user to describe the concept (Sentence model entry) and provide some additional information like curator names that will be doing classification, database name, database description, filter sensitivity and maximum distance between keywords. The last two parameters are used to adjust the NLP filtering algorithm sensitivity to accommodate misplaced words (different authors may have different writing styles) and to maximum distance between two keywords.

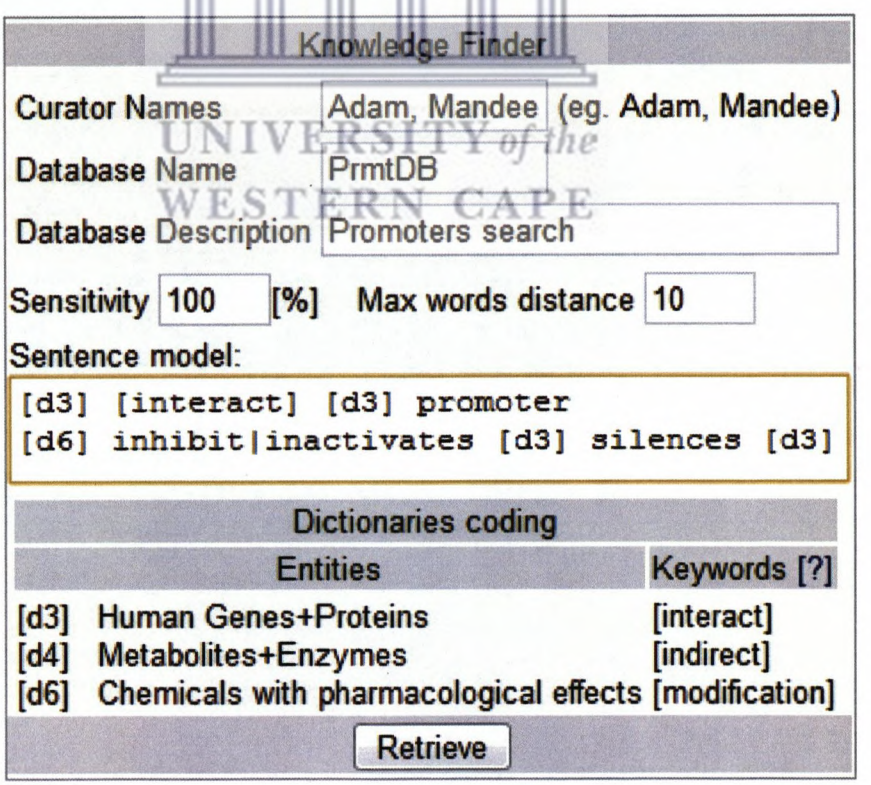


Figure 3.1.3. Sentence filtering/Knowledge finder dialog.

After the user's submission, the sentence database is created by using the currently opened document collection database. Figure 3.1.4 shows the conceptual diagram of the database:

- Table SentenceDB contains information copied from the user's form.
- Table Curator contains curator internal IDs and names.
- The Sentence table contains sentences extracted from the document collection, that match user defined concepts. This table is automatically filled in by the parser after the user's form submission. The system automatically assigns selected curator IDs to all sentences and annotates sentence with 'no information' tag.
- The SentenceEntity table provides additional information about the sentence: number of tagged entities and dictionary IDs they belong to.
- The Abstract table belongs to the main database and it is linked to the sentence by the PMID allowing the user to see the sentence in the original or tagged PubMed abstract.

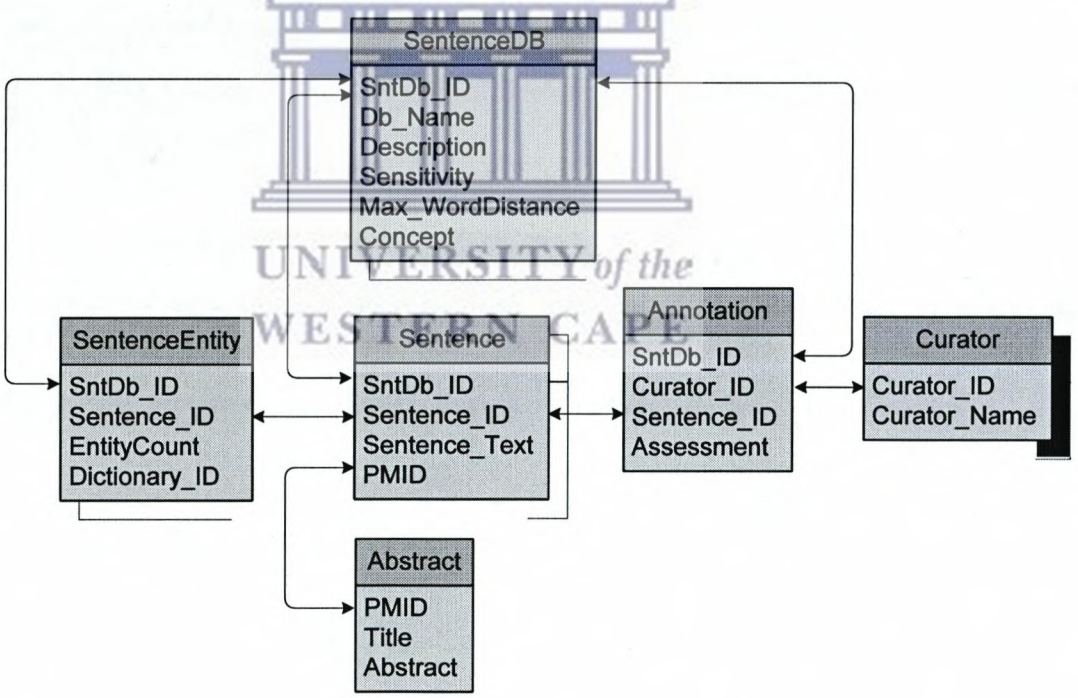


Figure 3.1.4 Sentence database.

3.2. Training set labeling

After the system has automatically created a list of sentences that match the concept description, one or more curators evaluate the sentences information content. In the case that a broader context was required, the curator has a choice to

retrieve the complete abstract containing this sentence in annotated or original form, by clicking a single button on the program interface.

Sentences can be labeled as: no information, inconclusive, low, low to medium, medium, medium to high, and high information (Figure 3.2.1). Internally, the labels were seen as a numbers from 0 to 6 and linked to curator names for editing purposes.

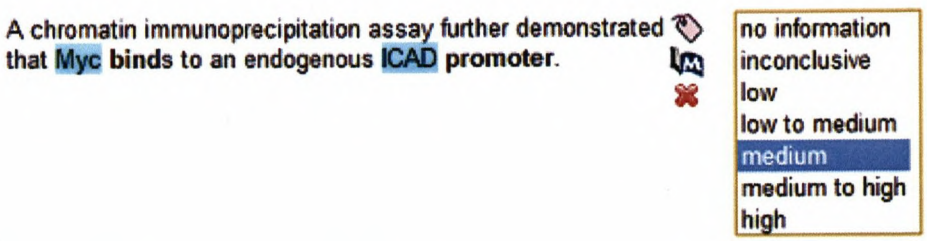


Figure 3.2.1 Training set sentences labeling.

Although sentences can be classified in seven categories, the classification algorithm was given the threshold value. If the sentence class was below this value the sentence was classified as incorrect. At the end, all sentences are classified as correct (marked as 1) or incorrect (marked as 0).

It was noticed that curators not always agree on the information value of the sentence. For example, in a test conducted, 100 sentences were marked by three curators assigning a mark from 0 to 6 to each of them. Figure 3.2.2 shows distribution of the marks.

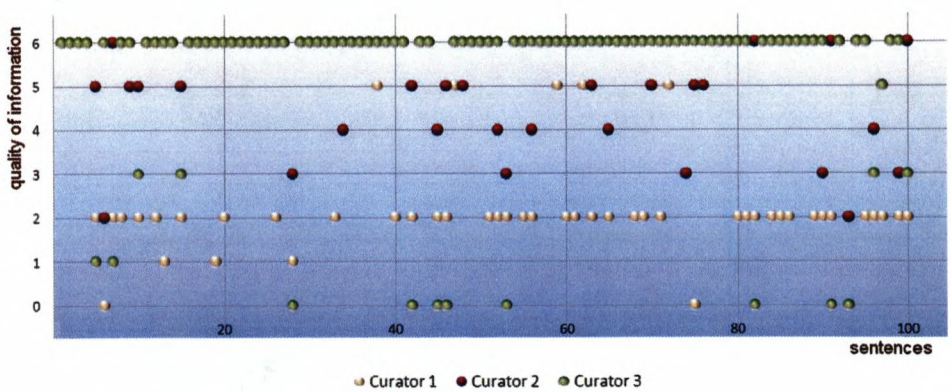


Figure 3.2.2 Information quality assesment by three curators.

If curators all agreed in their assessment, dots of different colors will be merged to one position for each sentence. However, they are not only scattered throughout

the graph but in certain cases express opposite opinions. This result shows that it is important having double curation but at the same time having an established base line on which the knowledge is assessed. In this particular experiment, the discrepancies between the curators appeared mainly because some curators were not sure how to assign intermediate values.

3.3. Features selection

The purpose of conducting feature selection is to find the most relevant features and to remove redundant ones from the feature set. In the system presented, the following sentence attributes were converted to features:

- Keyword distances.
- Existence of negation words.
- Levenshtein edit distance.
- Words frequency.

The lexical analysis can be performed with and without using the stop-words. Stop-words are words that most frequently appear in text, for example: 'a, 'an', 'are', 'at' etc. The stop-words list used in this research is the list of 133 words recommended by the U.S. National Library of Medicine[2] .

3.4. Keyword distances

Keyword distances are distances measured in a number of words between the keywords in the concept description. The measurement starts from the beginning of the sentence to the first keyword, continues with distances between the keywords and ends with the distance between the last keyword and the sentence end. For example, distances for the following sentence are 10213:

*A **NF- B binds** to the **OX40** basal **promoter** region in vivo.*
1 0 2 1 3

To ensure that the feature vectors are always the same size, sentences must be normalized to the concept description format. In other words, if the sentence contains multiple gene and protein names and keywords from the concept, the parser tries to locate the part of the sentence that matches the concept description most closely.

3.5. Negation words

A parser uses a list of negation words (e.g. not, fail, abolish, suffix n't, etc.) to determine the sentence 'sentiment' toward positive or negative statement. The

[2] Appendix C

feature vector value is assigned to a number of negations found inside the sentence.

3.6. The Levenshtein edit distance

The Levenshtein edit distance is used as a measure of similarity between a sentence normalized to a concept description and the concept itself.

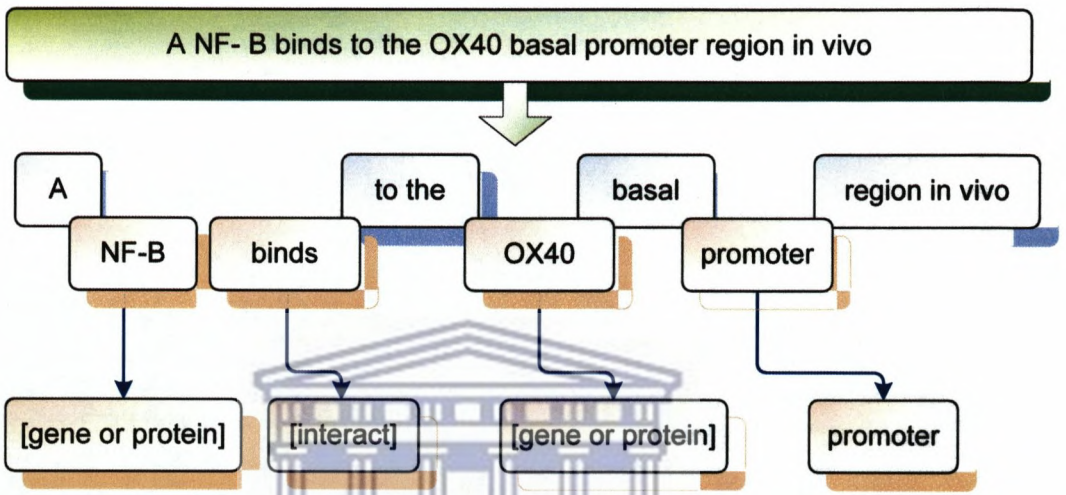


Figure 3.6.1 Calculating Levenshtein distance.

As Figure 3.6.1 shows, the part of the sentence that matches the concept description is extracted and the edit distance between this part and the template calculated. In an actual implementation, stop-words (a, to, the, in) are removed. Lower measure indicates the closer similarity between the sentence and the concept.

3.7. Words frequency

Words frequency feature vector values consist of flags '1' indicating presence or '0', indicating absence of the most common words. Common words are problem specific and they are calculated for the concept related training set. Words with the frequency of 5% and more are considered to be 'top-words'. Top-words are defined as a group of words that constitute 5% or more words in positive labeled sentences and a group of 5% or more in negative labeled sentences. In this calculation positive and negative sentences are treated as a 'bag of words'.

3.8. Feature vectors construction

Features and labels are used to make a collection of n instances. Each of them is described with vector F_i representing list of features and a variable $c \in$ (Consortium, 2009) representing a class:

$$C = (c_1, \dots, c_n)$$

$$F = (F_1, \dots, F_n)$$

$$F_i = (f_1, \dots, f_m)$$

Individual features can be named as:

$$F_i = (d_{i1}, \dots, d_{ij}, s_i L_i, t_{i1}, \dots, t_{ik})$$

So the collection of instances can be written as:

$$I = \begin{Vmatrix} d_{11}, \dots, d_{1j}, s_1, L_1, t_{11}, \dots, t_{1k} & c_1 \\ \vdots & \vdots \\ d_{i1}, \dots, d_{ij}, s_i, L_i, t_{i1}, \dots, t_{ik} & c_i \\ \vdots & \vdots \\ d_{n1}, \dots, d_{nj}, s_n, L_n, t_{n1}, \dots, t_{nk} & c_n \end{Vmatrix}$$

where:

n is the number of instances in the training set;
C, F is a class and feature vector space of the training set;
c_i, F_i is a class and feature vector of an instance;
$d_1, \dots, d_j$ represents a word distance between $j - 1$ keywords;
s_i is the sentence sentiment measure (negative words count);
L_i is a Levenshtein distance between concept description and the sentence normalized to the concept;
$t_1, \dots, t_k$ and $t_x \in \{0,1\}$ represents absence or presence of any of the k top-words.

3.9. Knowledge extraction

In the knowledge extraction process (Figure 3.9.1.) the user selects a predefined concept. Internally this concept is linked to the appropriate template, feature creation module and classifier model. The template is used to filter the sentences. The sentences themselves are used to form a base for unlabeled instances data set creation. This data set is then submitted to a classifier that applies the selected model and labels the instances. Pointers

from the positive labeled instances are used to retrieve original sentences that are finally presented to the user as an extracted knowledge.

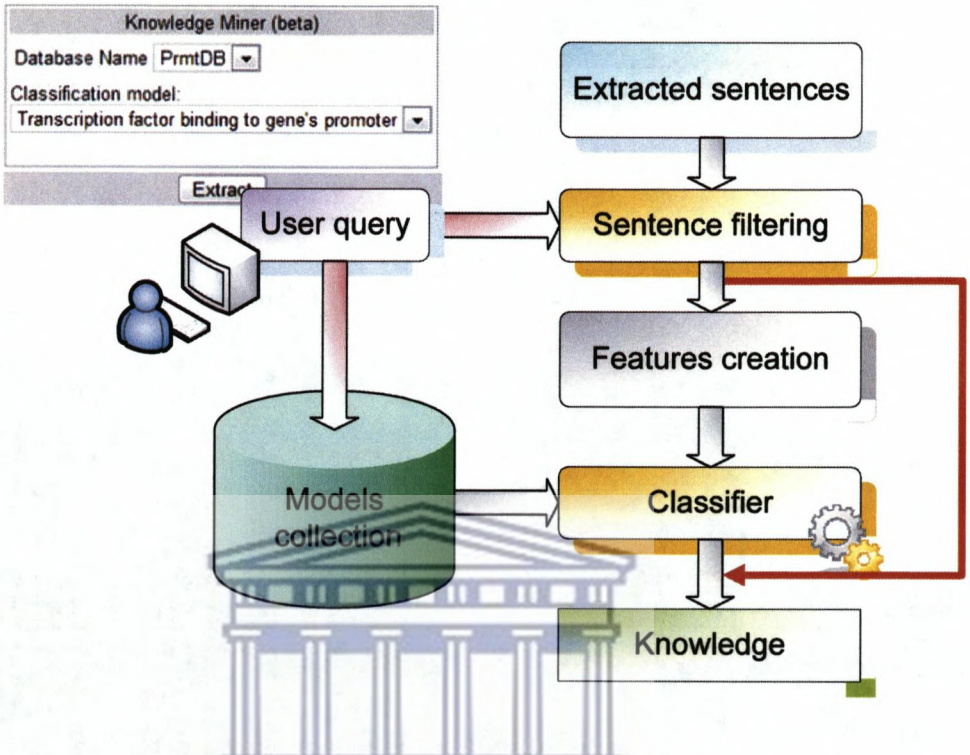


Figure 3.9.1 Knowledge extraction process.

3.10.Chapter Summary

Chapter 3 introduced a concept based knowledge discovery (CobKD) methodology based on the analysis of the sentences and the machine learning model. It was described how sentences can be extracted, stored, curated and used for creating training instances for a supervised machine learning algorithm.

Chapter 4. Extracting information about transcription factor binding to gene's promoter

Without affecting the generalization of earlier described CobKD concept, this chapter shows an original, especially for this study developed methodology for extracting information about transcription factor (TF) binding to gene's promoter. The system architecture and the results assessment techniques used in the experiments were described in previous chapters. This serves as a proof of concept example of how specific types of knowledge can be extracted.

4.1. Problem overview

Critical information for biologists working in the domain of gene regulation is information about interaction of TFs with promoters of genes they regulate. In general, TF are proteins required to initiate or regulate transcription of specific genes and include both gene regulatory proteins and general transcription factors (Hartl and Jones, 2009). TF binds to a promoter (a portion of the DNA that represents a part of the gene regulatory regions) in order to affect the transcription of the gene. To be able to predict potential effects that TFs may cause, either individually or in combination one needs first to know which TF is able to bind to which promoter. This information would reveal a part of important potential of gene transcription regulation.

A variety of techniques are used to obtain information about TF binding sites - short nucleotide motifs to which TFs bind DNA. These techniques are typically divided into experimental ('*in vivo*' and '*in vitro*') and computational ('*in silico*') techniques. In practice those techniques are commonly combined. A researcher may begin a project experimentally and expand it to computational methods. For example, the ChIP-chip technique that combines chromatin immunoprecipitation ('ChIP') with microarray technology ('chip') used to investigate interactions between proteins and DNA in vivo needs a computational follow up analysis. In an another method, the Electrophoretic Mobility Shift Assay (EMSA), the DNA-protein binding is determined *in vitro* and *in silico* algorithms are used to narrow the search for identification of TF binding sites (Hellman and Fried, 2007).

With the development of molecular biology, more techniques are becoming available and more TFs and promoter regions for different kinds of genes have been identified and characterized (Elnitski et al., 2006, Leblanc and Moss, 2009). These results are normally reported in scientific literature. However, obtaining such information experimentally is not simple, and it is slow and costly. The current situation is, that only a limited number of experimentally proved links

between the proteins (TFs) and DNA have been collected in the major databases. Therefore, there is a need for a tool that is able to collect as much as possible of the already reported protein-DNA interactions. Such a collection of protein-DNA interactions in a form of an online database could be utilized by the wider biomedical community.

4.2. Information retrieval

The first step of the experiment was to obtain a document collection that will be used as a source of information for creating training sets needed for a machine learning process. The simple PubMed query: 'promoter' returned 148,542 abstracts as on 25. April 2009.

For the entity recognition, a dictionary-based approach was adopted and the following entities were tagged: human genes and proteins, metabolites and enzymes, chemicals with pharmacological effects, disease concepts and human anatomy. After processing, an annotated database of 145,168 abstracts was created . 3,374 abstracts did not contain any of the concepts from the dictionaries so these were found to be irrelevant to the problem.

4.3. Data preprocessing

In the data preprocessing step, sentences needed for the experiment were extracted from the body of abstracts. The sentence parser produced a database of 1,049,949 sentences on which the concept that defines the rule of extraction (template) was applied:

[gene or protein] [interact] [gene or protein] promoter

This process returned 3,321 sentences. The extraction algorithm used is an essentially simple rule-based NLP that does not explore deep semantics of the sentence. It allows positive and negative examples that follow predefined syntactic structure.

