METHODOLOGIES FOR PROTEIN INTERACTION EXTRACTION FROM BIOMEDICAL ABSTRACTS USING A LINK GRAMMAR PARSER

By

Rania Ahmed Abdul Azeem Abdul Rahman Abul Seoud

A thesis Submitted to the
Faculty of Engineering at Cairo University
In Partial Fulfillment of the
Requirement for the Degree of
DOCTOR OF PHILOSOPHY
In

SYSTEM AND BIOMEDICAL ENGINEERING

FACULTY OF ENGINEERING, CAIRO UNIVERSITY
GIZA, EGYPT
July 2008

METHODOLOGIES FOR PROTEIN INTERACTION EXTRACTION FROM BIOMEDICAL ABSTRACTS USING A LINK GRAMMAR PARSER

By

Rania Ahmed Abdul Azeem Abdul Rahman Abul Seoud

A thesis Submitted to the
Faculty of Engineering at Cairo University
In Partial Fulfillment of the
Requirement for the Degree of
DOCTOR OF PHILOSOPHY
In

SYSTEM AND BIOMEDICAL ENGINEERING

Under the supervision of

Prof. Dr. Abou-Bakr M. Youssef
Biomedical Engineering department
Faculty of Engineering
Cairo University

Associate Prof.Dr. Yasser M. Kadah
Biomedical Engineering department
Faculty of Engineering
Cairo University

Associate Prof. Dr. Nahed H. Solouma
Laser Institute
Cairo University

FACULTY OF ENGINEERING, CAIRO UNIVERSITY
GIZA, EGYPT
July 2008

METHODOLOGIES FOR PROTEIN INTERACTION EXTRACTION FROM BIOMEDICAL ABSTRACTS USING A LINK GRAMMAR PARSER

By

Rania Ahmed Abdul Azeem Abdul Rahman Abul Seoud

A thesis Submitted to the
Faculty of Engineering at Cairo University
In Partial Fulfillment of the
Requirement for the Degree of
DOCTOR OF PHILOSOPHY

In

SYSTEM AND BIOMEDICAL ENGINEERING

Approved by

Examining Committee

Prof. Dr. Abou-Bakr M. Youssef	**Thesis Main Advisor**
Associate Prof. Dr. Yasser M. Kadah	**Advisor**
Prof. Dr. Samia Mashaly	**Examiner**
Prof. Dr. Mohamed Emad Mousa Rasmi	**Examiner**

FACULTY OF ENGINEERING, CAIRO UNIVERSITY
GIZA, EGYPT
July 2008

Table of Contents

CHAPTER 4 SYSTEM ARCHITECTURE

CHAPTER 5 LINK GRAMMAR

List of Tables

List of Figures

List of Abbreviations

PIELG	Protein interaction extraction system using a link grammar parser
LG	Link Grammar
LGP	Link Grammar Parser
IE	Information Extraction
IR	Information Retrieval
NLP	Natural Language Processing
BAN	Biological Association Network
TE	Tissue engineering
DF	Dentine Formation
POS	Part of Speech

ACKNOWLEDGMENTS

First of all, I would like to thank my supervisor Associate Prof. Dr. Yasser M. Kadah for his guidance in academics. He is a great researcher with keen interests in science, constant efforts in doing things by hand and great personality. Not only he gave me a gate to enter this new field, he has also advised and encouraged me all the time. Without his deep understanding of the best supervising manner for a student, I could not achieve the goal of my thesis. He is the example I followed and will continue to follow. I would like to thank my committee members for reviewing this thesis and providing valuable suggestions. I extend my sincere gratitude and appreciation to all the people who made this thesis possible. My sincere thanks are to Prof. Dr. Abou-Bakr M. Youssef and Associate Prof. Dr. Nahed H. Solouma for being on my thesis supervision. I give my special thanks to my friend: Eng. Mai Said Mabrouk, for her advice, comments and cheering continued from the beginning of my history as a researcher in that field. And, I also thank my friend, Eng: Vidan Fathy Ghoniem, for providing the guidance throughout my work. I also thank my friend, Eng: Dina Samy Mohamed, for reviewing this thesis. I am also grateful to my school friends who gave me light in the dark days in my life. Finally, I'm grateful to my family for their support and encouragement of my academic pursuit. I thank my sisters for supporting me in these years. I am very grateful for the love and the unconditioned support of my mother who makes my life more colorful and enjoyable.