A simple example of a sentence that contains such a structure is:

*Activated PEA3 **binds** to MMP-13 **promoter** and activates its expression.*

... or more complex structures:

*It is noteworthy that the ZAC promoter localized to the CpG island harboring the methylation imprint **associated** with TNDM and methylation of this **promoter** silenced its activity.*

It should be noted that the first sentence expresses a positive case for this study, while the second one is a negative case. The second sentence generally fits the

template but the information conveyed about TF binding to gene's promoter is wrong. However, biomedical text does not always have a simple form. What is obvious to a human reader might not be so evident from the computational perspective:

Binding of chicken ovalbumin upstream promoter-transcription factor 1 to the Nkx2.5 ***binding*** *site suppresses transcription from the calreticulin* ***promoter***.

The sentence is structurally correct, but it does not describe a direct interaction. There could be various ways how one TF can affect the behavior of some gene but the objective is to extract information about what TF can bind to the promoter of what gene. In other words, the following link is of the interest:

TF affect gene by interacting with gene's promoter

or in more structured way

TF→(promoter)→ gene

4.4. Sentences classification

The sentences classification was done by two biologists. In the assessment of quality of the extracted information using only compilation based on sequence structure, 428 sentences were initially annotated that resulted in 152 sentences classified as negative and 276 sentences classified as positive. This amounts to having 63.18% of all annotated sentences being positive (correct). The subsequent effort to annotate more sentences ended up with total of 490 sentences of which 191 were classified as negative and 299 as positive. This provided 61.02% of correct sentences out of all 490 sentences.

4.5. Features evaluation

Annotated sentences were used to generate the training data set with the following features:

$$F_i = (f_1, \ldots, f_m) \text{ or } F_i = (d_{i1}, \ldots, d_{ij}, s_i L_i, t_{i1}, \ldots, t_{ik})$$

where:

- $j = 5$ is number of distance measures for 4 keywords;
- s_i is negative words count;
- L_i is the Levenshtein distance;
- $k = 61$ is the number of top-words.

In a search for a more sophisticated extraction method of higher proportion of correct sentences, individual features were analyzed. The goal was to find to what

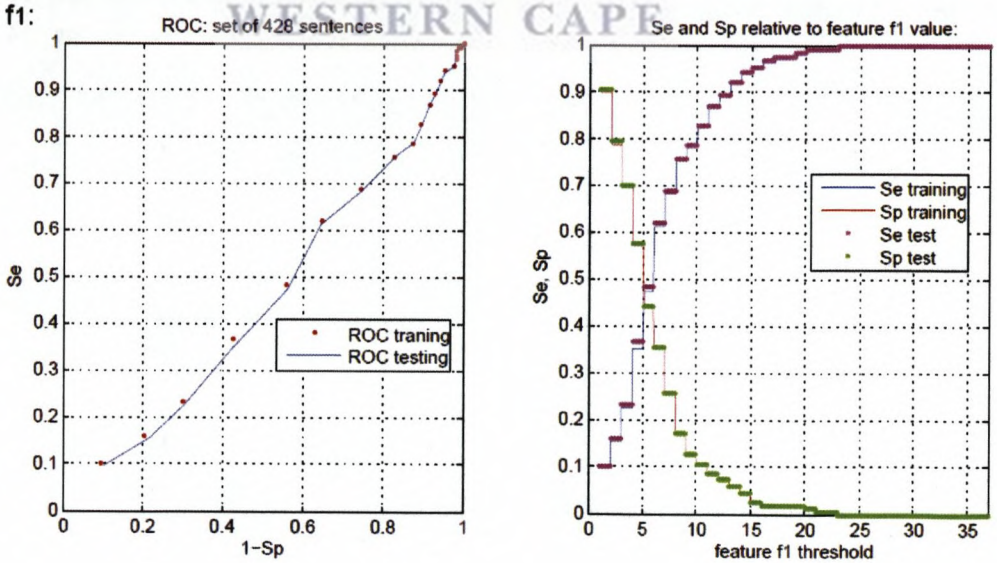
extent individual features are suitable for a simple, threshold-based selection of accurate sentence instances. The following seven features were analyzed:

$$(d_{i1}, \ldots, d_{i5}, s_i L_i) \rightarrow (f_1, \ldots, f_7)$$

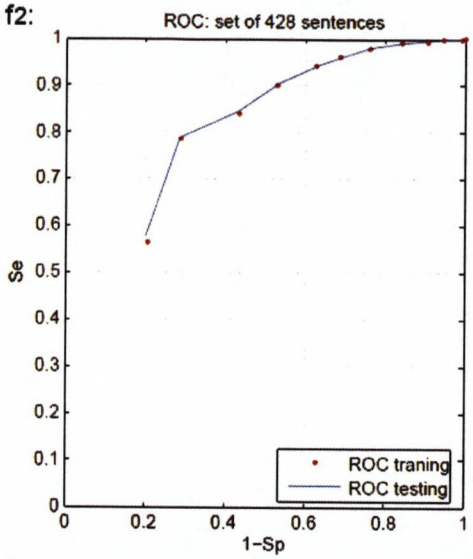
- f_1 = number of words before the first word from *[gene or protein]*.
- f_2= number of words between the first *[gene or protein]* and [interact].
- f_3= number of words between [interact] and second *[gene or protein]*.
- f_4= number of words between the second *[gene or protein]* and *promoter*.
- f_5= number of words after *promoter*.
- f_6= presence or absence of negation words in the sentence.
- f_7= Levenshtein edit-distance.

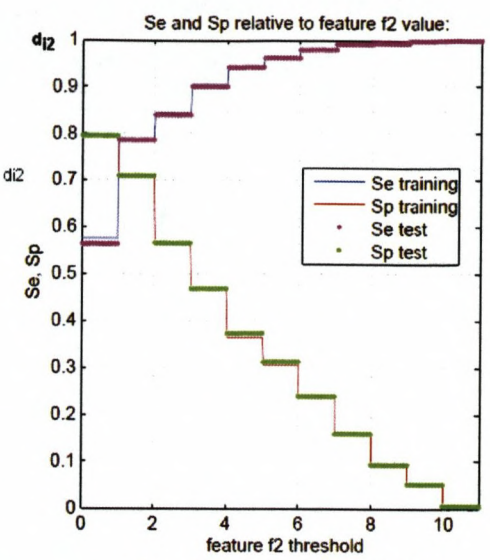
The experiments were based on a set of 428 sentences. For each of the features, the dataset was split randomly to 50% training and 50% testing data. The value of the feature is changed in the range of the feature value in 1000 equidistant steps. Each value is considered a threshold and the number of tp, fp, tn and fn, as well as Se and Sp were calculated for the training set. The same is also done for the test set using the range and threshold values from the training set. Then the original data was again randomly split and the process repeated. This was reiterated 100 times and the results from each individual run were averaged. These results are used to produce Figure 4.5.1. Left side graphs show ROC for training and testing data. The right side graphs show plots of the training and test data sensitivity (Se) and specificity (Sp) against the threshold.

f1:

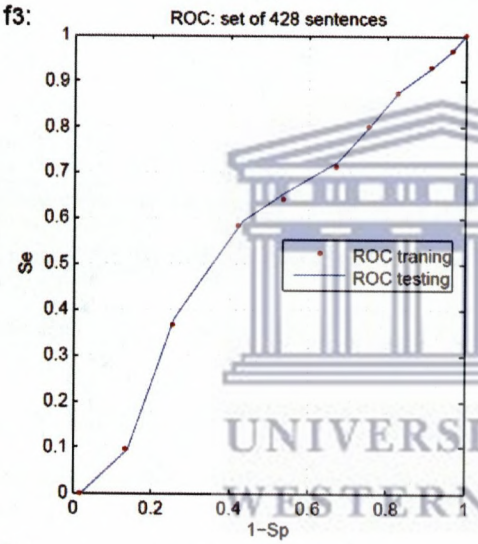


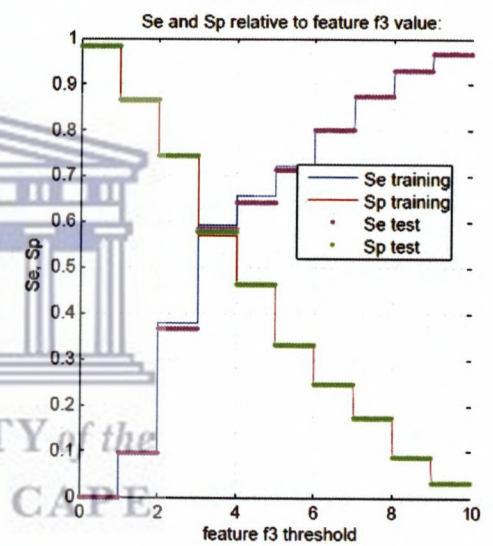
f2:



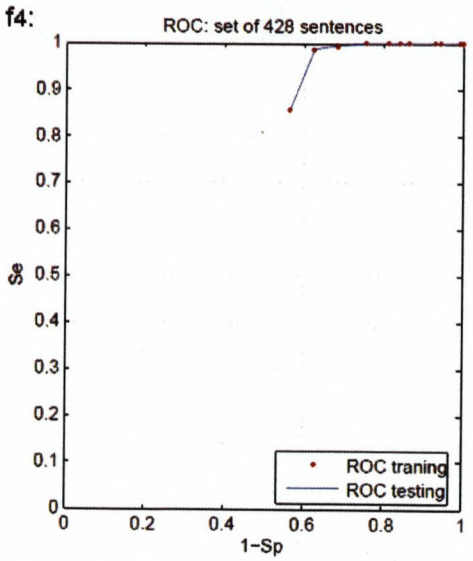


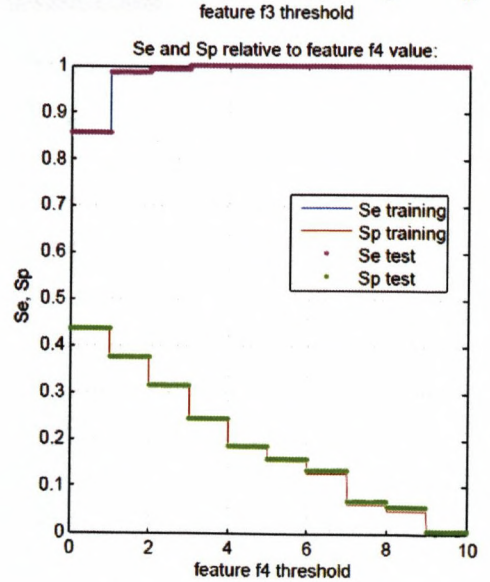
f3:





f4:





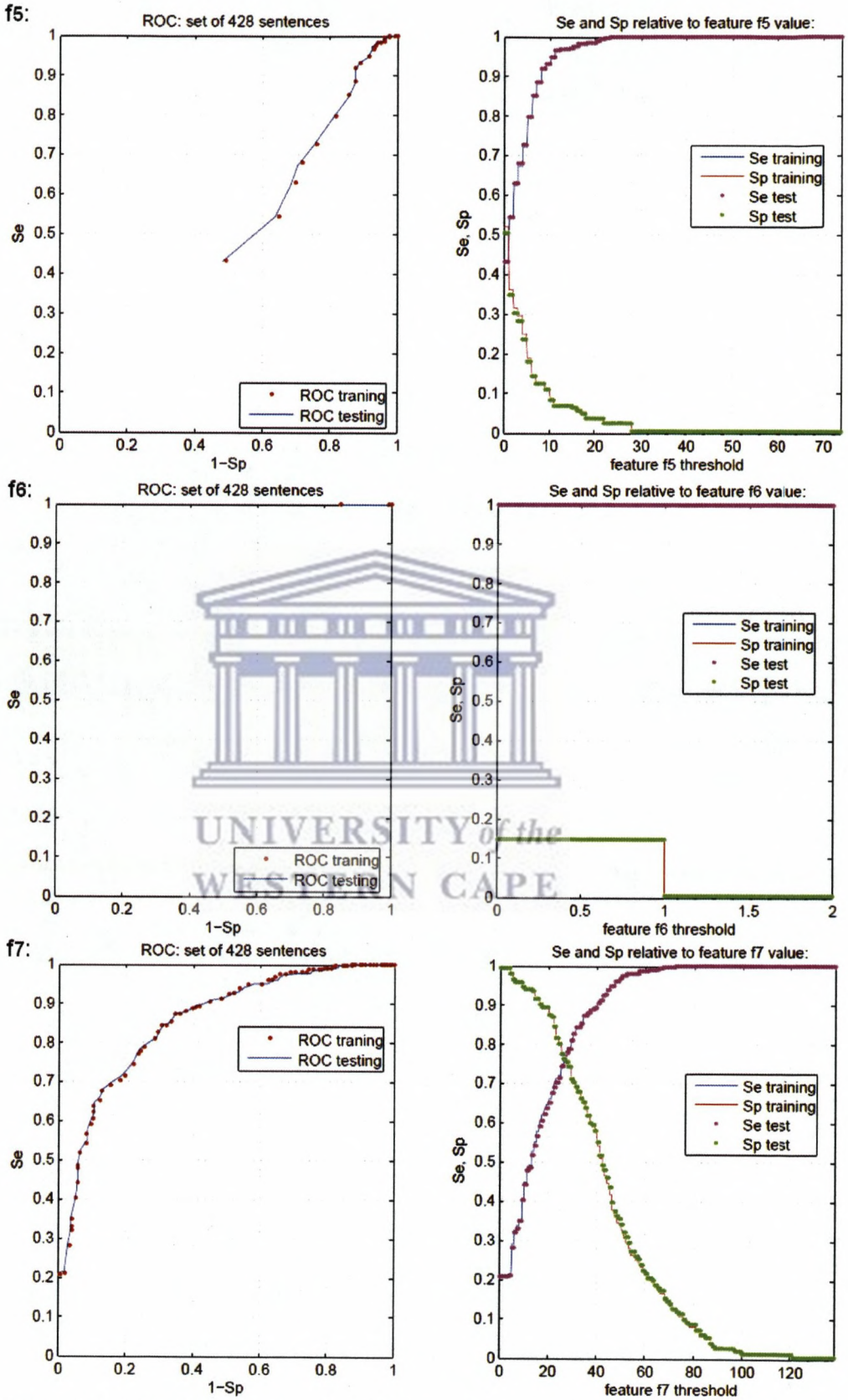


Figure 4.5.1 Usefulness of each of the seven features in separating positive and negative cases.

Figure 4.5.1. shows the usefulness of seven individual features when used independently in separating positive and negative sentence cases. By comparing areas under ROC (left side graphs) it can be observed that the most useful feature is f_7, the Levenshtein distance. Also, as expected the highest intersection point between Se and Sp for the training and the testing data is for the Levenshtein distance (right side graphs). Of the other features, f_2, f_3 and f_4 are of some value, but the other three remain of small or no value if used as single indicators of the correctness of the information extracted from the sentences.

Thus, as a simple method to filter out sentences that are mainly correct, one can use the Levenshtein distance feature with the threshold determined from Figure 4.5.1 . However, the overall accuracy of information might not be satisfactory for automated refining of the extracted data. More sophisticated methods should be used to separate correct from incorrect information contained in the data set.

4.6. Extended features test with LDA

A described problem of sentences classification is a 'two class problem' in the field of machine learning. There are total of 428 example sentences, classified in 276 positive and 152 negative cases. The hypothesis is that features assigned to these sentences contain sufficient information to develop a classifier that can improve automatic extraction of correct information. To evaluate feature weights three experiments were carried out.

- **Experiment 1**: In this experiment seven features were used: $f_1 - f_5$ for information about the number of words between the fixed elements in the template sentence, f_6 for count of negation words in the sentence and f_7 for L-distance.
- **Experiment 2**: In this experiment the initial feature set from the experiment 1 was expanded by set of indicators of most frequent words found in sentences: top-words. 5% or more of these words appears in positively classified and 5% or more in negatively classified sentences (3.7 above). This increased the total number of features to 68.
- **Experiment 3**: In this experiment the feature set was even more expanded by a set of synthetic features. Synthetic features were derived from the features from the Experiment 1 as a set of nonlinear relations among them:

$$\frac{(x-y)}{(x+y+1)}; \quad \frac{x}{(x+y+1)}; \quad \frac{1}{(x+1)}$$

- where x and y represent any of the features from Experiment 1

The classifier selected was a Linear Discriminant Analyzer (LDA), being a practical, simple and robust system. For the evaluation of performance the leave-one-out methodology was used. The results for all three experiments are shown in the table below.

	Se	Sp	Precision	F-measure	Accuracy
L	0.772	0.763	0.854	0.810	0.768
Experiment 1	0.870	0.855	0.916	0.892	0.864
Experiment 2	0.891	0.888	0.935	0.913	0.890
Experiment 3	0.924	0.928	0.959	0.941	0.925
(Chowdhary et al., 2009)	0.71	0.92	0.76	0.74	0.87

Table 4.6.1. Summarized results of experiments 1, 2 and 3.

It can be observed that seven features from Experiment 1 make an improvement of the overall performance as compared to using a single feature - Levenshtein distance. This is to be expected as LDA combines the effect of all seven features and use them simultaneously. Extension of these seven features by those that indicate the presence or absence of words that affect discrimination improves performance to an extent. Finally, addition of features derived from the initial seven features allows relatively high performance of the classifier with the balanced sensitivity and specificity of over 92%. It should be noted that the experiments did not deal with the imbalance of the numbers of positive and negative sentences. The possibility of improving performance by the selection of a more sophisticated classifier or a more sophisticated method of data preprocessing was not explored.

The conclusion of these experiments is that even with the relatively simple classifier, it is in principle possible to obtain a reasonably good classification performance. Obviously, the results are dependent of the data set. Since this is the first study of this type of template sentences, and no other work exists, comparing the system performance to results of others is difficult if not an impossible task. The most similar research is the recent study of protein-protein interaction by Chowdhary and co-researchers (Chowdhary et al., 2009). Although extracting information about protein-DNA interaction, more specifically about protein-promoter interaction, is not the same task (though the sentence structure is quite similar – as shown below), it will be used just for reference. Results obtained in this study will be compared with the results of Chowdhary et al. However, one must have in mind that this comparison has to be considered very cautiously. The differences between the two studies are in the type of template sentences considered, in dictionaries, in the corpus from where the sentences are extracted,

in the features that are used, in the utilized classifier, and in the performance evaluation method.

Chowdhary et al. used the template that consists of two protein names and an interaction keyword (PPI triplet). This template can we written as:

[entity] [interact] [entity]

where *[entity]* represents any protein name or symbol. In our study *[entity]* represent any protein or gene name or symbol so our sentence template is of the form:

[gene or protein] [interact] [gene or protein] promoter

Chowdhary et al. used a Bayesian network and 10-fold cross validation, while in our study the LDA and the leave-one-out validation method were used. Obviously these are very different frameworks in which the experiments were conducted. Although this is not a scientifically proper comparison, it should serve for illustration purposes only.

The best performance Chowdhary et al. obtained is the sensitivity of 71% and specificity of 92%. To explain the differences with results of our study, one can argue that our study deals with a more specific case. This means that the template sentences are more precise, thus providing a chance to have more accurate information from the start. Another component that has possibly contributed to the difference in performance is that in Experiment 3 a set of synthetic features was introduced that Chowdhary et al. did not have. Finally, the Levenshtein distance was not utilized by Chowdhary et al.

However, the claim that methodology presented here is better than the one used by Chowdhary et al., is not justified. Our methodology rather shows better results but for a different problem. Although the effect of using different, more efficient, classifiers was not explored it is reasonable to expect that more sophisticated classifiers could perform much better than LDA.

This study clearly shows that there is a possibility to extract information from biomedical text with very good accuracy by using template based concept descriptions, suitable feature selection and machine learning algorithms.

4.7. Algorithm selection

In the next series of experiments more sentences were annotated and the data set was extended to 490 sentences that contained 191 sentences classified as negative and 299 classified as positive. The Weka (Witten and Frank, 2005) machine learning suite was used to make comparisons between various machine learning algorithms. Initially the selection was: instance-based classifier K* (Kstar),

Multilayer Perception neural networks (MLP-NN), decision tree J48, Naive Bayes and Random Forest. The results are shown in Table 4.7.1.

Algorithm	Precision	Recall	F-measure	ROC area
K*	0.709	0.710	0.710	0.741
MLP-NN	0.713	0.712	0.713	0.787
J48	0.735	0.739	0.735	0.775
Naïve Bayes	0.786	0.788	0.785	0.841
Random Forest	0.795	0.796	0.795	0.837

Table 4.7.1 ML algorithms comparison for the set of 490 sentences classification.

All classifiers performed reasonably well but the best results were obtained by using Naïve Bayes and Random Forest as shown in Table 4.7.1 and Figure 4.7.1.

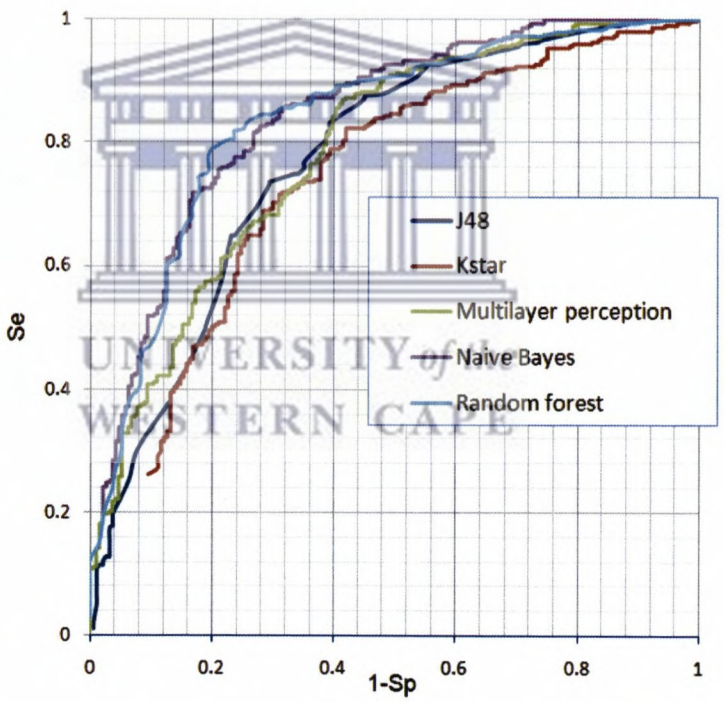


Figure 4.7.1 ROC curves for compared classifiers.

As the imbalance of positive and negative cases is significant it was worth exploring if this imbalance can be utilized in a way that would improve the overall prediction performance. One way of doing it is to synthesize more samples that can mimic but still be different enough from the original 490 cases. At the same time, to increase the specificity of the predictor more negative samples would be required. To achieve this, a method that would oversample the minority class was

needed. The ideal candidate was the Synthetic Minority Oversampling Technique (SMOTE) (Chawla et al., 2002).

The SMOTE algorithm was applied to the whole dataset of 490 instances and generated 872 synthetic cases. A sew set of synthetic cases contained about twice as much negative examples than positive ones. Without going into the algorithm details, these are obtained by interpolating feature values in the feature space covered by positive and negative examples. In the Weka environment the SMOTE algorithm applied three times to the data set using 1.0 as a seed for the random sampling and 5 as a number of nearest neighbors.

The technique resulted in a total of 1362 unique (no instance was repeated) cases. This set consisted of 299 positive and 573 negative synthetic cases produced by SMOTE and the original 490 cases dataset. Finally, from the resultant set of 1362 cases the original 490 cases were taken out leaving a set of 872 synthetic instances.

After testing a few ML algorithms, the algorithm of choice was the instance-based classifier K* (Cleary and Trigg, 1995). The Weka implementation of K* classifier was trained (with the algorithm authors recommended settings) on the synthetic 872 instances. The classifier was then applied to the original 490 cases. To see the system predictive performance a few more experiments were setup:

- **Experiment 4**: same as in the Experiment 1, seven features were used.
- **Experiment 5**: The feature set was expanded by a set of 50 synthetic features (as explained in the Experiment 3) bringing to total the number of features to 57.
- **Experiment 6**: Set of seven features was expanded by top-words features bringing the total number of features to 68.
- **Experiment 7**: The complete set of 117 features (original seven, synthetic and top-words) were used.

Results of experiments with K* algorithm are shown in the table below:

	Precision	Recall	F-measure	AUC
Experiment 4 (7 features)	0.899	0.892	0.893	0.962
Experiment 5 (57 features)	0.912	0.908	0.909	0.970
Experiment 6 (68 features)	0.963	0.961	0.961	0.993
Experiment 7 (117 features)	0.960	0.959	0.959	0.988
(Chowdhary et al., 2009)	0.92	0.71	0.740	0.870

Table 4.7.2 Summarized results of experiments 4, 5, 6 and 7 using K algorithm.*

It is interesting to note that K* algorithm produced Se = 100% and Sp = 100% on the synthetic training set in all cases. On the test set (original 490 sentences instances) different experiments produced different results, but much somehow better than those obtained from the results from the smaller data set of 428 sentences.

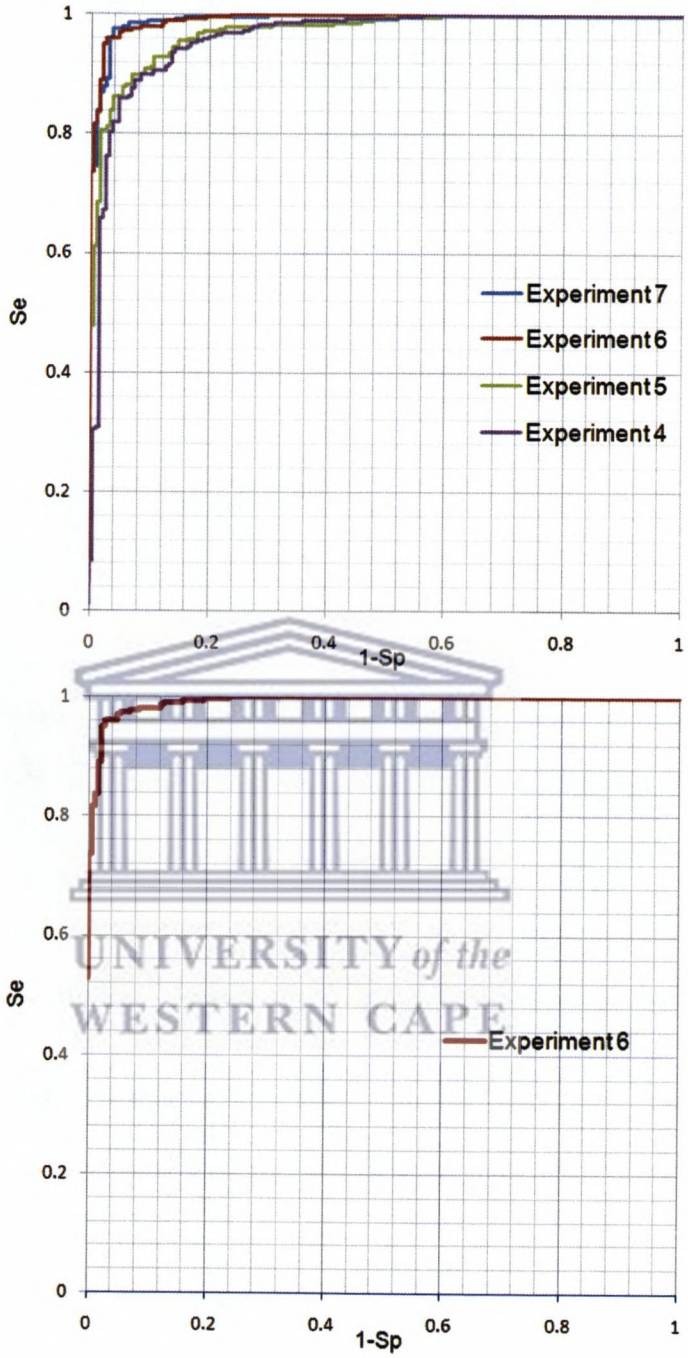


Figure 4.7.2 ROC curves for examples 4 to 7 and ROC for the best performing system in experiment 6.

It can be observed that the ROC is in the upper 90% of absolute scale, which makes it rather good. Synthetically derived features in Experiment 5, improved classification by comparison to Experiment 4 where 7 features were used. The best results were obtained in Experiment 6 with 68 features used, followed by Experiment 7 when all 117 features were used.

The conclusion is that this approach brings the automated assessment of information to the level of human curator accuracy. However, it requires preparation of data and derivation of a model for information extraction. It must be highlighted that the selection of the most relevant features was not used which might have improved the classification performance. However, the existing results show that the preprocessing of samples by described methodology results in an excellent performance that otherwise is not possible to achieve even using the original set of 490 cases as the training set.

The experimental results were implemented in a working framework as will be described in the next chapter. This is of significance because although a considerable amount of studies have been done: researchers rarely go beyond the test-bed (Chowdhary et al., 2009, Koussounadis et al., 2009); applications are short-lived (Tanabe et al., 1999); limited to a controlled set of curated data (Breitkreutz et al., 2008) or subject specific (He et al., 2009). This research aims to bring together theoretical advancements and practical implementation of machine learning techniques in the biomedical field.

4.8. Chapter summary

Chapter 4 showed without affecting generalization of the CobKD concept, a methodology for extracting information about transcription factor (TF) binding to gene's promoter. The process of information retrieval, data preprocessing, deriving and evaluation of features used in ML algorithms were described. A number of experiments were performed by using various set of features and machine learning techniques. The results were compared with the most recent work in this area. The conclusion was that the CobKD approach brings the automated assessment of information to the level of human curator accuracy.

Chapter 5. Implementation methodology

Extensive studies are being done in research of methodologies and algorithms in various fields of biological text mining. However most of the research findings do not always transfer to practical applications. This study aims to close the full circle of research and development by practical implementation of methodologies presented in this thesis.

5.1. Dragon Exploration System (DES)

The implementation guidelines in developing an integrated biomedical text mining software framework that combines named entity recognition, knowledge extraction and information integration has been adopted from the work of researchers that had success in this area (Pan et al., 2006) and updated to match the current technologies:

- to be web based, interactive and easy to use;
- to support multi-user access and collaboration;
- to be able to handle large volume of information;
- to provide suitable interactive summary reports;
- to show association maps in graphical format;
- to be able to generate hypotheses;
- to be able to extract user-defined relationships among biomedical concepts.

User requirements can be summarized in five simple points:

- **Annotate** entities in the document collection.
- **Explore** entities and their relationships.
- **Visualize** association networks.
- **Hypothesize** about new associations
- **Mine** for relationships between concepts.

These guidelines were used to develop the Dragon Exploration System (DES)[3] - an integrated web based biomedical text mining environment based on the methodology, ideas and code behind the Dragon text mining applications (Pan et al., 2004, Bajic et al., 2005, Pan et al., 2006). The application was rewritten and further expanded by an entity association network module and modules for hypotheses generation and knowledge extraction described in this study.

[3] Proprietary software owned by OrionCell cc.

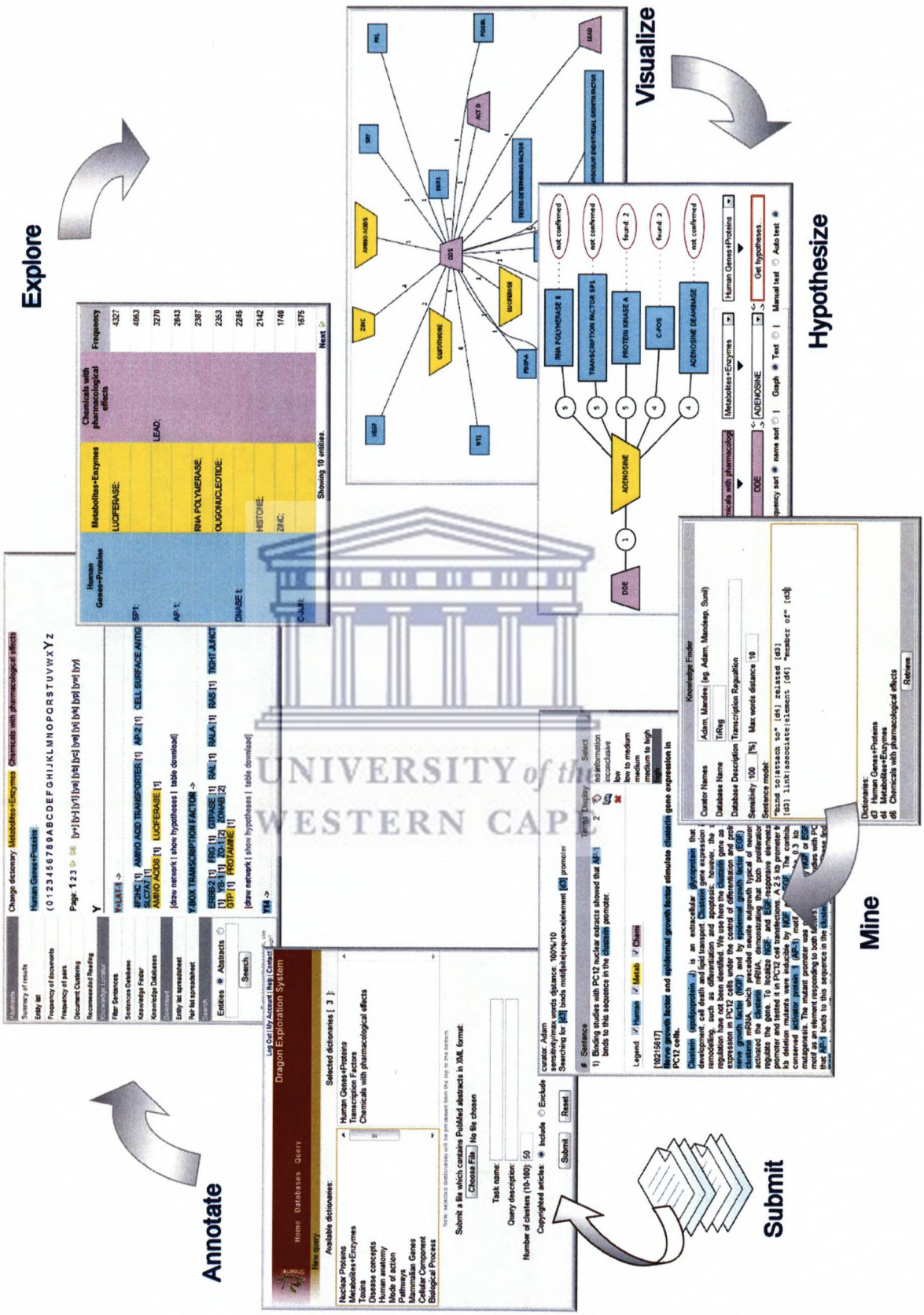


Figure 5.1.1 Working with Dragon Exploration System.

5.2. System architecture overview

This system is designed as a three-tier client-server architecture as shown in Figure 5.2.1. Presentation, logic and data tiers are clearly defined and separated. The presentation tier displays information in the user's web browser. It consists of display and business logic modules for formatting and user events handling. This tier is implemented by using Dynamic Hypertext Markup Language (DHTML) technologies, a combination of HTML, JavaScript and Cascading Style Sheets (CSS). The logic tier combines a number of sub-tiers: web tier for content delivery; business tier for data processing and transaction tier for data access. Separate data tier handles database functions.

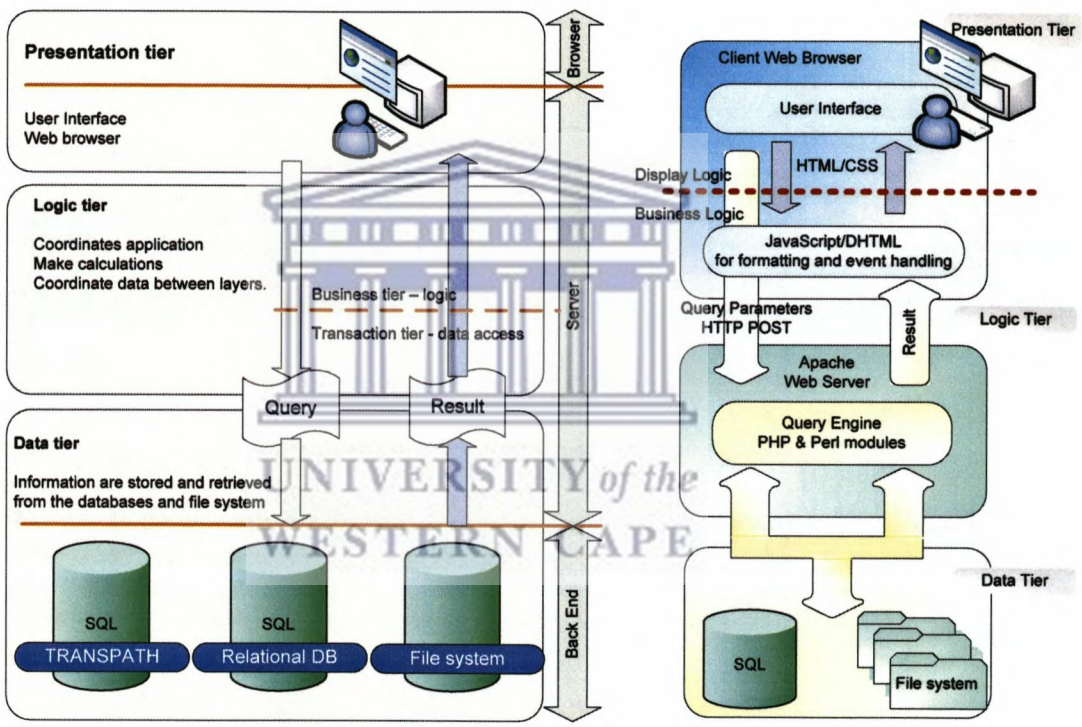


Figure 5.2.1 DES system architecture overview.

5.3. Information retrieval methodology

The first step in automated knowledge discovery is obtaining a document collection from PubMed. The PubMed 'native' retrieval system was seen as the best suited tool as it is designed to retrieve comprehensively all relevant publication based on the user query.

Commonly, a user can retrieve a document collection from PubMed by sending a query to PubMed and downloading the documents. Although the PubMed

retrieval system is primarily an interactive system, when the number of queries and amount of publications is larger, this process can be automated.

Figure 5.3.1 shows a conceptual diagram of the simple technology developed to handle a large number of queries. The user, in advance prepares a list of standard PubMed queries and submits it to the query generator. The list can be manually written as a text file or retrieved from some database (gene names for example). The list is processed and one query at a time sent to PubMed. Each of the queries results in the creation of a separate XML document collection.

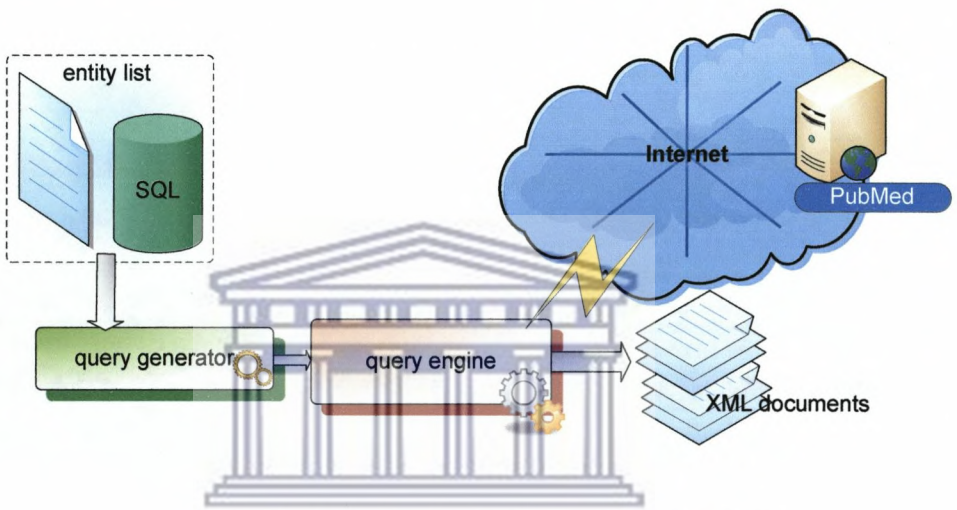


Figure 5.3.1 DES information retrieval system

For example, for the purpose of research in genes implicated in ovarian cancer described later in this thesis, the following template was used:

($Gene_Symbol OR $Gene_Alias..) AND mammal AND cancer.

The variables *$Gene_Symbol*, and *$Gene_Alias* were replaced by real gene symbols and with one or more of aliases obtained by querying the local database. Such queries returned more than a half a million documents organized in more than 300 document collections.

The DES web based interface is then used to submit the document collection as shown in the figure below. The user can focus on the specific biomedical concepts by selecting relevant dictionaries. These dictionaries will be used for annotation purposes.