ABSTRACT

The last decade has seen unprecedented growth in both the production of biomedical data and amount of published literature discussing it. Tissue engineering laboratories at ***Alexandria University*** aim to regenerate dental tissues by tissue engineering principles and technology (dentine formation process). Dentine formation is governed by biological mediators or growth factors (protein) and interactions amongst different proteins. Dentine formation needs the support of continuous updated information about protein-protein interactions. Thus, having a scalable, robust system for protein interaction discovery provides a major information extraction tool for molecular biologists to automatically extract and transfer updated biological data about protein-protein interactions from unstructured form, to a structured form to be used in their respective applications.

Thus in this thesis, we present PIELG: a system for extracting information about protein – protein interactions from abstracts of biomedical papers. The data obtained from this system will be first confirmed partially in their laboratory. Then they will use those extracted information about protein – protein interactions in Dentine formation process. Our approach is based on first splitting abstracts into simple sentences. Then, the system tags biological entities with the help of biomedical and linguistic ontologies. Finally, the system extracts complete interactions by analyzing the matching contents of syntactic roles and their linguistically significant combinations.

PIELG handles complex sentences and extracts multiple and nested interactions specified in a sentence. The scope of our experiments is limited to abstracts describing human protein function. The corpus of the PIELG is selected in order to evaluate the proposed protein-protein interaction validation method. This corpus is selected to be about proteins currently considered to have roles in the *dentine formation* process and involved in dentinogenesis. We performed experimental evaluations of the PIELG systems.

The interactions extracted by the PIELG system are manually examined for precision and recall. The sensitivity of the system is given by the recall measure, calculated as the ratio between the interactions extracted correctly and the interactions present in text. Precision is a measure of correctness of the system by measuring the number of times the results are extracted correctly in comparison with the total number of results. Our experimental results show that the PIELG system presented here achieves better performance without the need of manual pattern creation (by user) which is required for the other systems.

Publications

The preliminary contribution regarding protein - protein interactions extraction system has been published in [1]. More advanced contribution regarding protein - protein interactions extraction system has been published in [2].

"بسم الله الرحمن الرحيم"

Dedication

I dedicate this work to my dear Mother, may God bless her.

CHAPTER 1

INTRODUCTION

Information Extraction (IE) is a task of Natural Language Processing (NLP) to extract useful information from a set of text. One of applications of IE is to help researchers who struggle with large amount of research papers. In research domains with great scientific successes, such as molecular-biology, research papers are numerically exploding. Knowledge buried in natural language representations are hard to be searched manually in a practical speed. The last decade has seen unprecedented growth in both the production of biomedical data and the amount of published literature discussing it.

Advances in computational and biological methods have remarkably changed the scale of biomedical research. Complete genomes can now be sequenced within months and even weeks**,** using computational methods which expedite the identification of tens of thousands of genes and large-scale experimental methods. The data generated by these experiments is highly connected; the results from sequence analysis and micro-arrays depend on functional information and signal transduction pathways cited in peer-reviewed publications for evidence.

Though scientists in the field are aided by many online databases of biochemical interactions, currently a majority of these are created by domain experts. Information extraction from text has therefore been pursued actively as an attempt to extract knowledge from published material and to speed up the creation process significantly. Thus there is an increasing need for IE tools to support extracting such knowledge from text and building databases. If we have accumulation of such knowledge database, novel knowledge is also expected to be found by reconstructing the knowledge accumulation.

The growth in both the production of biomedical data and the amount of published literature discussing proteins and their interactions are seen unprecedented. Proteins are the macromolecules that make a living organism tick. For example, the transportation of oxygen in blood, the working of a nervous system and the movement of muscles are all highly dependent on proteins and their interactions. Thus, knowledge about the interactions between proteins is valuable information to the scientists who study the biology of living beings. This information can be used to gain insight into, for example, how cancer cells work or what triggers epilepsy, and to ultimately find better cures for these diseases or to prevent them altogether.

The number of different proteins is huge and the exact number is still not known. In any case, the number of proteins in the human body alone runs in the hundreds of thousands and consequently the number of possible interactions is far greater. Furthermore, the interactions between proteins are like chemical chain reactions with the product of one interaction being the input, inhibitor or catalyst for the next. It is the knowledge about these networks of interactions that are the most useful to the scientists. All in all, what is required by the modern biologists developing, for example, new drugs, are huge databases with information about the proteins and their interactions that affect| directly or indirectly the disease or process under study. Moreover, the information must be in a computer intelligible format suitable to be the input to various tools used to visualize and further process the information.

The problem is that most of the information about proteins is scattered in scientific papers and the subject of one paper is usually no more than a few proteins and one or two interactions. The information would have to be gathered from these papers manually; reading through every relevant paper and picking up the information as it is found. However, there are few problems associated with this approach.

Finding all the relevant articles is not the problem. To allow the scientists to search through these publications, the biomedical field has devised online bibliographic databases such as the MEDLINE [3] and PubMed [4]. It is the sheer number of articles published every day that swamps any attempt to follow anything but a very specific portion of the biomedical field manually. Take MEDLINE which contains abstracts of biomedical papers as an example: containing abstracts from 4800 journals and in 40 languages, there has been a constant rate of 1500 new citations added every day since 2002, totaling to approximately 571000 citations added in year 2004 alone. Looking at the numbers, it is easy to understand that it is impossible for any single individual or organization to read through all the published papers. The need for some automation is imminent.

1.1 Justification of the Study

Genomic and proteomic research in the last decade has resulted in the production of a large amount of information about protein function. The generated data is highly connected; hence such data is made easily available. Scientists in that field are aided by many online databases covering different aspects of protein function, such as protein–protein interaction. However, since they are dependent on human experts, they rarely store more than a few thousand of the best-known protein relationships and do not contain the most recently discovered facts and experimental details.

Tissue engineering laboratories at ***Alexandria University*** aim to construct a biological association network (BAN) for the process of dental pulp formation in normal and pathogenic cases. They will carry out some structured studies on one or several proteins in this BAN. So they need continues updated information about protein- protein interactions. Thus, having a scalable, robust system for protein interaction discovery provides a major information extraction tool for molecular biologists to automatically extract and transfer updated biological data about protein-protein interactions from unstructured form, to a structured form to be used in their respective applications.

Information extraction from text has therefore been pursued actively as an attempt to extract knowledge from published material and to speed up the creation process significantly. An automated extraction tool would not only save time and effort, but would also pave the way to discover new unknown information implicitly conveyed in text. This thesis presents a fully automated extraction system, named PIELG, to identify protein interactions in abstracts of biomedical text. PIELG is a protein interaction extraction system using link grammar parser. The system will provide biologists and people in *Tissue* *engineering laboratories* with continuous updated information about protein-protein interactions.

1.1.1 Tissue engineering

Tissue loss or organ failure is one of the most tragic as well as costly problem in human health care. Currently, the major approaches to tissue or organ loss are either reconstructive or transplantation surgery. In a sense, transplantation can be viewed as the most extreme form of reconstructive surgery, transplanting tissue from one individual to another, or implanting foreign body materials. As with successful undertaking; insufficient, rejection due to immune system…etc has appeared. It is within the previous context that the field of tissue engineering has emerged. In essence, new and fundamental living tissue is fabricated using living cells, which are usually associated in one way or another with a matrix or scaffolding to guide tissue development. Living cells can migrate into the scaffold or can be associated with the matrix in cell culture before transplantation.

Tissue engineering construct should resemble native tissues as closely as possible. At present, histology and biochemical methods are commonly used to compare tissue engineered constructs with natural tissues. These techniques are useful to assess the general structure of the implant, although, they don't provide a comprehensive description of the tissue at the molecular level. The shift in the life sciences become possible with the beginning of understanding life at the molecular level and it progresses, largely because of evidences in information technology and mathematical analysis[5],[6].

1.1.2 Bioinformatics

Bioinformatics is the computational techniques for management and analysis of biological data and knowledge. It is the science in which biology, information technology, computer science, mathematics and statistics merge into a single discipline. The science began in 1972 with Professor Margaret Dayoff. [7]. She and her coworkers at the National biochemical Research Foundation (NBRF) assembled the proteins sequences into database. Then around 1979 at the European Molecular Biology laboratory (EMBL) professor Walter Goad and her coworkers' assembled DNA sequences and the translated DNA sequences into databases which are called EMBL Data bank of Japan (DDBJ) came into existence followed by the GeneBank database in 1992 [8].