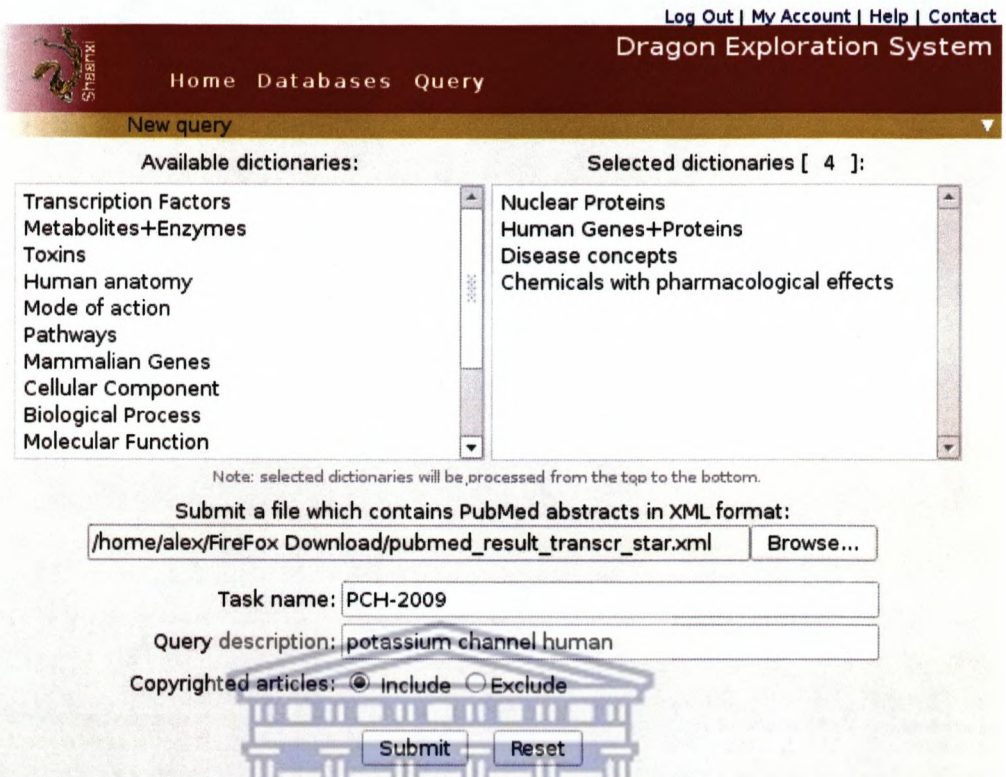


Figure 5.3.2 DES user query interface.

The text pre-processing module extracts the title and abstract information from the submitted XML collection, and creates an internal database that will be used by the entity recognition module.

5.4. Entity recognition methodology

The DES system uses a dictionary-based approach for recognizing biomedical entities combined with simple stemming process for recognizing word suffixes. Dictionaries can provide high accuracy, and when linked to entity information data sources, provide easy mapping to well-known IDs (e.g. to EntrezID or UniProtID) (Yang et al., 2008). They also provide a simple solution for to the spelling variation which is a common occurrence in biomedical literature. The drawback of this method is that its performance depends on the size and quality of the dictionary. The dictionary must be constantly kept up to date as new terms are continuously created. However, groups such as Human Genome Organization (HUGO), Mouse Genome Institute (MGI), Universal Protein resource (UniProt), and the National Center for Biotechnology Information (NCBI) constantly collect and organize information on gene and proteins so dictionaries can be relatively easy to update using these genomics databases.

To allow the user to focus on a specific topic, the DES uses a collection of manually curated dictionaries containing various biomedical terms as shown in Table 5.4.1. The dictionaries were developed by Bajic V.B in 2005 (Bajic et al., 2005) and are continuously being upgraded through various research projects including DES. This collection can be presented as:

$$D_1 = \{d_{11}, d_{12}, d_{13} \ ...\}$$
$$D_2 = \{d_{21}, d_{22}, d_{23} \ ...\}$$
$$...$$
$$D_n = \{d_{n1}, d_{n2}, d_{n3} \ ...\}$$

where D_i represents a dictionary number, and d_{ij} term j in dictionary i.

Dictionary	Terms
Biological Process	1113
Cellular Component	276
Chemicals with pharmacological effects	47478
Disease concepts	97016
HIV	270
HLA alleles	3477
Human anatomy	1290
Human Genes+Proteins	289240
Mammalian Genes	142930
Metabolites+Enzymes	70547
miRNA Human	2833
Mode of action	427
Molecular Function	1384
Nuclear Proteins	21884
Pathways	954
TB-HIV drugs	215
Toxins	8331
Transcription Factors	18222

Table 5.4.1 DES Dictionaries

The entity recognition module uses one dictionary at a time, with the dictionaries being processed sequentially. Once an entity has been recognized and tagged, it is no longer available for processing, even if present in other, still to be processed dictionaries. After processing of all the dictionaries is complete, the PubMed abstracts are annotated with the entities from the dictionaries as identified in the text.

For entity recognition the case insensitive 'longest matching' algorithm is used – if more than one possible match is found, the entity with the longest number of

matching characters takes precedence. A previously developed data structure (Bajic et al., 2005) was used and optimized for accuracy and processing speed (Figure 5.4.1).

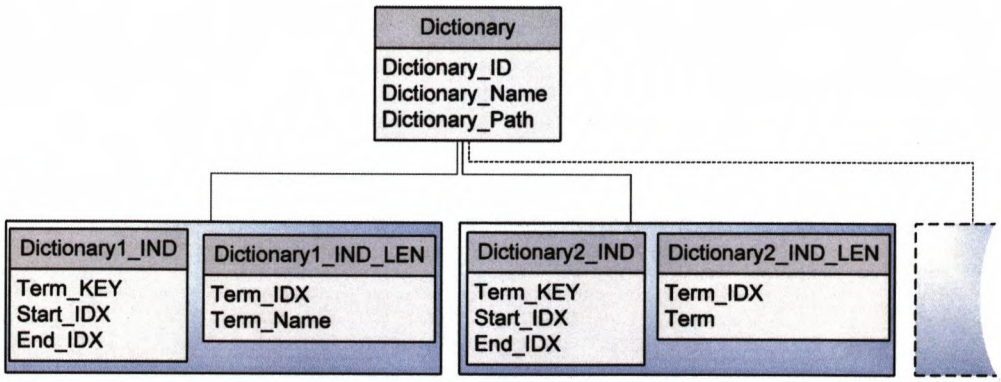


Figure 5.4.1 DES dictionary structure

The root of the dictionary tree contains a record of the dictionary ID, name and the physical path on the disk. The table Dictionary(x)_IND_LEN contains dictionary terms sorted by the first three letters of the term and by the term length. Each of the terms is indexed with Term_IDX followed by the term name (Term_NAME). Data needed for the fast access to the term are kept in the table Dictionary(x)_IND. The TERM_KEY in this table represents three starting letters of the term, START_IDX points to the starting, and END_IDX to the ending record of all terms beginning with that letter as shown in Table 5.4.2.

Dictionary(x)_IND	
Term_KEY	CEL
Start_IDX	22
End_IDX	26

⇨

Dictionary(x)_IND_LEN	
Term_IDX	Term_Name
22	CELL SURFACE (SENSU MAGNOLIOPHYTA)
23	CELL WALL (SENSU MAGNOLIOPHYTA)
24	CELL SURFACE
25	CELL PLATE
26	CELL WALL

Table 5.4.2Three letters key mapping to term name.

By having pointers stored in tables and terms sorted by descending length and alphabetical order, the matching software has fast access to term names and ability to discard non-terms by performing an only three letters match.

Text mining systems commonly perform term recognition and entity mapping at the same time. (Robert Gaizauskas, 2000). Since the DES is designed to be a generic application, the mapping module is dynamic and allows linking to various

resources. The initial annotation has been expanded by dynamically linking tagged entities with the databases that provides additional genes, gene ontology and protein details as shown in the figure below.

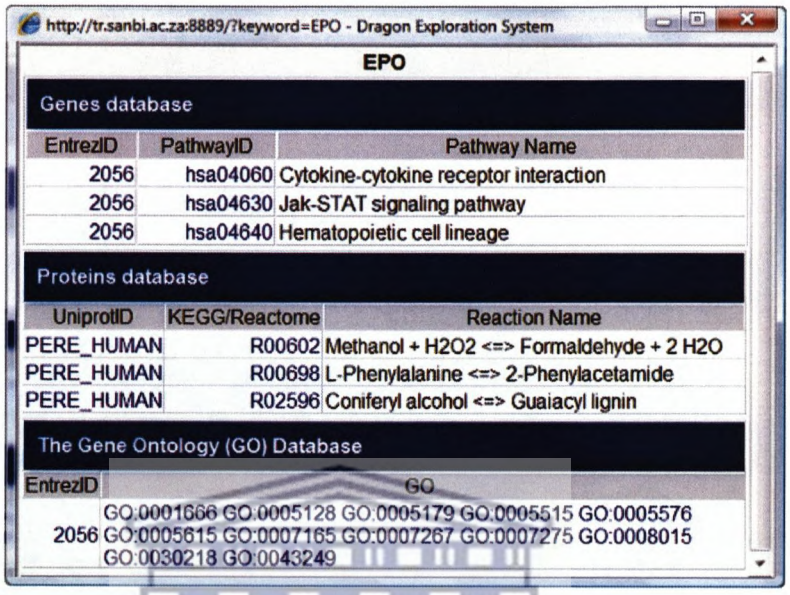


Figure 5.4.2 Gene name (EPO/erythropoietin) mapping to external databases.

To achieve this task, the mapping module is linked to a local database that contains gene and protein identifiers and publicly available databases URLs that accept those identifiers as an input. The system provides term mapping to the following online gene/protein repositories:

- **Entrez Gene** database - gene-specific database at NCBI. This database provides comprehensive information associated with the selected gene (Maglott et al., 2007).
- **Kyoto Encyclopedia of Genes and Genomes (KEGG).** A collection of databases that provide a reference knowledge base for linking genomes to biological systems. It provides information in the genomic space (KEGG GENES), the chemical space (KEGG LIGAND) and connection diagrams of interaction networks and reaction networks (KEGG PATHWAY) (Kanehisa et al., 2006).
- **Universal Protein Resource (UniProt).** A comprehensive resource for protein sequence and functional information ((UniProt_Consortium, 2009).
- **Gene Ontology (GO).** A database that provides three detailed, structured vocabularies of terms (ontologies) regarding molecular functions that gene products normally carry out, the biological processes that gene products are

involved in and the subcellular locations that gene products are located in (Barrell et al., 2009).

5.5. Entity database design

During the named entity recognition process the system creates the database that contains entities found in document collection. Conceptually the entity database is based on Entity[4]-Relationship Model (ERM) but to ensure easy portability of 'ready-made' data it is encoded in 'flat files'. It consists of tables (Figure 5.5.1) that use two main indexes: dictionary index (Dictionary_ID) and Publication ID (PMID). The original PMID is kept from the PubMed database to allow seamless link to the original source.

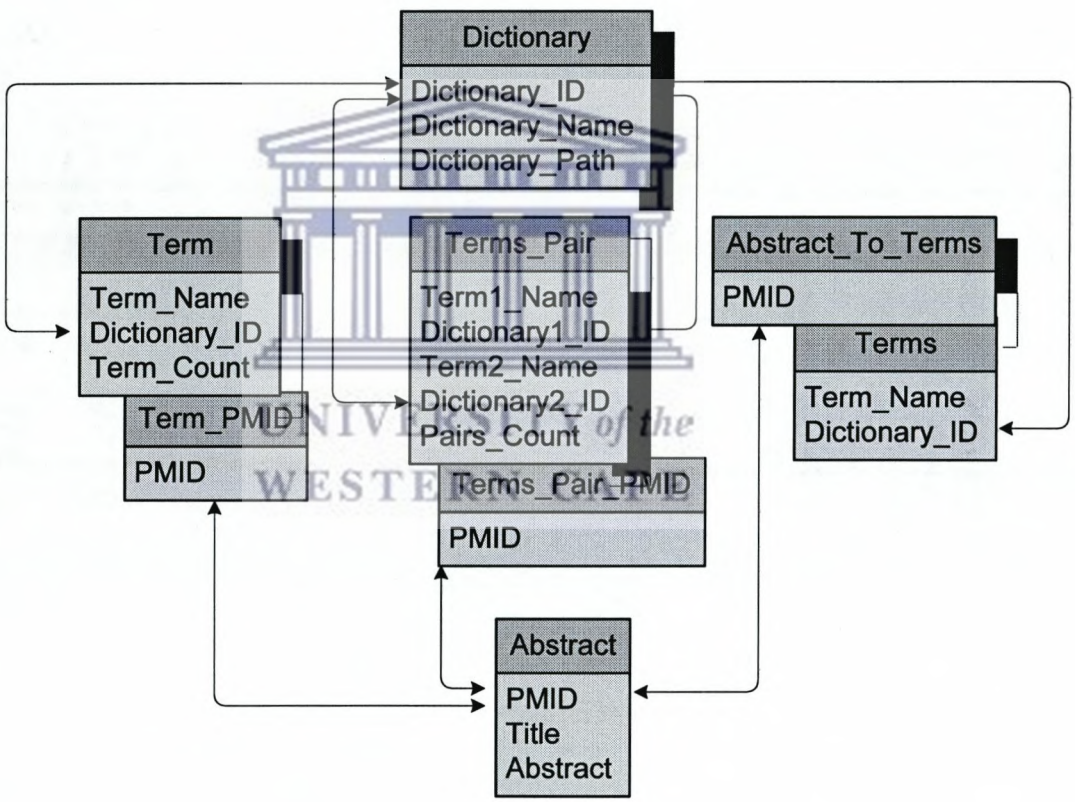


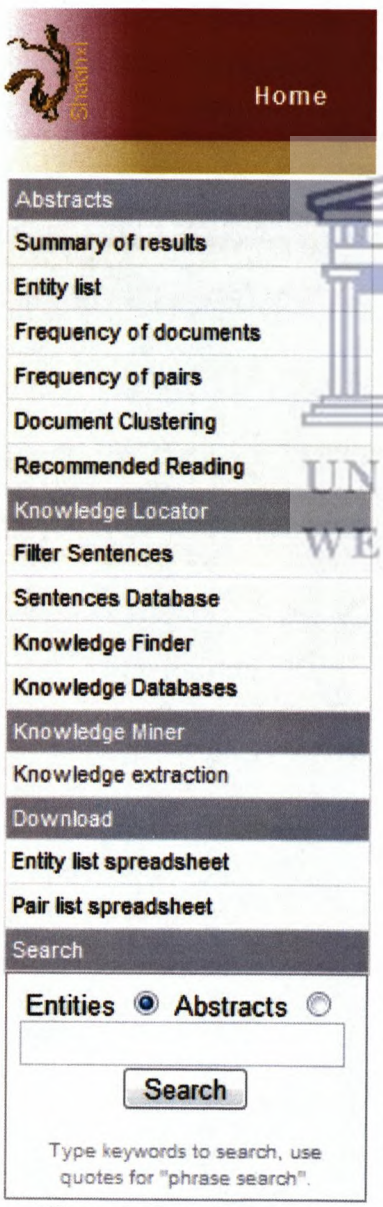
Figure 5.5.1 The core DES database tables.

- Table Abstract contains PMID, title and abstract of publications retrieved by the IR process. The PMID is used as a foreign key to link every entity to its abstract of origin.
- Table Dictionary contains other key Dictionary_ID that is used to identify the type of entity (gene, protein, chemical, etc…).

[4] entity is a software engineering term in this context

- Table Term contains the term name, the dictionary the term belongs to and the total frequency (count) of appearance within the document collection. A child table *Term_PubID* records all PubIDs terms appears in.
- Table *Term_Paris* contains pair of terms, including their dictionary IDs, and total frequency (count) of pair appearance within the document collection. A child table *Term_Pairs_PubID* records all PubIDs terms pair appears in.
- The table *Abstract_To_Terms* maps a list of terms with their dictionary IDs to PubID they appear in.

5.6. Reports



Access to the annotated database and other system functionality is summarized in a menu presented to the user.

Summary of results presents an alphabetically sorted dictionary of entities paired with correlated entities within the same abstract. The related entities are grouped by the topic (dictionary) and sorted alphabetically and by the frequency of appearance. The user can change viewpoint by selecting a different dictionary.

Entity list presents list of entities sorted by the frequency of appearance.

Frequency of documents displays documents sorted by number of entities found.

Frequency of pairs list pairs of entities sorted by frequency of their co-occurrence.

Document clustering: list of document clusters. The documents are clustered according to their feature vectors. The feature vector contains a name of the entity followed by its weight. The weight equals to entity frequency in the cluster divided by the total number of documents in the cluster.

Recommended reading: top ten documents selected according to frequency of entity co-occurrence.

Knowledge locator group can filter sentences that are of interest for the

researcher. It is also used for creation of training sets for the machine learning algorithms.

Knowledge extraction is sentence based knowledge extraction by using machine learning techniques.

Entities and entity pair spreadsheets allows the user to download entity database summarized in two spreadsheets

Search allows user to search for entities and displays them as a list with other correlated entities or inside the abstracts they appear in

In addition, the software allows visualization of entities correlation. The researcher can zoom into the chosen entity by expanding the correlation tree and selecting the subsets of dictionaries. Some of the features will be described in more detail in the next chapter. Menu section 'Knowledge Locator' refers to knowledge discovery based on the algorithms described in previous chapters.

5.7. Generating hypotheses

The first literature based hypothesis generation was presented by Swanson (Swanson, 1986). This type of linking of two seemingly disjointed informations through a common, shared information is known as Swanson ABC model.

As shown in the Figure 5.7.1, the premise is that if concepts A and B are related, and B and C are related then A and C might be indirectly related. At the same time, in published literature concepts A and C do not, or very few times appear together.

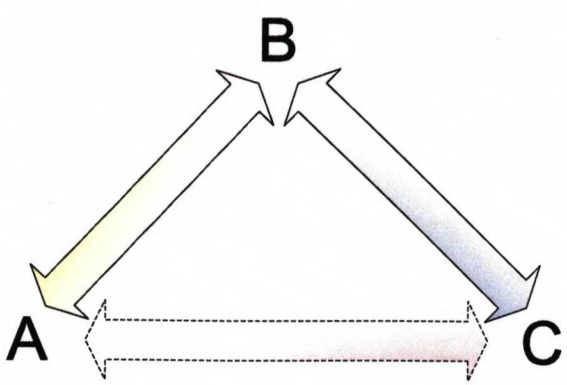


Figure 5.7.1 Swanson's ABC model

From the researcher's perspective, Swanson's ABC model can be seen from two angles: hypothesis generation or 'open discovery process' and hypothesis testing or 'closed discovery process' (Marc Weeber, 2001) as shown on the Figure 5.7.2.

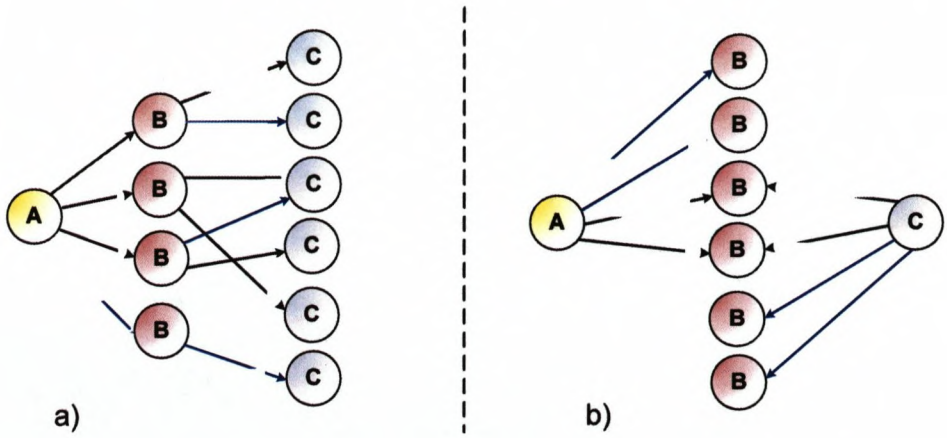


Figure 5.7.2 Open a) and closed b) discovery process (adopted from (Marc Weeber, 2001))

The open discovery process starts from A and results in C. The researcher is trying to find interesting concepts (B) (e.g. physiological processes) that links his researched topic (A) disease for example, with pharmacological substance (C). Among possibly many links some might have biomedical sense (black lines) and some not (blue lines). When verifying the hypothesis, in a closed discovery process, the search starts in both directions resulting in overlapping concepts (B). Black arrows suggest potentially interesting pathways of discovery; the blue ones are unsuccessful pathways.

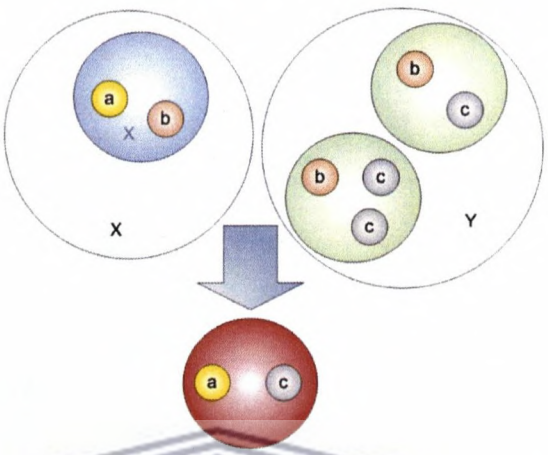
In their implementation, very often both of the approaches are limited in dealing with possible explosion of hypotheses and lacking a method of ranking them (Smalheiser et al., 2009a, Hristovski et al., 2006, Yetisgen-Yildiz and Pratt, 2006).