1.1.3Tissue engineering and bioinformatics

Since then databases continue to grow. Now these databases can generally be classified into protein databases and DNA databases. Recently EMBL, SP and PIR united together to give the Uniport database. Meanwhile, in the last ten years different powerful techniques toke the control in the field of genome sequencing whereas the field of protein structure – function relationship did not advance with the same rate. Database are not only limited on those of DNA and proteins but we can find scientific literature, books and taxonomy databases (as PubMed, bookshelf and taxonomy database respectively generated by the National Institute of health) as well. Mining these databases needs certain programs that can analyze their data with respect to certain keywords given by the researcher. In the field of tissue engineering, bioinformatics is widely applied especially in material science and scaffold design and formation.

1.1.4 Alexandria University's project

Tissue engineering laboratories at ***Alexandria University*** were established in 1999, their focus is to emphasize the stem cell research. Their junior researchers have isolated bone marrow mesenchymal stem cells and utilized in alveolar bone regeneration [9]. Currently, they developed experience in the model of regenerating dental tissues by tissue engineering principles and technology. Throughout their research work**,** they utilize isolated pulp stem cells seeded onto porous scaffold to foster pulp healing and repair by dentinogensis. Realizing the fact that ***Dentine formation*** process is governed by biological mediators or growth factors (naturally occurring protein) that regulate cell proliferation, differentiation and mineralization have drawn our attention that biological interactions amongst different protein have strong link to pulp repair and healing. Using classical research methods will take a long period of time to stand on this relationship. In this case bioinformatics becomes the method of choice to study this relationship.

The aim of their project is based on three steps:

1. Firstly, mining the databases for relation between different factors and the proteins involved in the dentinogensis process. Then biological association network (BAN) will be developed based on different bioinformatics tool to build a possible relation between all the parameters in dentinogeesis.
2. This BAN will be tested in their laboratory to stand on its validity.
3. Using bioinformatics in 3D structure modeling, one or several proteins will be chosen to study their structure and how it can affect the dentinogenesis process in normal and pathogenic cases.

1.1.5 PIELG and Tissue Engineering

The information extraction systems will provide the area of proteomics with unlimited and updated knowledge towards the novel sequencing applications which will increase number of new drug targets, therapeutics molecules and biological disease marker. So, the data obtained from the PIELG system will help people in the ***tissue engineering laboratories at Alexandria University*** in ***Dentine formation*** process.

Their project in its first step aimed to develop biological association network (BAN) based on different bioinformatics tool to build a possible relation between all the parameters in dentinogeesis. The PIELG system will be combined with visualization tool (Cytoscape) for evaluating and drawing the extracted interaction by drawing its pathways. Cytoscape is an open source bioinformatics software platform for *visualizing* molecular interaction networks. Then this BAN will be tested in their laboratory to stand on its validity. They study the structure of specific proteins and how it can affect the dentinogenesis process in normal and pathogenic cases. The data obtained from the PIELG system will be confirmed partially in their laboratory.

1.2 Objectives of the Proposed System

This thesis presents the PIELG system. PIELG is a Protein Interaction Extraction System using a Link Grammar Parser from biomedical abstracts. PIELG is a fully automated extraction system to extract protein interactions in natural language texts. Our approach tags protein names with the help of protein names and linguistic ontologies. PIELG uses a dependency based English grammar parser, the Link Grammar Parser, to identify the roles. The system extracts complete interactions by analyzing the matching contents of syntactic roles and their linguistically significant combinations. Our scheme follows the following steps:-

1. The user has to give as input some keywords (protein names) which he\she thinks best represents and characterizes the required protein.
2. PIELG starts retrieving all PubMed's abstracts satisfying user's specification.
3. PIELG identifies all interaction words which best represent and characterize the required protein- protein interaction.
4. PIELG takes all synonyms and hyponyms of the chosen interaction words. A hyponym of a word is essentially similar in meaning but is more specific.
5. PIELG now searches for all occurrences of the interaction words identified in step 4.
6. PIELG runs the chosen documents through the link grammar parser which tags the words according to the part of speech and assigns a syntactic structure to the sentence.
7. Having identified all sentences where either the interaction words or one of its synonyms and hyponyms acts as a main verb. Each occurrence of the interaction word or one of *its* synonyms and hyponyms is considered to be one occurrence of the required interaction.
8. PIELG uses rules to identify the subject and object (if present) of the verb as well as the modifiers of all three (verb, subject and object). *So,* by finding the subject, object as well as all available modifiers, almost all information about that instance of the event can be extracted from the document.
9. PIELG extracts the complete interaction.