5.8. DES implementation of the ABC model

This study implements an ABC model as an entity based open discovery approach with automatic hypotheses verification. This approach allows the user to closely control concepts used for hypotheses generation and provide a ranking method.

When a relation between entities is defined by simple co-occurrence in the same abstract, an association hypothesis can be suggested where two entities, **a** and **c** are both linked by a third entity **b**. This is characterized by subsets of documents **X** where **a** and **b** co-occur together, and likewise by subsets **Y** where **b** and **c** are

both present. However, **a** and **c** are not found present together in any of the documents inside the analyzed collection. Therefore it can be hypothesized that **a** and **c** can be inferred to be linked together through an intermediate entity, **b** (i.e. a link is inferred of the form **a – b - c**).



Thus:

$$\forall a,b \in X, \forall b,c \in Y, a \notin Y, c \notin X, aRb \wedge bRc \ \exists \ aRc \quad (1)$$

For predetermined entities **a** and **b**, the following matrix can be written:

$$\begin{Vmatrix} (a, w_{ab}) & (b, w_{bc1}) & (c_1, w_{c1a}) \\ (a, w_{ab}) & (b, w_{bc2}) & (c_2, w_{c2a}) \\ \cdot & \cdot & \cdot \\ (a, w_{ab}) & (b, w_{bcn}) & (c_n, w_{cna}) \end{Vmatrix}$$

Matrix elements are vectors consisting of entity names associated with a weight, usually a frequency of two co-occurring entities. A hypothesis n in the last column is valid if entity c_n satisfy the following condition: if $w_{cna} = 0$, hypothesis is valid. If $w_{cna} > 0$, the link between **a** and **c** already exist in document collection. The full hypotheses space will consists of matrices that will cover all possible combinations of entities **a**, **b** and **c.** Entities can exchange places so, for example, entity **c** can move to the first column thus shifting focus of the hypotheses generation. Figure 5.8.1 shows network representation of a simple matrix consisting of two entities generating three hypotheses (n = 3). Typically, the number of entities can vary from thousands to tens of thousands. It is important to mention that mathematical validity does not always means validity in biomedical sense.

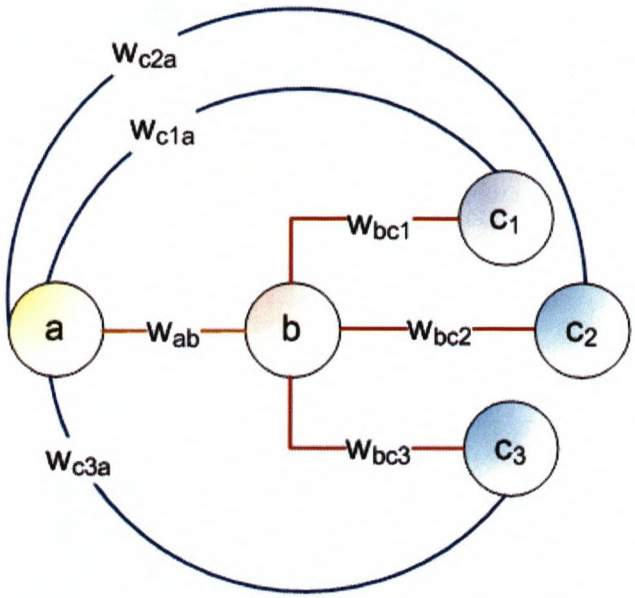


Figure 5.8.1 Entity based hypothesis generator.

Later in the thesis it will be shown how, for example, the user can generate hypotheses that could suggest links between chemicals, genes and diseases. Each hypothesis will be ranked by a number of abstracts containing chosen entities (weights $w_{ab}, w_{bc1}, w_{bc2}, ...$), and will be immediately tested online with the rest of the PubMed repository (weights $w_{c1a}, w_{c2a}, w_{c3a}, ...$). If such an abstract or abstracts are found, the hypothesis might be true. If not found, the hypotheses has possibly never been tested which makes it attractive for further research.

5.9. Academic Biomedical Extreme Programming (ABXP)

In this study, to take advantage of the multidisciplinary environment in which biomedical software is usually created a fresh approach to software development has been suggested. A biomedical project usually relies on collaboration. It requires expertise in various fields such as medicine, biology, computer science and statistics. Software teams, especially in an academic environment, are small and fluid. Members of the team are often driven by limited time frames and competing interests. Furthermore, developments in both biomedical and computer science fields are rapid, requiring software development methodologies to adapt quickly and efficiently to scientific advancements and new technologies.

In an environment that requires an adaptable approach to software engineering agile software development seems an obvious choice. It allows flexibility with ever changing technologies and advancement in knowledge, effective collaboration, prioritizing of various interests and can deliver results in limited

time. Various studies have shown agile methods to be very well suited for biomedical and academic software development (Kane et al., 2006).

The term 'agile' refers to a group of lightweight development methodologies that promote the following: collaboration, development through iterations, self-organization, accountability and process adaptability throughout the life-cycle of the project. The term refers more to a philosophy behind the software development rather than a formalized set of rules and procedures. The Agile Manifesto published in early 2001 summarizes this principles as follows (Beck et al.)(Beck et al.)(Beck et al.)(Beck et al.)(Beck et al.)(Beck et al.):

"***Individuals and interactions*** *over processes and tools.*
Working software *over comprehensive documentation.*
Customer collaboration *over contract negotiation.*
Responding to change *over following a plan.*"

Agile methods promote cyclic, rapid software delivery by small teams usually located in open workspace that interact frequently. Each cycle includes planning, requirement analysis, design, coding, testing and possibly deploying the code into production. There are a number of different methodologies, each emphasizing different aspects of the Agile Manifesto. The most well know is Extreme Programming (XP) (Ambler, 2002), but there are many others.

Extreme programming (XP) is an agile software development methodology that takes successful software development principles and practices to the extreme level. Although XP focuses on delivering business values that sometimes clash with motivations behind scientific research, it has been successfully used in this environment (Wood and Kleb, 2003). Businesses software development is driven by increasing productivity and making revenue more quickly. In academic environment software development is usually driven by grants, software releases tied to publications and a reward structure based on peer reviews of the researcher's stature in the field. Some work has been done in adopting XP to suite academic needs in the biomedical field (Pitt-Francis et al., 2008).

XP core principles that distinguish it from other methodologies are (Beck and Andres, 2005):

- Short development cycles and early and continuing feedback.
- An incremental plan that evolves.
- Flexible schedule implementation responding to business needs.
- Reliance on oral communication, tests and code to describe system structure and intent.
- An evolutionary design process.
- Close collaboration of development team members.

- Reliance on practices that work with both the short-term instincts of the team members and the long-term interests of the project.

In addition, a non formalized approach makes XP a good candidate for software development method in a scientific environment.

5.10. ABXP principles

During this research, instead of following and modifying each of the XP principles to suit academic software development in bioinformatics a method based on Agile Manifesto and core XP principles has been developed and named the Academic Biomedical Extreme Programming (ABXP). Figure 5.10.1 illustrates the philosophy behind the ABXP:

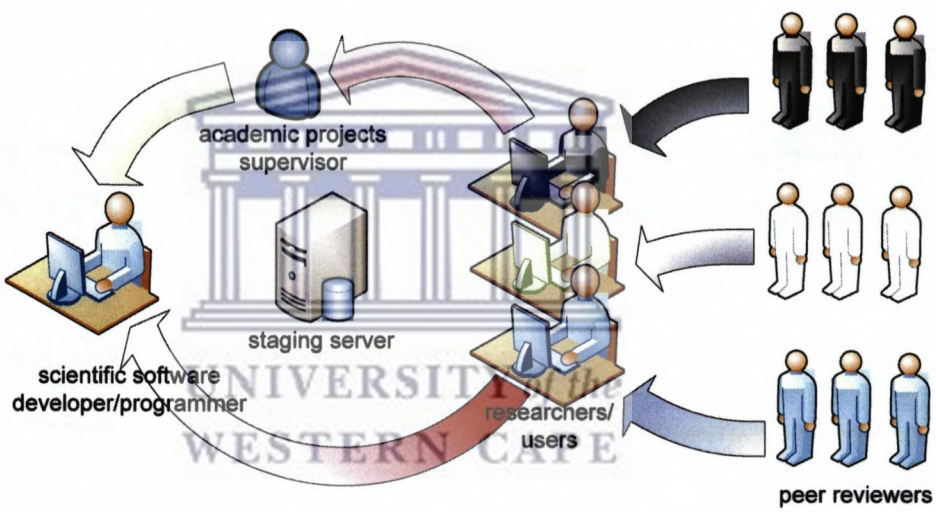


Figure 5.10.1Academic Biomedical Extreme Programming

The principles of ABXP are:

- Close collaboration.
- A software developer works in a team that consists of experts from various biomedical related fields. It is in academic tradition for scientists to have a fair degree of independence to pursue their own research, so the first challenge ABXP is facing is the teamwork. To promote informal communication the team is located in an open space office. The team is managed by the academic project supervisor who initiates frequent formal or informal meetings.
- Evolutionary design process and incremental planning.

- Science research is an iterative process that often includes trials and errors. ABXP starts software development with simple but functional design that evolves as understanding of problems inherent to the research project is redefined. The requirement documents are short and descriptive. The team members constantly refine details through verbal communication, instant messaging, or email.
- Short development cycles.
- The software development cycle is short, measured in weeks, sometimes days. Each version is fully functional giving the user opportunity to interact with the software and get better understanding of the program features. All releases are hosted on the staging server.
- Continuous code refactoring.
- The programmer revisits a code, improving its effectiveness, flexibility, robustness, simplicity and readability without changing its basic functionality.
- Frequent testing.
- All team members test the program all the time by using exploratory testing method (Bach, 2003). This intellectually stimulating method emphasizes academic values like knowledge, diversity of ideas, critical thinking and personal freedom. Team member develop tests, interact with the program and report results. The programmer receives feedback from each team member and the supervisor. Less frequent feedback, which is received by researchers from peer reviewers, is passed on to the programmer. Upon project publication, the program is moved to the production server and updated with major releases.

ABXP has shown to be valuable method for the DES and number of other DES-based research projects development. These projects will be presented in the next chapter.

5.11. Chapter summary

Chapter 5 described practical implementation of concepts described in previous chapters: Dragon Exploration System (DES), an integrated web based biomedical text mining environment. Ideas behind the implementation of a hypothesis generator based on the Swanson ABC model was presented. In addition, this chapter describes methodology used for software development.

Chapter 6. Integrated text mining framework: Case studies

The implemented framework has been successfully applied to a number of research studies, with three of them published (Sagar et al., 2008, Kaur et al., 2009, Essack et al., 2009) and one awaiting publication. By referencing these case studies, this chapter presents results and critical assessment of the DES as an integrated text mining system.

6.1. Database for exploration of sodium channels in human

A collaborative effort between group of biologists and bioinformaticians produced the first publicly available database that supports exploration of sodium channel related information. Dragon Database for Exploration of Sodium Channels in Human (DDESC) provides comprehensive information related to sodium channels regarding genes and proteins, metabolites and enzymes, chemicals with pharmacological effect, toxins, diseases and human anatomy and their possible relations and interactions (Sagar et al., 2008).

Molecular passageways, ion channels, control the flow of ions in and out of cells and they are responsible for many features of a nerve cell's electrical behavior. Neurons are capable of carrying electrical signals over long distances because of their ability to generate an action potential. This is a regenerative electrical signal whose amplitude does not attenuate as it moves down to an axon. Sodium ions (Na^+) influx in particular, is responsible for the rising phase of the action potential (Kandel et al., 2000). This influx is possible due to existence of sodium channels, the proteins that conduct Na^+ ions through a cells plasma membrane.

Sodium channel proteins and related genes could have a significant effect on human health. The mutations in genes coding for sodium channel proteins have been linked with a number of inherited genetic disorders called 'sodium channelopathies'. In spite of the importance of sodium channels, there was no publicly available resource that would serve as repository of the sodium channel related information.

6.2. Text mining methodology

The document collection needed for the database was downloaded from PubMed. The query:

('sodium channel' OR 'sodium channels') human

returned 5,243 documents that were analyzed by the DES. For the tagging purpose six manually curated dictionaries were used: Hhuman genes and proteins, Metabolites and enzymes, Toxins, Chemicals with pharmacological effects, Disease concepts and Human anatomy. Some dictionaries contain broader contexts than their name suggests. For example, the Disease concepts dictionary contains not only disease names and their common names, but also other terms related to diseases like pain, speech delay, genetic predisposition, neurodegenerative etc. to mention a few. Concepts related to the Human anatomy dictionary is not only limited to human anatomical regions, but also includes biological terms to define cellular components and terms that are related to localization of the disease, as well as disease states and conditions e.g. chromosome, cell membrane, cytoplasm and blood etc. Figure 6.2.1 shows how tagged entities are spread among dictionaries.

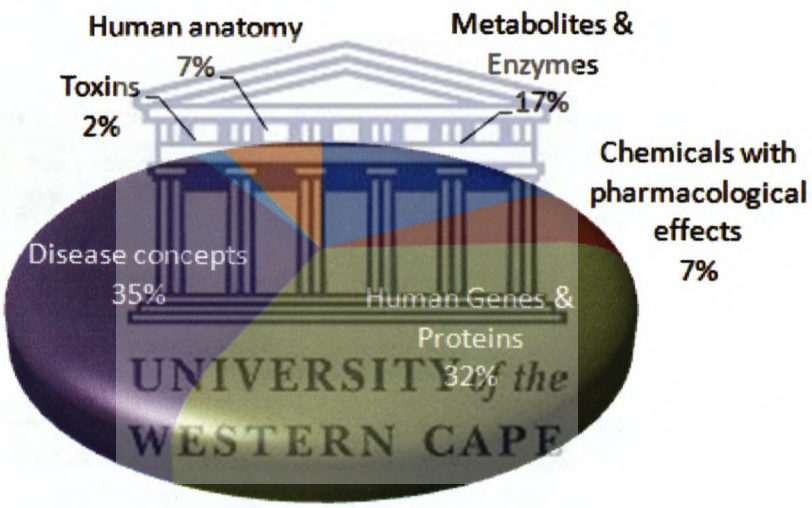


Figure 6.2.1 Tagged entities spread among dictionaries.

6.3. DDESC user interface

DDESC uses adopted an version of DES user interface as shown in the figure below. The user is presented with a number of DES standard reports that provide the insights into various aspects of information and knowledge contained in the analyzed set of documents.

Figure 6.3.1 DDESC user interface.

Figure 6.3.2 shows results of a search for Flurothyl , a liquid convulsant. The results display the entity paired with co-occurring possibly associated, biomedical entities as found in the literature.

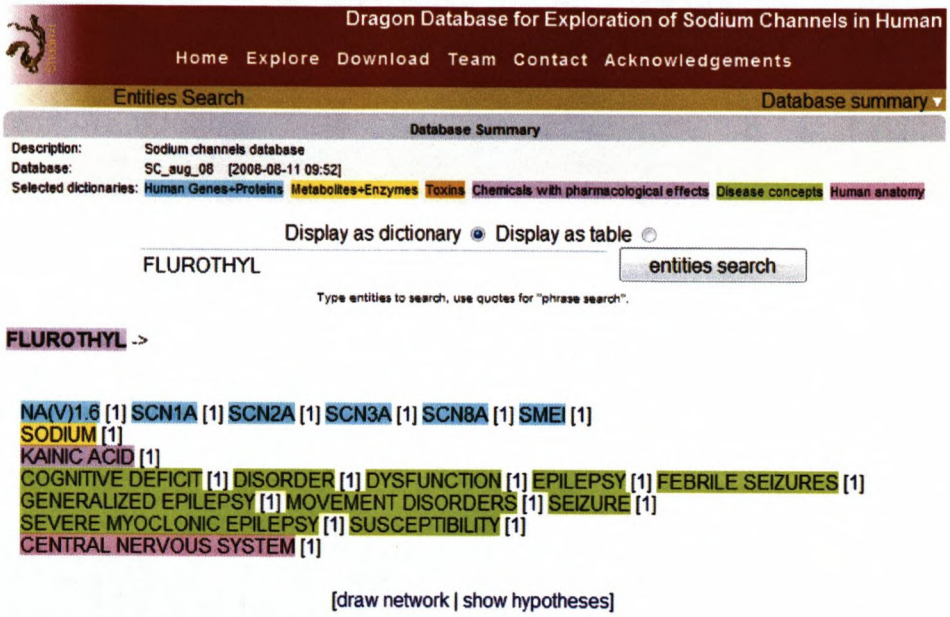


Figure 6.3.2 Potential association of Flurothyl with other entities (genes/proteins, metabolites/enzymes, disease concepts and human anatomy). Clicking the number next to entity will open relevant abstracts.

This system allows visualization of the text presented associations. Color-coded biomedical entries are interconnected with weighted links representing frequency of appearance of an entity and its neighbors in abstracts (Figure 6.3.3). By clicking on a node, the user gets relevant abstracts containing the selected term, and associated terms (surrounding nodes in the network). In addition, the user can 'zoom' into the network by selecting classes of dictionaries that will appear in the association as well as reduce the complexity of the network by removing less weighted nodes.

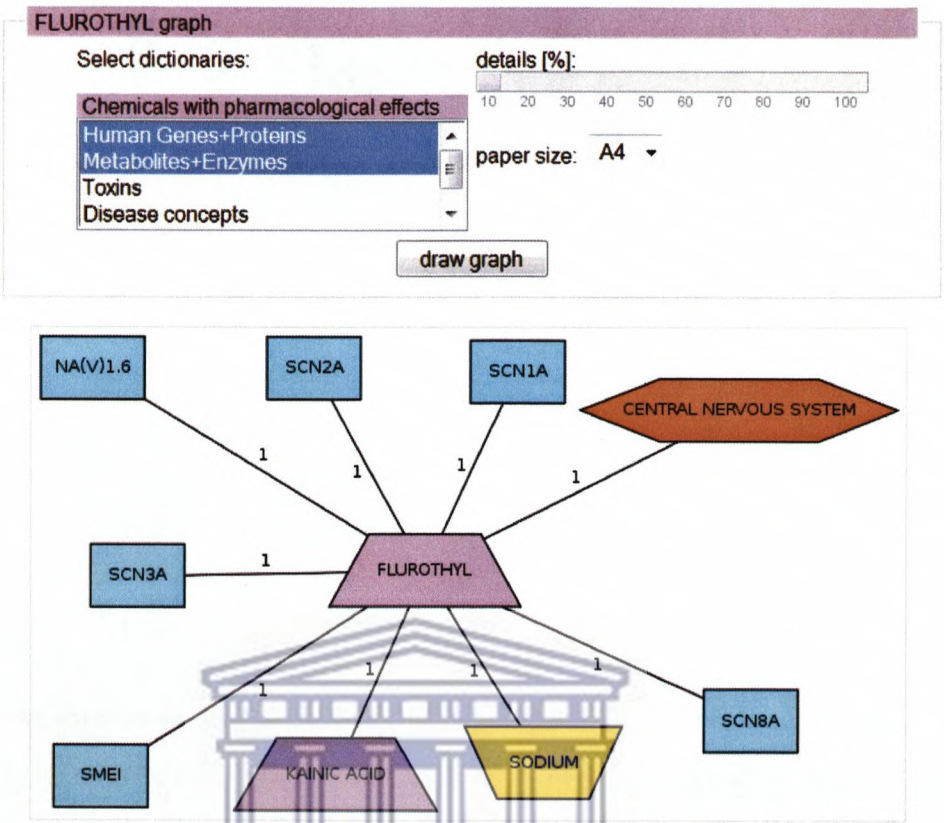


Figure 6.3.3 Interactive network of entities created for Flurothyl. Network is focused on chemicals with pharmacological effects, human genes/proteins, metabolites/enzymes and human anatomy.

By using the hypothesis generator a researcher can go one step further than the information exploration. Figure 6.3.4 shows an example of a hypotheses generator exploring links between Flurothyl, SCN1A gene and number of disease concepts. Various biomedical entities are shaped and color coded. The diagram shows that there is link between Flurothyl and a number of seizures, which is presented as number of abstracts on the far right of the diagram. However, there is nothing in the literature (referred to as 'not confirmed' on the diagram) that would connect Flurothyl with afebrile seizures. The association is of the form that Flurothyl may affect gene SCN1A which may be related to Afebrile seizures. This can potentially represent new knowledge, worth further exploration. This naïve interpretation is just for illustration. Before exploring the hypotheses, the reality of the existing links should be checked by studying the associated PubMed documents. These documents are accessible to a user by clicking the numbers between the terms in the diagram.

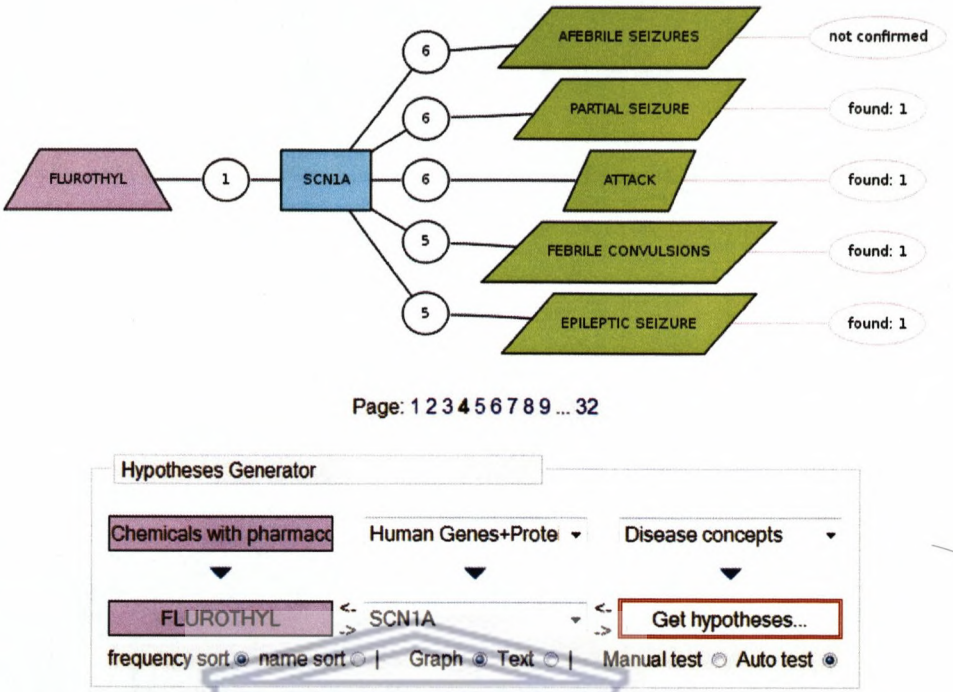


Figure 6.3.4 Hypotheses generated with Flurothyl as the reference.

6.4. Accuracy evaluation

The database effectiveness can be evaluated by using the standard measures of precision (P), recall (R) and F-measure. Due to the database complexity it is not possible to evaluate each concept from each of the dictionaries across all documents. As a reference, the SCN1A gene was selected because of its importance for the generation and propagation of action potentials, primarily in nerves and muscles. This clinically important gene is one of the most studied sodium channel genes and more than 330 mutations have been registered to date (Lossin, 2009).

For evaluation purposes, 131 abstract abstracts in which the SCN1A gene has been explicitly identified, have been manually curated by a team of biologists and Precision, Recall and F measures calculated. The results are summarized in the table below.

Dictionary	Entities		Precision [%]	Recall [%]	F-measure [%]
	Total tp+fp	Correct tp			
Genes and proteins	74	60	81.1	96.8	88.3
Metabolites and enzymes	28	28	100.0	100.0	100
Chemicals with pharmacological effects	7	6	85.7	100.	92.3

Table 6.4.1 Precision, recall and F-measure of entity recognition in documents related to SCN1A gene (Sagar et al., 2008)

Table shows that depending on the dictionary used, the precision and recall are in the range of 81%-100%. In addition, the system has successfully identified most of the entities related to the SCN1A gene with an average F-measure value of 93.5%.

The precision of identifying genes and proteins was 81.1%. Wrongly identified as genes or proteins, were the following entities: SCN, SCN1, Potassium channel, VOLTAGE-GATED K+ CHANNEL, VOLTAGE-GATED POTASSIUM CHANNEL, GTP, Parvalbumin, P17, P21, PL-3, AED, GEF and SMEI. This can be explained due to the following reasons:

- Partially recognized entity: SCN1 was part of SCN1-3A.
- Some entities refer to the family of genes, for example VOLTAGE-GATED POTASSIUM CHANNEL, SCN and CYP.
- Some represent synonyms, aliases or abbreviations of biological entities: SMEI is one of the aliases for SCN1A gene (sodium channel, voltage-gated, type I, alpha) and has been placed in the list of genes and proteins. However, in the original text SMEI is refering to the term 'Severe myoclonic epilepsy of infancy' (PubMed ID: 11359211). In the same way GEF is a synonym for ARHGEF2 gene (rho/rac guanine nucleotide exchange factor (GEF) 2), but it is also an acronym for generalized epilepsy with febrile seizures plus (GEFS+).

There were no incorrectly identified entities from the Metabolites and enzymes dictionary.

Finally, in the list of identified chemicals with pharmacological effects, the only misidentified term is 'lead', which is a metal and metabolite, but also a common English word.

To test sensitivity of the system to the specific area of interest, manually curated abstracts were used to identify how many entities actually relate to sodium channels.

- Genes and proteins group: 18 (30%) out of total 60 entities were found to be either genes coding for various sodium channel proteins, or genes or proteins that could directly affect the functionality of sodium channel proteins.
- Metabolites and enzymes for SCN1A gene, 19 (68%) entities out of total 28 were found to be directly associated with sodium channels.
- Chemicals with pharmacological effects: there were 3 (50%) out of total 6 chemicals that affect the functionality of sodium channels.
- Disease concepts group is too broad to be linked directly to sodium channels.

The following table summarizes those findings.

Gene or protein	Metabolites	Chemical with Pharmacological effects
CALMODULIN	Aspartate	Aspartic acid
NOVA2	Decarboxynucleic acid	Flurothryl
SCN11A	Glutamine	Kainic acid
SCN1B	Glycine	
SCN2A1	Histidine	
SCN2B	Luciferase	
SCN3A	Arginine	
SCN4A	Oxcarbazepine	
SCN5A	Carbamazepine	
SCN7A	Valproate	
SCN8A	Topiramate	
SCN9A	PHT	
SCN1A	Sodium ion	
Nav1.1	Threonine	
Nav1.2	Triphosphatase	
Nav1.3	Lysine	
Nav1.6	Calcium	
PN1	Sodium	
	Phenytoin	

Table 6.4.2 Entities linked with sodium channel biology based on selected abstracts

6.5. Comparison assessment

As additional method of critical assessment, the DDESC database was compared with one of the new biomedical text-mining tools, PolySearch, that specializes in extracting relationships between human diseases, genes, mutations, drugs and metabolites. PolySearch authors reported an F-measure for gene synonym

identification, protein-protein interaction identification and disease gene identification to be 88%, 81% and 79%, respectively (Cheng et al., 2008).

To compare two text-mining systems, the SCN1A gene was used as a reference.

- In the category of metabolites and drugs the PolySearch identified only 6 entities. The DES has returned 28 metabolites and enzymes including drugs.
- In the category of genes and proteins PolySearch has found 159 entries linked to SCN1A gene. As 20 entities were found to be duplicates, the true number of identified entities is 139. The number of genes found was 20, but 5 of those did not belong to this category, so the true number is 14.
- The number of synonyms found was 120.
- DES has found 74 entities linked to the same gene. Out of these 14 were ambiguous and 15 synonyms, so the true number of entities found is 45. 14 out of 45 entities were the same as found by PolySearch. In other words, the DES has found all the entities found by PolySearch in addition to 31 entries in this category.

Tables below summarizes the results of this comparison.

	PolySearch	DES
Total entities identified	139 (after removing 20 duplicates)	74
Ambiguous entries	5	14
Synonyms	120	15
True entries	14	45
Common between PolySearch and DES	14	14
Entries found by DES but not by PolySearch	NA	31
Entries found by PolySearch but not by DES	0	NA

Table 6.5.1 Comparison of results between PolySearch and DDESC for genes and proteins

Gene or protein	DES/PolySearch		Gene or protein	DES/PolySearch	
Feb1	+	-	HIRIP5	+	-
Feb2	+	-	KCNA1	+	-
Feb5	+	-	KCNQ2	+	+
Feb6	+	-	KCNQ3	+	+
ATPASE	+	-	LAFORIN	+	-
CACNA1A	+	-	LGI1	+	-
CACNB4	+	+	MAPT	+	+
CALMODULIN	+	-	MASS1	+	-
CHRNA4	+	+	NHLRC1	+	+
CHRNB	+	-	NIFU	+	-
CHRNB2	+	+	NOVA2	+	+
CLCN2	+	-	P-GLYCOPROTEIN	+	-
COMT	+	-	SCN11A	+	-
CYP2C9	+	-	SCN1B	+	+
CYP3A5	+	-	SCN2A1	+	+
EFHC1	+	-	SCN2B	+	-
FHM2	+	+	SCN3A	+	+
FHM3	+	+	SCN4A	+	+
GABRA1	+	+	SCN5A	+	-
GABRD	+	+	SCN7A	+	-
GABRG2	+	+	SCN8A	+	-
GCH1	+	-	SCN9A	+	-
GM3 SYNTHASE	+	-			

Table 6.5.2 Genes and proteins identified by DES and PolySearch

To conclude this case study, the amount, quality and accuracy of information in the database created by the DES, in addition to its ability to generate association hypotheses, makes DDESC not only a valuable repository of the sodium channel information but also a useful tool for research purposes or drug design.

6.6. A database of text-mined associations for reproductive toxins potentially affecting human fertility

The Dragon Exploration System for Toxicants and Fertility (DESTAF) is another specialized public resource that was created by using DES system. The system focuses on reproductive toxins that are chemicals of biological and non-biological origin that affect reproductive systems resulting in various reproductive disorders. The source of the toxicants is mainly man-made and it can vary from common household cleaning and maintenance products to industrial chemicals (Carson, 2002). The aim of this research conducted by a team of biologists, toxicologists and bioinformaticians was to create an online database that would enable researchers in the field to efficiently explore both known and potentially new information and associations in the area of toxicology and fertility.

6.7. DESTAF in brief

The first step was to obtain documents needed for the text-mining process. PubMed was queried as follows:

(fertil OR reproduc*) human (toxico* OR toxici* OR toxica* OR toxin*)*

The query returned 8690 abstracts that were submitted to DES for processing. Five dictionaries were selected for tagging purposes: Human genes and proteins, Chemical with pharmacological effects, Metabolites and enzymes, Disease concepts and the Human anatomy. The dictionaries were processed in the order mentioned and concepts present in more than one dictionary were only processed once upon their first occurrence. Of the 8690 abstracts analyzed, 8404 (96.7%) contained dictionary entities which were identified in the text.

The DESTAF database uses specially written user interface with all underlying DES reporting modules that remains unchanged. Entities and their relations are provided in a concise, easy to explore format, with links to supporting PubMed abstracts and where available, associated EntrezGene, UniProt (UniProt_Consortium, 2009) and Reactome (Vastrik et al., 2007) pathway and pathway reaction identifiers, as well as the pathway and reaction names. As with other DES based databases, the user is able to search for abstracts, for entities, explore relations, to change perspectives based on preferred dictionaries selected and to explore hypothetical associations through the use of a hypothesis generator.

6.8. Accuracy evaluation

Since this database is complex, the accuracy is measured by using a well known common toxicant. In toxicology this is Dichloro-diphenyl-trichloroethane (DDT), a pesticide with well-characterized toxic reproductive effects. The total number of entities associated from each dictionary with 'DDT' is given in Table 6.8.1, along with a measure of the Precision, Recall and the F-measure.

Dictionary	Entities			Precision [%]	Recall [%]	F-measure [%]
	Total tp+fp	Correct tp	Missed fn			
Human genes and proteins	48	38	3	79.2	92.7	85.4
Chemicals with pharmacological effects	181	181	56	100.0	76.4	86.6
Metabolites and Enzymes	15	14	1	93.3	93.3	93.3
Disease Concepts	86	80	8	93.0	90.9	91.9
Human Anatomy	67	67	3	100.0	95.7	97.8

Table 6.8.1 Precision, Recall and F-measure of entities identified for the toxin/chemical-related entity, DDT

When measuring the precision a number of rules have been applied:

- The entity had to be present in at least one supporting abstract with the same contextual meaning as expected from the dictionary.
- For the Human genes and proteins dictionary, the gene/protein had to be mentioned specifically e.g. 'matrix metalloproteinase 19' and not refer to the less specific gene group or family, e.g. 'metalloproteinase'.
- For the Chemicals with pharmacological effects dictionary, recognition of a chemical group (e.g. 'phthalates' or 'dioxin') within a chemical name was permitted.

The strict rule b) was applied to allow unambiguous mapping of specific human gene/protein entities to Entrez ID. Rule c) permitted partial recognition of chemical names in order to make possible the exploration of associations between potentially toxic chemical groups and other entities.

Recall was measured by reading through all underlying abstracts associated with DDT and identifying potential entities that had not been included in the dictionaries. F-measure was calculated as described in an earlier chapters.

Of the 48 entities from the Human genes and proteins dictionary:

- 3 were chemicals: 'BHC' for 'benzohexachlorene', 'EDB' for 'ethylene dibromide', 'BIS' partial recognition of 'bis-phenol A'.
- 1 not a gene: 'Hormone receptor'.
- 6 referred to pesticides other than DDT in the supporting abstracts: 'alpha-glucosidase' and 'BCL-XL' linked to 'CB-153'; 'DNA polymerase beta',

'RNA polymerise II' and 'p53' to diethylstilbestrol ('DES') and 'AHR' to 'dioxins'.

- 'DDT' was found not to interact with 'progesterone receptor' (PubMed ID: 9073609).

This left 38 entities which were either affected by DDT or involved in its metabolism. These entities produced a non-redundant gene list of 22 genes for which EntrezGene ID's could be unambiguously assigned.

6.9. Results discussion

Most of the concepts recognized fell within Disease concepts (31.4%), Chemicals with pharmacological effects (29.2%) and Human genes and proteins (23.8%) dictionaries (figure below). The low recognition of concepts from the Human anatomy dictionary is likely to do with the fact that many of these are not explicitly mentioned in abstracts that were analyzed.

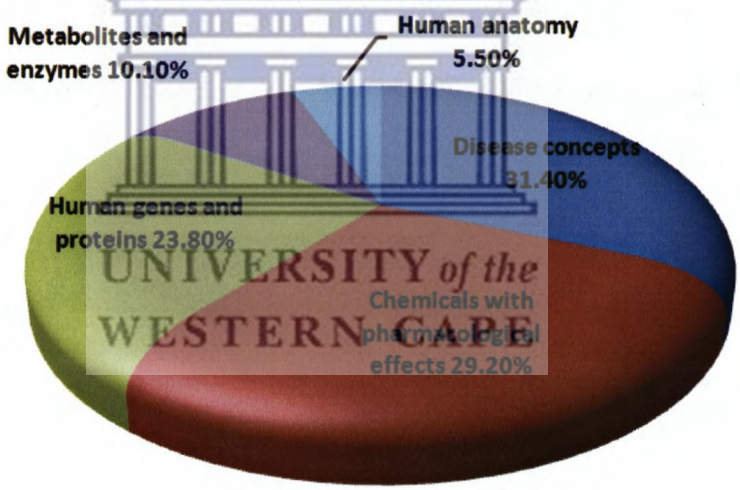


Figure 6.9.1Concepts distribution across the different dictionaries

DDT is found in the 'Chemicals with pharmacological effects' dictionary with high Precision (100.0%) but much lower Recall (76.4%). One of the reasons for this was an excessive number of variants for the chemical names and abbreviations of DDT and its closely related compounds (dichlorodiphenyltrichloroethane; 1,1,1-trichloro-2,2-bis(p-chlorophenyl)ethane; p,p-DDT; o,p-DDT, p, p'-DDT, pp-DDT etc). This case was similar to the metabolite DDE. At the same time, the fact that DDT was used as an abbreviation in the text for so many different variants was also an advantage in bringing together all these entities under a single 'umbrella' entity. DDT is also a symbol

for a human gene, 'D-dopachrome tautomerase', however no reference was found for this gene within the abstract collection.

The Human genes and proteins is a comprehensive dictionary containing some symbols that are no longer in use, but which may have been referred to in older articles. At the same time, these can lead to name confusion with other abbreviations in the text, including those representing diseases, chemicals or even experimental techniques. Also, some genes can share a symbol/alias in their history. For example, GRIP1 is the gene symbol for 'glutamate receptor interacting protein 1', (EntrezGene ID: 23426) as well as an alias for 'nuclear receptor coactivator 2(NCOA2, EntrezGene ID: 10499)' and 'lectin, galactoside-binding, soluble, 12 (LGALS12, EntrezGene ID: 85329)'.

Another issue that can affect precision and recall is that of spelling variants and spelling mistakes which occur within abstracts. For example, 'prostrate' instead of 'prostate' and 'diethyl-stilbestrol' instead of 'diethylstilbestrol'.

In order to map all the genes and proteins first, the Human genes and protein dictionary was the first in the process of abstract analysis. This could potentially lead to wrong identification of entities from other subsequently used dictionaries as being genes or proteins. However, the high F-measure (84.5%) for genes associated with DDT as well as expected link between those genes and reproductive toxicity indicates that the great majority of the genes within DESTAF is correctly identified.

6.10. Ovarian, esophageal and prostate cancer databases

This section describes three different projects but due to their similarity from the methodology point of view they are presented together. Three teams with different principal investigators were working on the ovarian, esophageal and prostate cancer databases. Each of those databases represents the first online repositories of specialized information about each of the cancers. In addition, the research on ovarian cancer and esophageal resulted in two publications (Kaur et al., 2009, Essack et al., 2009).

6.11. Database implementation

Information about these complex diseases are scattered throughout the literature and various databases, making extraction of relevant information a difficult task. The Dragon Database for Exploration of Ovarian Cancer Genes (DDOC), Dragon Database of Genes Implicated in Esophageal Cancer (DDEC) and Dragon Database of Genes Implicated in Prostate Cancer (DDPC) contains information about genes involved in each of all three types of cancers, transcription regulation sequence analysis, and text mining reports that provide insights into relations

between the cancer and other genes, metabolites, pathways and nuclear proteins. Although different in data, all three database share similar user interface and common DES modules.

The work in creating each of these databases involved three steps. The first step was conducted by the research team of biologists and consisted of an extensive literature and databases search for genes implicated in specific cancers. This search effort produced a list of 379 genes for ovarian, 529 for esophageal and 706 genes for prostate cancer. The collected data was used to populate the gene database. Each of the databases consists of 82 tables containing various gene related information.

The second step was the information retrieval from the related literature and the data mining process. The PubMed database was queried for each of the genes by using the following template:

('Gene Symbol' OR 'Gene `Alias' OR 'Gene Alias', etc.) AND mammal AND cancer.

Such queries produced a list of 588,727 abstracts (ovarian cancer) that were submitted to DES to be analyzed and indexed by dictionaries for Nuclear proteins, Pathways, Enzymes and Mammalian genes. The text-mining software created the database of associations that was integrated into each of the databases in the next step.

The third step was software development that will allow the Internet users to access the databases. It is a three-tier web based application that provides access to information stored in the databases combined with access to TRANSPATH database. TRANSPATH is an information system on gene-regulatory pathways that combines manually curated information on signal transduction with tools with tools for visualization and analysis (Choi et al., 2004). User interfaces for all databases are similar and shown in the Figure 6.11.1.

The web based user interface presents results as a lists of tables and as a graphic system of interactive networks. The Figure 6.11.2 shows one of such networks where color-coded biomedical entries are interconnected with weighted links representing frequency of appearance of an entity and its neighbors in abstracts. By clicking on a node, the user gets relevant abstracts containing the selected term, and associated terms (surrounding nodes in the network).