The system uses Phrasal-prepositional Verbs Patterns to overcome preposition combinations problems. PIELG is purely implemented with Perl under Linux platform. The scope of our experiments is limited to abstracts describing human protein function. The corpus of the PIELG is selected in order to evaluate the proposed protein-protein interaction validation method. This corpus is selected to be about proteins currently considered to have roles in *dentine formation* process and involved in dentinogenesis. We performed experimental evaluations with two other state-of-the-art extraction systems – the BioRAT and IntEx indicate that PIELG system achieves better performance.

The evaluation of the performance for the PIELG system is measured with two traditional meters: precision and recall. The recall and precision are 47.4% and 62. 65%. For further evaluation, the PIELG system is augmented with a graphical package for extracting protein interaction information from sequence databases. We used Cytoscape[1] which is a good tool for drawing directed graphs that can be adapted for drawing interaction pathways. The augmentation process is done for two reasons. The first reason is to visualize the extracted pathways. The second reason is to evaluate the extracted interaction by drawing the pathways for the extracted interaction. Then we compare those pathways with the stored pathways in Cytoscape. Our experimental results show that the PIELG system presented here achieves better performance without the need of manual pattern creation (by user) which is required for other systems.

[1] http://www.cytoscape.org/

1.3 Thesis Organization

This section describes the structure of this thesis.

Chapter 1 is an introduction of the central issues and the new approach *proposed in this thesis and* presents the justification of the study.

Chapter 2 covers the background of information extraction and its basic component technologies, including named entity recognition, entity relation detection and event extraction.

Chapter 3 surveys the classical approaches for Information Extraction, ranging from rule-based approaches and symbolic learning to statistical models. The related work is surveyed.

Chapter 4 presents an architectural overview of the PIELG system.

Chapter 5 and 6 explain and illustrate the individual modules of PIELG system.

Chapter 7 provides the results of the PIELG system with an analysis of the results. A detailed evaluation of the system is presented with the visualization process for the results of PIELG system using Cytoscape which is a good tool for drawing directed graphs that can be adapted for drawing interaction pathways. Then we evaluate the extracted interactions by drawing the pathways for them. Then we compare those pathways with the stored pathways in Cytoscape.

Chapter 8 is the conclusion.

CHAPTER 2

INFORMATION EXTRACTION

Human knowledge about the world is complicated. Even after decades of research, there is still no effective way to represent the full range of real world knowledge. Although it is impossible to obtain a universal representation of knowledge, we can make the problem tractable by confining the domain of the text. Then it is possible to represent the underlying world knowledge or semantics in a simple format like templates. Text has been a major way to store and convey information in human society. With the development of the internet and digital media, a user can have instant access to a huge amount of text. The volume of text available on the web is accumulating at a constantly increasing speed. The world in text is full of information and how to locate the specific information a user needs becomes a critical issue. The automated handling of text is an active research area, spanning several disciplines. These include the following:

1. *Information retrieval*, which mostly deals with finding documents that satisfy particular information need within a large database of documents.
2. *Natural language processing (NLP)*, a broad discipline concerned with all aspects of automatically processing both written and spoken language. A central goal of Natural Language Processing (NLP) is to be able to understand the underlying meaning of texts and translate them into machine comprehensible representations. Then the computational power of machines would enable us to manipulate the information in more user-friendly ways, such as producing summaries or answering questions.

3. *Information extraction (IE),* a subfield of NLP, centered on finding explicit entities and facts in unstructured text. It is the practical way to get one step closer to the goal of NLP. It is domain-dependent.

2.1 Information Retrieval (IR)

Information retrieval is concerned with identifying documents that are most relevant to a user's need within a very large set of documents. More precisely, given a large database of documents, and a specific information need—usually expressed as a *query* by the user—the goal of information retrieval methods is to find the documents in the database that satisfy the information need. Naturally, the task has to be performed accurately and efficiently [11].

2.1.1 Boolean queries and index structures.

There are several ways to express, as well as to satisfy, the information need. A simple and common way for a user to express her need is through a *Boolean* query. Under this setting, the user provides a term (e.g. *OLE1*), or a Boolean term combination (e.g. *OLE1* and *lipid*). The result is the set of *all* the documents in the database satisfying the query constraints, e.g. containing both the query terms *OLE1* and *lipid*. This query paradigm is used by the biomedical literature database and search engines over the World Wide Web. It is supported by an index covering all the terms in the whole database of documents. Each *term* may be a single word (e.g., *blood*) or a phrase (e.g., *blood pressure*) [12].