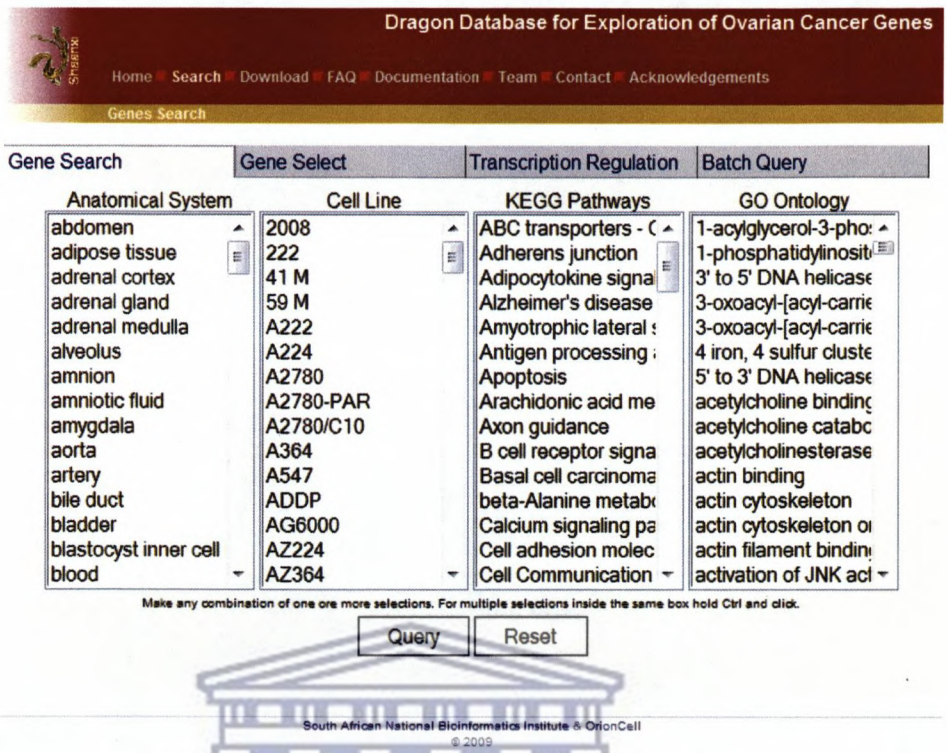


Figure 6.11.1n cancer database 'Search' user interface.

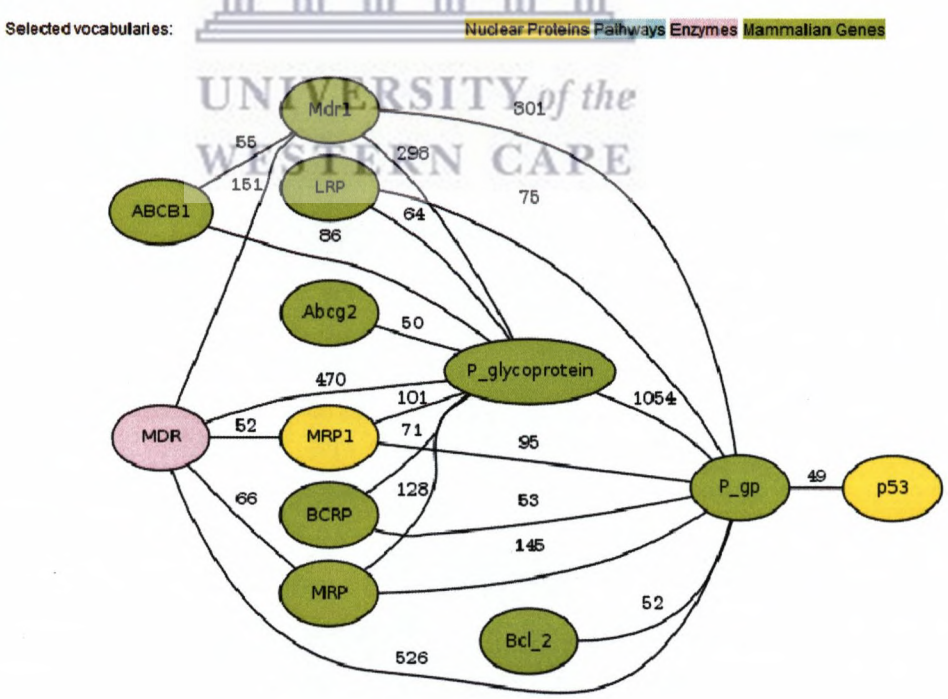


Figure 6.11.2 Interactive network of entities created for ABCB1 gene.

The Figure 6.11.2 is an example of a network of different biological entities which have appeared in literature linked to the ABCB1 gene. It can be seen that, for example, ABCB1 is linked to P-glycoprotein within 86 abstracts. Also, there is a strong link (1054) between P-gp and P-glycoprotein which is not surprising as both are names for the same protein that is (in humans) encoded by the ABCB1 gene. These proteins are also connected to MRP1 (95, 101) as both have been shown to transport same substrates. There is also a strong connection to medium-chain dehydrogenase/reductase (MDR) enzyme (470).

6.12. Chapter Summary

Chapter 6 presented five case studies that show how text mining methodology integrated into the research framework can be used as a useful tool for biologists and bioinformaticians. Not only can resulting databases be created for a specific project, but also presents benefit to a wider biomedical community. High accuracy of such databases, compared with the contemporary text-mining tools can make them valuable and reliable repositories of biomedical information.

Chapter 7. Conclusions

The concluding chapter brings together the theoretical arguments and empirical findings presented in this thesis. It revisits the research questions and highlights the main contributions of this thesis as well as its limitations. Finally, it suggests directions for further research in the fields of automated knowledge extraction.

7.1. Research questions revisited

The following section gives a brief overview of the research questions and answers this study provided.

- *How to develop a method for an automatic knowledge extraction from biomedical literature by using a hybrid approach that combines advantages of the natural language processing, rule based and supervised machine learning techniques?*

  This study described the Concept based Knowledge Discovery (CobKD), a new and original methodology based on integration of multiple text mining techniques with a new, flexible approach for translating user's queries to machine understandable patterns and feature vectors. Rather than being a rigid system based on the fixed templates or database queries, CobKD methodology allows user to define its own concepts for knowledge discovery. This is done by using (although simplified) natural language to describe the type of knowledge the system should look for. Theoretical and some of the implementation aspects of the method are presented in chapters 3.

- *Can the proposed methodology be effectively used for extracting information about a specific relationship between transcription factors and promoters of genes? In this case we were interested in whether a transcription factor does bind to the promoter of a specific gene.*

  Chapter 4 gives a comprehensive evaluation of the CobKD method. It uses an original CobKD based methodology for extraction of information about transcription factor binding to gene's promoter. It is demonstrated that specific methodology can be very efficient in extracting correct information. This methodology is new. It employs specific data preprocessing and machine learning technology that results in a highly efficient recognition system. The results presented at the end of this chapter not only proved the research hypothesis but showed that

the CobKD technique for this given problem, can extract information that is to the level of human curator accuracy (about 5% error).

- *How to develop an integrated biomedical text mining software framework that combines named entity recognition, knowledge extraction and information integration?*

  Based on the methodology, ideas and code behind the Dragon text mining applications (Pan et al., 2004, Bajic et al., 2005, Pan et al., 2006) a new framework enriched with new technologies and methodology presented in this study has been developed. Dragon Exploration System[5] has already proven to be valuable tool for biologists and bioinformaticians as shown through five case studies presented in Chapter 6 and papers arising from this thesis (Sagar et al., 2008, Kaur et al., 2009).

- *How to develop a method for generating potential new knowledge based on relational networks of biomedical entities extracted from disparate biomedical articles?*

  Inspired by Swanson's ideas and his ABC model, an implementation of this model built around relational networks among biomedical entities has been suggested as described in Chapter 5. The model uses frequency of entity co-occurrences to build a hypotheses tree. The hypotheses are generated based on information retrieved from local document collection and automatically verified by using the whole PubMed documents repository. This model was successfully used to consider a possible link, for example between Flurothyl, SCN1A gene and Afebrile seizures as described in Chapter 6.

In summary, each of the research questions has been successfully answered.

7.2. Research contribution and limitations

This research relies on the work of many researchers and methodologies as outlined in the literature review section. It is an attempt to give a contribution to the field of biomedical text mining by the discovering, testing and developing of new methods in knowledge discovery. This thesis presents a novel method for concept based knowledge discovery that is able to achieve precision of a human curator. This is illustrated through specific, but important biological information about which transcription factors bind to promoters of which gene. In an

[5] Proprietary software owned by OrionCell cc.

automated mode of information extraction it was possible to achieve a reasonably high specificity using Levenstein distance. However, with the additional effort of building a relevant machine learning model, the level of accuracy of human curation in extraction of that information was achieved.

The methodology has been implemented in the integrated text mining framework designed to be an original, all-in-one text mining tool that enables researchers to quickly and efficiently analyze the literature in their fields of interest, to reveal relationships and connections among biomedical concepts, extract hidden knowledge and make new literature based, hypotheses. In addition it allows creations of targeted databases for specific organisms, diseases, genes, chemicals and generally, any other biomedical topic.

However, certain limitations also have to be considered. The methodology of knowledge extraction relies on supervised machine learning that include costly and lengthy processes of human curation of data needed to create the models. Also, extracting more straightforward relationships seems to be easier, while indirect and relationships hidden in segmented phrases seems to be more difficult.

7.3. Future work

One of the directions for future work is solving the problem of extraction of relationships hidden in complicated, fragmented sentences with indirect references, or over more than one sentence. To address this problem requires using more sophisticated NLP methodologies.

The expanding PubMed database with increasing information content linked to the publication, shows that work in this area is evolving and requires continued attention. The existing and potential feature space of XML and the standard generalized markup language (SGML) that includes a large array of information types such as references, investigators, headings, formatting, comments, dates, links… have just begun to be explored. Expanding this space even further to full text and multimedia content, using concepts in addition to words as basic units, will require additional analytical methods and promises exciting times ahead.

Appendix A: Entity Dictionaries

Dictionary	Terms
Biological Process	1113
Cellular Component	276
Chemicals with pharmacological effects	47478
Disease concepts	97016
HIV	270
HLA alleles	3477
Human anatomy	1290
Human Genes+Proteins	289240
Mammalian Genes	142930
Metabolites+Enzymes	70547
miRNA Human	2833
Mode of action	427
Molecular Function	1384
Nuclear Proteins	21884
Pathways	954
TB-HIV drugs	215
Toxins	8331
Transcription Factors	18222

Appendix B: Entities relations keywords

Direct Interaction [interact]	Indirect effect [indirect]	Modifications [modification]	Other [other]
associate	activate	acetylate	oxidization
associated	activated	acetylated	oxidize
associates	activates	acetylates	oxidized
associating	activating	acetylating	oxidizes
association	activation	acetylation	oxidizing
attach	cleavage	carbamoylated	ligand
attached	cleave	carbamoylation	peroxidizing
attaches	cleaved	carboxylate	recognize
attaching	cleaves	carboxylates	recognized
attachment	cleaving	carboxylation	recognizes
attack	down-regulate	deacetylate	recognizing
attacked	down-regulates	deacetylated	acceptor
attacking	down-regulating	deacetylates	
attacks	down-regulation	deacetylating	
bind	downregulate	deacetylation	
binding	downregulates	deaminated	
binds	downregulating	deamination	
bound	downregulation	decarboxylated	
co-immunoprecipitate	hydrolyse	decarboxylates	
co-immunoprecipitated	hydrolysed	decarboxylation	
co-immunoprecipitates	hydrolyses	dehydrated	
co-immunoprecipitating	hydrolysing	dehydrogenated	
co-immunoprecipitation	hydrolysis	dehydrogenation	
co-immunoprecipitations	inactivate	dephosphorylate	
complex	inactivated	dephosphorylated	
complexation	inactivates	dephosphorylates	
complexed	inactivating	dephosphorylating	
complexes	inactivation	dephosphorylation	
complexing	inhibit	formylated	
conjugate	inhibited	glycosylated	
conjugated	inhibiting	glycosylates	
conjugates	inhibition	glycosylation	
conjugating	inhibitor	isomerization	
conjugation	inhibits	isomerize	
contact	recruit	isomerized	
contacted	recruited	isomerizes	

contacting	recruiting	isomerizing	
contacts	recruits	methylate	
dock	regulate	methylated	
docked	regulated	methylates	
docking	regulates	methylating	
docks	regulating	methylation	
heterodimer	regulation	phosphorylate	
heterodimerization	up-regulate	phosphorylated	
heterodimerize	up-regulated	phosphorylates	
heterodimerized	up-regulates	phosphorylating	
heterodimerizes	up-regulating	phosphorylation	
heterodimerizing	up-regulation	transamination	
heterodimers	upregulate	ubiquitinate	
homodimer	upregulated	ubiquitinated	
homodimerization	upregulates	ubiquitinates	
homodimerize	upregulating	ubiquitinating	
homodimerized	upregulation	ubiquitination	
homodimerizes			
homodimers			
interact			
interacted			
interacting			
interaction			
interacts			
ligate			
ligated			
ligates			
ligating			
ligation			
pair			
paired			
pairing			
pairs			
tether			
tethered			
tethering			
tethers			

Appendix C: NLM stop-words

	U.S. National Library of Medicine Stop-words
A	a, about, again, all, almost, also, although, always, among, an, and, another, any, are, as, at
B	be, because, been, before, being, between, both, but, by
C	can, could
D	did, do, does, done, due, during
E	each, either, enough, especially, etc
F	for, found, from, further
H	had, has, have, having, here, how, however
I	i, if, in, into, is, it, its, itself
J	just
K	kg, km
M	made, mainly, make, may, mg, might, ml, mm, most, mostly, must
N	nearly, neither, no, nor
O	obtained, of, often, on, our, overall
P	perhaps, pmid
Q	quite
R	rather, really, regarding
S	seem, seen, several, should, show, showed, shown, shows, significantly, since, so, some, such
T	than, that, the, their, theirs, them, then, there, therefore, these, they, this, those, through, thus, to
U	upon, use, used, using
V	various, very
W	was, we, were, what, when, which, while, with, within, without, would

Table of figures

List of Tables

References

Ambler, S. W. (2002) Agile modeling : effective practices for eXtreme programming and the unified process. New York, J. Wiley.

Andrade, M. A. & Valencia, A. (1998) Automatic extraction of keywords from scientific text: application to the knowledge domain of protein families. *Bioinformatics,* 14, 600-7.

Atkinson, J. & Rivas, A. (2008) Discovering novel causal patterns from biomedical natural-language texts using Bayesian nets. *IEEE Trans Inf Technol Biomed,* 12, 714-22.

Bach, J. (2003) Exploratory Testing Explained. v.1.3 ed.

Bairoch, A. & Apweiler, R. (1996) The SWISS-PROT protein sequence data bank and its new supplement TREMBL. *Nucleic Acids Res,* 24, 21-5.

Bajic, V. B., Veronika, M., Veladandi, P. S., Meka, A., Heng, M. W., Rajaraman, K., Pan, H. & Swarup, S. (2005) Dragon Plant Biology Explorer. A Text-Mining Tool for Integrating Associations between Genetic and Biochemical Entities with Genome Annotation and Biochemical Terms Lists. *Plant Physiol,* 138, 1914-1925.

Barrell, D., Dimmer, E., Huntley, R. P., Binns, D., O'Donovan, C. & Apweiler, R. (2009) The GOA database in 2009--an integrated Gene Ontology Annotation resource. *Nucleic Acids Res,* 37, D396-403.

Baxevanis, A. D. (2008) Searching NCBI databases using Entrez. *Curr Protoc Bioinformatics,* Chapter 1, Unit 1 3.

Beck, K. & Andres, C. (2005) *Extreme programming explained : embrace change,* Boston, MA, Addison-Wesley.

Beck, K., Beedle, M., Bennekum, A. v., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J. & Thomas, D. Manifesto for Agile Software Development.

Bekhuis, T. (2006) Conceptual biology, hypothesis discovery, and text mining: Swanson's legacy. *Biomed Digit Libr,* 3, 2.

Blagosklonny, M. V. & Pardee, A. B. (2002) Conceptual biology: unearthing the gems. *Nature,* 416, 373.

Blake, J. B. (1986) From Surgeon General's bookshelf to National Library of Medicine: a brief history. *Bull Med Libr Assoc,* 74, 318-24.

Blaschke, C., Andrade, M. A., Ouzounis, C. & Valencia, A. (1999) Automatic extraction of biological information from scientific text: protein-protein interactions. *Proc Int Conf Intell Syst Mol Biol,* 60-7.

Breiman, L. (2001) Random Forests. *Machine Learning,* V45, 5-32.

Breitkreutz, B. J., Stark, C., Reguly, T., Boucher, L., Breitkreutz, A., Livstone, M., Oughtred, R., Lackner, D. H., Bahler, J., Wood, V., Dolinski, K. & Tyers, M. (2008) The BioGRID Interaction Database: 2008 update. *Nucleic Acids Res,* 36, D637-40.

Bunescu, R., Mooney, R., Weiss, Y., Scholkopf, B. & Platt, J. (2006) Subsequence Kernels for Relation Extraction. *Advances in Neural Information Processing Systems 18.* MIT Press.

Bush, V. (1996) As we may think. *interactions,* 3, 35-46.

Carson, R. (2002) *Silent spring,* Boston, Houghton Mifflin.

Chawla, N., Bowyer, K. & Kegelmeyer, P. (2002) SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research,* 16, 321-357.

Chen, H. & Sharp, B. M. (2004) Content-rich biological network constructed by mining PubMed abstracts. *BMC Bioinformatics,* 5, 147.

Chen, Z. (1993) Let documents talk to each other: A computer model for connection of short documents. *Journal of Documentation,* 49, 44-54.

Cheng, D., Knox, C., Young, N., Stothard, P., Damaraju, S. & Wishart, D. S. (2008) PolySearch: a web-based text mining system for extracting relationships between human diseases, genes, mutations, drugs and metabolites. *Nucleic Acids Res,* 36, W399-405.

Choi, C., Krull, M., Kel, A., Kel-Margoulis, O., Pistor, S., Potapov, A., Voss, N. & Wingender, E. (2004) TRANSPATH-A High Quality Database Focused on Signal Transduction. *Comp Funct Genomics,* 5, 163-8.

Chowdhary, R., Zhang, J. & Liu, J. S. (2009) Bayesian Inference of Protein-protein Interactions from Biological Literature. *Bioinformatics.*

Cleary, J. & Trigg, L. (1995) K*: An instance-based learner using an entropic distance measure. *In Proceedings of the 12th International Conference on Machine Learning.*

Cortes, C. & Vapnik, V. (1995) Support-vector networks. *Machine Learning,* 20, 273-297.

Davis, J. & Goadrich, M. (2006) The relationship between Precision-Recall and ROC curves. *ICML '06: Proceedings of the 23rd international conference on Machine learning.* ACM.

Dee, C. R. (2007) The development of the Medical Literature Analysis and Retrieval System (MEDLARS). *J Med Libr Assoc,* 95, 416-25.

Ding, J., Berleant, D., Nettleton, D. & Wurtele, E. (2002) Mining MEDLINE: abstracts, sentences, or phrases? *Pac Symp Biocomput,* 326-37.

Edhlund, B. (2005) *Basic Principles of Pubmed,* Lulu Press.

Elnitski, L., Jin, V. X., Farnham, P. J. & Jones, S. J. (2006) Locating mammalian transcription factor binding sites: a survey of computational and experimental techniques. *Genome Res,* 16, 1455-64.

Essack, M., Radovanovic, A., Schaefer, U., Schmeier, S., Seshadri, S., Christoffels, A., Kaur, M. & Bajic, V. (2009) DDEC: Dragon database of genes implicated in esophageal cancer. *BMC Cancer,* 9, 219.

Friedman, C., Alderson, P. O., Austin, J. H., Cimino, J. J. & Johnson, S. B. (1994) A general natural-language text processor for clinical radiology. *J Am Med Inform Assoc,* 1, 161-74.

Friedman, C., Kra, P., Yu, H., Krauthammer, M. & Rzhetsky, A. (2001) GENIES: a natural-language processing system for the extraction of molecular pathways from journal articles. *Bioinformatics,* 17 Suppl 1, S74-82.

Fukuda, K., Tamura, A., Tsunoda, T. & Takagi, T. (1998) Toward information extraction: identifying protein names from biological papers. *Pac Symp Biocomput,* 707-18.

Gaizauskas, R., Demetriou, G. & Humphreys, K. (2000) Term Recognition and Classification in Biological Science Journal Articles. *In Proc. of the*

Computional Terminology for Medical and Biological Applications Workshop of the 2 nd International Conference on NLP.

Glenisson, P., Coessens, B., Van Vooren, S., Mathys, J., Moreau, Y. & De Moor, B. (2004) TXTGate: profiling gene groups with text-based information. *Genome Biol,* 5, R43.

Govindarajan, M. & Chandrasekaran, R. M. (2007) Classifier Based Text Mining for NeuralNetwork. *Proceedings of World Academy of Science, Engineering and Technology.*

Guyon, I. & Elisseeff, A. (2003) An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research,* 3, 1157-1182.

Hansen, M. A. (1998) Free online access to medical information: MEDLINE Web interfaces. *Health Care Internet,* 2, 29-43.

Harris, M. A., Clark, J., Ireland, A., Lomax, J., Ashburner, M., Foulger, R., Eilbeck, K., Lewis, S., Marshall, B., Mungall, C., Richter, J., Rubin, G. M., Blake, J. A., Bult, C., Dolan, M., Drabkin, H., Eppig, J. T., Hill, D. P., Ni, L., Ringwald, M., Balakrishnan, R., Cherry, J. M., Christie, K. R., Costanzo, M. C., Dwight, S. S., Engel, S., Fisk, D. G., Hirschman, J. E., Hong, E. L., Nash, R. S., Sethuraman, A., Theesfeld, C. L., Botstein, D., Dolinski, K., Feierbach, B., Berardini, T., Mundodi, S., Rhee, S. Y., Apweiler, R., Barrell, D., Camon, E., Dimmer, E., Lee, V., Chisholm, R., Gaudet, P., Kibbe, W., Kishore, R., Schwarz, E. M., Sternberg, P., Gwinn, M., Hannick, L., Wortman, J., Berriman, M., Wood, V., de la Cruz, N., Tonellato, P., Jaiswal, P., Seigfried, T. & White, R. (2004) The Gene Ontology (GO) database and informatics resource. *Nucleic Acids Res,* 32, D258-61.

Hartl, D. L. & Jones, E. W. (2009) Genetics : analysis of genes and genomes. 7th ed. Sudbury, Mass., Jones and Bartlett.

Hatzivassiloglou, V., Duboue, P. A. & Rzhetsky, A. (2001) Disambiguating proteins, genes, and RNA in text: a machine learning approach. *Bioinformatics,* 17 Suppl 1, S97-106.

He, M., Wang, Y. & Li, W. (2009) PPI finder: a mining tool for human protein-protein interactions. *PLoS ONE,* 4, e4554.

Hearst, M. (1999) Untangling text data mining. *Proceedings of ACL'99: the 37th Annual Meeting of the Association for Computational Linguistics.*

Hellman, L. M. & Fried, M. G. (2007) Electrophoretic mobility shift assay (EMSA) for detecting protein-nucleic acid interactions. *Nat Protoc,* 2, 1849-61.

Hersh, W. (2005) Evaluation of biomedical text-mining systems: lessons learned from information retrieval. *Brief Bioinform,* 6, 344-356.

Hobohm, U. & Sander, C. (1994) Enlarged representative set of protein structures. *Protein Sci,* 3, 522-4.

Hristovski, D., Friedman, C., Rindflesch, T. C. & Peterlin, B. (2006) Exploiting semantic relations for literature-based discovery. *AMIA Annu Symp Proc,* 349-53.

Huang, M., Zhu, X., Hao, Y., Payan, D. G., Qu, K. & Li, M. (2004) Discovering patterns to extract protein-protein interactions from full texts. *Bioinformatics,* 20, 3604-12.

Irina, R. (2001) An empirical study of the naive Bayes classifier. *IJCAI-01 workshop on "Empirical Methods in AI".*

Jensen, L. J., Saric, J. & Bork, P. (2006) Literature mining for the biologist: from information retrieval to biological discovery. *Nat Rev Genet,* 7, 119-29.

Jenssen, T. K., Laegreid, A., Komorowski, J. & Hovig, E. (2001) A literature network of human genes for high-throughput analysis of gene expression. *Nat Genet,* 28, 21-8.

Joachims, T. (1998) Text categorization with Support Vector Machines: Learning with many relevant features. *Machine Learning: ECML-98.*

Kandel, E. R., Schwartz, J. H. & Jessell, T. M. (2000) Principles of neural science. 4th ed. New York, McGraw-Hill, Health Professions Division.

Kane, D. W., Hohman, M. M., Cerami, E. G., McCormick, M. W., Kuhlmman, K. F. & Byrd, J. A. (2006) Agile methods in biomedical software development: a multi-site experience report. *BMC Bioinformatics,* 7, 273.

Kanehisa, M., Goto, S., Hattori, M., Aoki-Kinoshita, K. F., Itoh, M., Kawashima, S., Katayama, T., Araki, M. & Hirakawa, M. (2006) From genomics to chemical genomics: new developments in KEGG. *Nucleic Acids Res,* 34.

Kaur, M., Radovanovic, A., Essack, M., Schaefer, U., Maqungo, M., Kibler, T., Schmeier, S., Christoffels, A., Narasimhan, K., Choolani, M. & Bajic, V. B. (2009) Database for exploration of functional context of genes implicated in ovarian cancer. *Nucleic Acids Res,* 37, D820-3.

Kim, H., Park, H. & Drake, B. L. (2007) Extracting unrecognized gene relationships from the biomedical literature via matrix factorizations. *BMC Bioinformatics,* 8 Suppl 9, S6.

Kim, S., Shin, S. Y., Lee, I. H., Kim, S. J., Sriram, R. & Zhang, B. T. (2008) PIE: an online prediction system for protein-protein interactions from text. *Nucleic Acids Res,* 36, W411-5.

Kotsiantis, S. B. (2007) Supervised Machine Learning: A Review of Classification Techniques. *Informatica.*

Koussounadis, A., Redfern, O. C. & Jones, D. T. (2009) Improving classification in protein structure databases using text mining. *BMC Bioinformatics,* 10, 129.

Kramer-Hammerle, S., Hahn, A., Brack-Werner, R. & Werner, T. (2005) Elucidating effects of long-term expression of HIV-1 Nef on astrocytes by microarray, promoter, and literature analyses. *Gene,* 358, 31-8.

Krauthammer, M. & Nenadic, G. (2004) Term identification in the biomedical literature. *J Biomed Inform,* 37, 512-26.

Lafferty, J., McCallum, A. & Pereira, F. (2001) Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. *Proc. 18th International Conf. on Machine Learning.* Morgan Kaufmann, San Francisco, CA.

Larranaga, P., Calvo, B., Santana, R., Bielza, C., Galdiano, J., Inza, I., Lozano, J. A., Armananzas, R., Santafe, G., Perez, A. & Robles, V. (2006) Machine learning in bioinformatics. *Brief Bioinform,* 7, 86-112.

Leblanc, B. t. & Moss, T. (2009) *DNA-Protein Interactions: Principles and Protocols*, Springer Protocols, Humana Press.

Leroy, G. & Chen, H. (2002) Filling preposition-based templates to capture information from medical abstracts. *Pac Symp Biocomput,* 350-61.

Levenshtein, V. (1966) Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady.*

Lin, Y., Li, W., Chen, K. & Liu, Y. (2007) A document clustering and ranking system for exploring MEDLINE citations. *J Am Med Inform Assoc,* 14, 651-61.

Liu, H., Hu, Z. Z., Torii, M., Wu, C. & Friedman, C. (2006) Quantitative assessment of dictionary-based protein named entity tagging. *J Am Med Inform Assoc,* 13, 497-507.

Lossin, C. (2009) A catalog of SCN1A variants. *Brain Dev,* 31, 114-30.

Luhn, H. P. (1958) The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development,* 2.

Maglott, D., Ostell, J., Pruitt, K. D. & Tatusova, T. (2007) Entrez Gene: gene-centered information at NCBI. *Nucleic Acids Res,* 35.

Makoto, M., Satre, R., Miyao, Y., Ohta, T. & Tsujii, J. i. (2008) Combining Multiple Layers of Syntactic Information for Protein-Protein Interaction Extraction. *Proceedings of the Third International Symposium on Semantic Mining in Biomedicine (SMBM 2008).* Turku, Finland.

Malik, R., Franke, L. & Siebes, A. (2006) Combination of text-mining algorithms increases the performance. *Bioinformatics,* 22, 2151-7.

Marc Weeber, H. K., Lolkje T.W. de Jong-van den Berg, Rein Vos, (2001) Using concepts in literature-based discovery: Simulating Swanson's Raynaud-fish oil and migraine-magnesium discoveries. *Journal of the American Society for Information Science and Technology,* 52, 548-557.

Marneffe, M., Maccartney, B. & Manning, C. (2006) Generating Typed Dependency Parses from Phrase Structure Parses. *Proceedings of LREC-06.*

McCallum, A., Freitag, D. & Pereira, F. (2000) Maximum Entropy Markov Models for Information Extraction and Segmentation. *Proc. 17th International Conf. on Machine Learning.* Morgan Kaufmann, San Francisco, CA.

McCallum, A. & Li, W. (2003) Early results for named entity recognition with conditional random fields.

Mika, S. & Rost, B. (2004) Protein names precisely peeled off free text. *Bioinformatics,* 20 Suppl 1, i241-7.

Miles, W. D., & National Library of Medicine (U.S.) (1982) *A history of the National Library of Medicine : the nation's treasury of medical knowledge,* Bethesda, Md.

Washington, D.C., U.S. Dept. of Health and Human Services, Public Health Service, National Institutes of Health

For sale by the Supt. of Docs., U.S. G.P.O.

Miller, N., Lacroix, E. M. & Backus, J. E. (2000) MEDLINEplus: building and maintaining the National Library of Medicine's consumer health Web service. *Bull Med Libr Assoc,* 88, 11-7.

Mitchell, T. (1997) Decision Tree Learning. *Machine Learning.* The McGraw-Hill Companies.

Miyao, Y., S\aetre, R., Sagae, K., Matsuzaki, T. & Tsujii, J. i. (2008) Task-oriented Evaluation of Syntactic Parsers and Their Representations. *Proceedings of ACL-08: HLT.* Association for Computational Linguistics.

Mueller, L. A., Zhang, P. & Rhee, S. Y. (2003) AraCyc: a biochemical pathway database for Arabidopsis. *Plant Physiol,* 132, 453-60.

Muller, H. M., Kenny, E. E. & Sternberg, P. W. (2004) Textpresso: an ontology-based information retrieval and extraction system for biological literature. *PLoS Biol,* 2, e309.

Navarro, G. (2001) A guided tour to approximate string matching. *ACM Computing Surveys,* 33, 31-88.

Neveol, A., Shooshan, S. E., Mork, J. G. & Aronson, A. R. (2007) Fine-grained indexing of the biomedical literature: MeSH subheading attachment for a MEDLINE indexing tool. *AMIA Annu Symp Proc,* 553-7.

Ono, T., Hishigaki, H., Tanigami, A. & Takagi, T. (2001) Automated extraction of information on protein-protein interactions from the biological literature. *Bioinformatics,* 17, 155-61.

Pan, H., Zuo, L., Choudhary, V., Zhang, Z., Leow, S. H., Chong, F. T., Huang, Y., Ong, V. W., Mohanty, B., Tan, S. L., Krishnan, S. P. & Bajic, V. B. (2004) Dragon TF Association Miner: a system for exploring transcription factor associations through text-mining. *Nucleic Acids Res,* 32, W230-4.

Pan, H., Zuo, L., Kanagasabai, R., Zhang, Z., Choudhary, V., Mohanty, B., Tan, S., Krishnan, S., Veladandi, P., Meka, A., Choy, W., Swarup, S. & Bajic, V. (2006) Extracting Information for Meaningful Function Inference through Text-Mining. *Discovering Biomolecular Mechanisms with Computational Biology.*

Park, J. C., Kim, H. S. & Kim, J. J. (2001) Bidirectional incremental parsing for automatic pathway identification with combinatory categorial grammar. *Pac Symp Biocomput,* 396-407.

Parri, R. & Crunelli, V. (2003) An astrocyte bridge from synapse to blood flow. *Nat Neurosci,* 6, 5-6.

Pharkya, P., Nikolaev, E. V. & Maranas, C. D. (2003) Review of the BRENDA Database. *Metab Eng,* 5, 71-3.

Pitt-Francis, J., Bernabeu, M. O., Cooper, J., Garny, A., Momtahan, L., Osborne, J., Pathmanathan, P., Rodriguez, B., Whiteley, J. P. & Gavaghan, D. J. (2008) Chaste: using agile programming techniques to develop computational biology software. *Philos Transact A Math Phys Eng Sci,* 366, 3111-36.

Quinlan, R. (1993) *C4.5: programs for machine learning*, Morgan Kaufmann Publishers Inc.

Rabiner, L. R. (1989) A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE,* 77, 257-286.

Raju, S. B., Chandrasekhar, P. V. S. & Prasad, M. K. (2002) Application of Multilayer Perceptron Network for Tagging Parts-of-Speech. *Proceedings of the Language Engineering Conference (LEC'02).* IEEE Computer Society.

Rensink, W. A. & Buell, C. R. (2004) Arabidopsis to rice. Applying knowledge from a weed to enhance our understanding of a crop species. *Plant Physiol,* 135, 622-9.

Robert Gaizauskas, G. D. K. H. (2000) Term Recognition and Classification in Biological Science Journal Articles. unknown.

Roberts, R. J. (2001) PubMed Central: The GenBank of the published literature. *Proc Natl Acad Sci U S A,* 98, 381-2.

Rogers, A., Antoshechkin, I., Bieri, T., Blasiar, D., Bastiani, C., Canaran, P., Chan, J., Chen, W. J., Davis, P., Fernandes, J., Fiedler, T. J., Han, M., Harris, T. W., Kishore, R., Lee, R., McKay, S., Muller, H. M., Nakamura, C., Ozersky, P., Petcherski, A., Schindelman, G., Schwarz, E. M., Spooner, W., Tuli, M. A., Van Auken, K., Wang, D., Wang, X., Williams, G., Yook, K., Durbin, R., Stein, L. D., Spieth, J. & Sternberg, P. W. (2008) WormBase 2007. *Nucleic Acids Res,* 36, D612-7.

Rumelhart, D. E., Hinton, G. E. & Williams, R. J. (1986) Learning internal representations by error propagation. 318-362.

Sagar, S., Kaur, M., Dawe, A., Seshadri, S. V., Christoffels, A., Schaefer, U., Radovanovic, A. & Bajic, V. B. (2008) DDESC: Dragon database for exploration of sodium channels in human. *BMC Genomics,* 9, 622.

Salton, G. & Buckley, C. (1987) Term Weighting Approaches in Automatic Text Retrieval. Cornell University.

Saric, J., Jensen, L. J., Ouzounova, R., Rojas, I. & Bork, P. (2006) Extraction of regulatory gene/protein networks from Medline. *Bioinformatics,* 22, 645-50.

Sayers, E. W., Barrett, T., Benson, D. A., Bryant, S. H., Canese, K., Chetvernin, V., Church, D. M., DiCuccio, M., Edgar, R., Federhen, S., Feolo, M., Geer, L. Y., Helmberg, W., Kapustin, Y., Landsman, D., Lipman, D. J., Madden, T. L., Maglott, D. R., Miller, V., Mizrachi, I., Ostell, J., Pruitt, K. D., Schuler, G. D., Sequeira, E., Sherry, S. T., Shumway, M., Sirotkin, K., Souvorov, A., Starchenko, G., Tatusova, T. A., Wagner, L., Yaschenko, E. & Ye, J. (2009) Database resources of the National Center for Biotechnology Information. *Nucleic Acids Res,* 37, D5-15.

Schneider, R., de Daruvar, A. & Sander, C. (1997) The HSSP database of protein structure-sequence alignments. *Nucleic Acids Res,* 25, 226-30.

Shi, L. & Campagne, F. (2005) Building a protein name dictionary from full text: a machine learning term extraction approach. *BMC Bioinformatics,* 6, 88.

Skusa, A., Ruegg, A. & Kohler, J. (2005) Extraction of biological interaction networks from scientific literature. *Brief Bioinform,* 6, 263-76.

Smalheiser, N. R., Torvik, V. I. & Zhou, W. (2009a) Arrowsmith two-node search interface: a tutorial on finding meaningful links between two disparate sets of articles in MEDLINE. *Comput Methods Programs Biomed,* 94, 190-7.

Smalheiser, N. R., Torvik, V. I. & Zhou, W. (2009b) Arrowsmith two-node search interface: A tutorial on finding meaningful links between two disparate sets of articles in MEDLINE. *Computer Methods and Programs in Biomedicine,* 94, 190-197.

Smith, A. M. (2004) An examination of PubMed's ability to disambiguate subject queries and journal title queries. *J Med Libr Assoc,* 92, 97-100.

Sohn, S., Kim, W., Comeau, D. C. & Wilbur, W. J. (2008) Optimal training sets for Bayesian prediction of MeSH assignment. *J Am Med Inform Assoc,* 15, 546-53.

Spackman, K. A. (1989) Signal detection theory: Valuable tools for evaluating inductive learning. *Proceedings of the Sixth International Workshop on Machine Learning.* Morgan Kaufman.

Srinivasan, P. (2003) Text mining: Generating hypotheses from MEDLINE. *Journal of the American Society for Information Science and Technology,* 55, 396-413.

Stapley, B. J. & Benoit, G. (2000) Biobibliometrics: information retrieval and visualization from co-occurrences of gene names in Medline abstracts. *Pac Symp Biocomput,* 529-40.

Stegmann, J. & Grohmann, G. (2003) Hypothesis generation guided by co-word clustering. *Scientometrics,* 56, 111-135.

Swanson, D. R. (1986) Fish oil, Raynaud's syndrome, and undiscovered public knowledge. *Perspect Biol Med,* 30, 7-18.

Swanson, D. R. (1990) Medical literature as a potential source of new knowledge. *Bull Med Libr Assoc,* 78, 29-37.

Swanson, D. R. (2008) Running, esophageal acid reflux, and atrial fibrillation: a chain of events linked by evidence from separate medical literatures. *Med Hypotheses,* 71, 178-85.

Swanson, D. R. & Smalheiser, N. R. (1996) Undiscovered Public Knowledge: A Ten-Year Update. *KDD.*

Swanson, D. R. & Smalheiser, N. R. (1997) An interactive system for finding complementary literatures: a stimulus to scientific discovery. *Artificial Intelligence,* 91, 183-203.

Tanabe, L., Scherf, U., Smith, L. H., Lee, J. K., Hunter, L. & Weinstein, J. N. (1999) MedMiner: an Internet text-mining tool for biomedical information, with application to gene expression profiling. *BioTechniques,* 27, 1210-4, 1216-7.

Temkin, J. M. & Gilder, M. R. (2003) Extraction of protein interaction information from unstructured text using a context-free grammar. *Bioinformatics,* 19, 2046-53.

Thomas, J., Milward, D., Ouzounis, C., Pulman, S. & Carroll, M. (2000) Automatic extraction of protein interactions from scientific abstracts. *Pac Symp Biocomput,* 541-52.

Torvik, V. I. & Smalheiser, N. R. (2007) A quantitative model for linking two disparate sets of articles in MEDLINE. *Bioinformatics,* 23, 1658-65.

UniProt_Consortium (2009) The Universal Protein Resource (UniProt) 2009. *Nucleic Acids Res,* 37, D169-74.

Van Landeghem, S., Saeys, Y., De Baets, B. & Van de Peer, Y. (2008) Extracting Protein-Protein Interactions from Text using Rich Feature Vectors and Feature Selection. IN SALAKOSKI, T., SCHUHMANN, D. & PYYSALO, S. (Eds.) *Proceedings of the Third International Symposium on Semantic Mining in Biomedicine (SMBM 2008), Turku, Finland.* Turku Centre for Computer Science (TUCS).

Van Rijsbergen, C. J. (1979) *Information retrieval,* London ; Boston, Butterworths.

Vastrik, I., D'Eustachio, P., Schmidt, E., Gopinath, G., Croft, D., de Bono, B., Gillespie, M., Jassal, B., Lewis, S., Matthews, L., Wu, G., Birney, E. & Stein, L. (2007) Reactome: a knowledge base of biologic pathways and processes. *Genome Biol,* 8, R39.

Weeber, M., Klein, H., Aronson, A. R., Mork, J. G., de Jong-van den Berg, L. T. & Vos, R. (2000) Text-based discovery in biomedicine: the architecture of the DAD-system. *Proc AMIA Symp,* 903-7.

Witten, I. & Frank, E. (2005) *Data Mining: Practical Machine Learning Tools and Techniques*, Morgan Kaufmann.

Wood, W. A. & Kleb, W. L. (2003) Exploring XP for scientific research. *Software, IEEE,* 20, 30-36.

Yakushiji, A., Tateisi, Y., Miyao, Y. & Tsujii, J. (2001) Event extraction from biomedical papers using a full parser. *Pac Symp Biocomput,* 408-19.

Yang, Z., Lin, H. & Li, Y. (2008) Exploiting the performance of dictionary-based bio-entity name recognition in biomedical literature. *Computational Biology and Chemistry,* 32, 287-291.

Yetisgen-Yildiz, M. & Pratt, W. (2006) Using statistical and knowledge-based approaches for literature-based discovery. *J Biomed Inform,* 39, 600-11.

Yu, L. & Liu, H. (2004) Efficient Feature Selection via Analysis of Relevance and Redundancy. *J. Mach. Learn. Res.,* 5, 1205-1224.

Zhou, D. & He, Y. (2008) Extracting interactions between proteins from the literature. *J Biomed Inform,* 41, 393-407.

Zhou, G. & Su, J. (2004) Exploring Deep Knowledge Resources in Biomedical Name Recognition. IN COLLIER, N., RUCH, P. & NAZARENKO, A. (Eds.) *COLING 2004 International Joint workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA/BioNLP) 2004.* COLING.

Index