It is common practice to omit from the index terms that are frequent and non–content–bearing, such as prepositions. These terms are usually referred to as *stop words* and are viewed as delimiters when processing text. The index structure contains all the terms, typically sorted alphabetically for quick access, and holds for each term a reference to all the documents in the database that contain it. When a user poses a query, the index structure is efficiently searched for the query terms occurring in it, and all the documents found to contain the terms (or the Boolean combination of the terms) are retrieved. Further information on this subject is available in books concerning databases and information access, such as the one by [12].

2.2 Natural language processing: General techniques

Natural language processing is concerned with all aspects and stages of converting spoken, handwritten, or printed text from a raw signal to information that can be used by either humans or automated agents. In the context of bioinformatics, we are concerned only with printed text that is already stored in a machine accessible format and therefore concentrate on common text processing operations [13] as used by typical text mining systems. These include the tokenization and zoning tasks, part of speech tagging, and (shallow) parsing.

2.2.1Tokenization.

The first step in text analysis is the process of breaking the text up into its constituent units—or its *tokens*. This process is known as *tokenization*. Tokens may vary in granularity depending on the particular application. Consequently, tokenization can occur at a number of different levels: the text could be broken up into chapters, sections, paragraphs, sentences, words, syllables, or phonemes. For any level of tokenization, many different algorithms exist for breaking up the text. The most common form of tokenization in mining systems is the fragmentation of text into words and sentences. The main challenge of fragmentation at the sentence boundaries is distinguishing between a period that signals an end of sentence and a period that is part of a previous token like the shorthand *Mr.*, *Dr.*, etc.

2.2.2 Part-of-speech tagging.

Part-of-speech tags are a set of word-categories based on the role that words may play in the sentence in which they appear. *Part of Speech (POS) Tagging* is the annotation of words with the appropriate POS tags, based on their context within the sentence. POS tags convey information about the semantic content of a word. *Nouns* usually denote tangible and intangible entities while *prepositions* express relationships between entities. While sets of tags may vary, most part-of-speech tag sets make use of the same basic categories. The most common set contains seven different tags: *Article, Noun, Verb, Adjective, Preposition, Number,* and *Proper Noun*. Some systems use a much more elaborate set of tags. For example, the complete Brown Corpus [14] tag-set has 87 basic tags.

Several approaches exist to POS tagging. The most common taggers are either *rule-based* taggers or *probabilistic* ones based on hidden Markov models (HMMs). HMM-based taggers, [15] estimate the probability of a sequence of part-of-speech tags to be assigned to a given sequence of words, based on a probabilistic (Markov) model. In order to estimate the model parameters, the tagger undergoes a training phase, using an annotated corpus, such as the WSJ corpus in the Penn Treebank [16].

The latter consists of about one-million tagged words. Using a tri-gram model (that is, a model in which the current word-tag depends only on the tags assigned to the two preceding words), HMM-based taggers have achieved 94–96% accuracy on held-out test sets, i.e., sets other than the ones used for training the model. On the other hand, typical rule-based approaches [17] rely on rules that use contextual information to assign tags to unknown or ambiguous words. These rules are often known as *context frame rules*. For instance, a context frame rule might say: “*If an ambiguous/unknown word X is preceded by a determiner and followed by a noun, tag it as an adjective*.”

In addition to contextual information, many rule-based taggers use morphological information to aid in the disambiguation process. For example [18] if an ambiguous/unknown word ends with an *"ing" suffix* and is preceded by a verb; it may be tagged as a verb. Another source of hints for the correct tagging of words can be obtained from orthography such as capitalization and punctuation. For some languages, such as English and German, information about capitalization proves extremely useful in the tagging of unknown nouns; usually capitalized nouns would be tagged as proper nouns. In other languages, such as Hebrew and Arabic, there are no capital letters; hence, no hints can be derived from orthography.

Initially, rule-based taggers required human-tagged training sets, for what is known as *supervised* learning of rules. However, more recently, several researchers [19] started to work on *unsupervised* rule-learning, or *bootstrapping*. Starting with an untagged text corpus and a coarse, generic tagger, the tagger assigns tags to the corpus. An expert reviews the tagged text and corrects any mistake found. In practice, the expert does not typically have to correct more than 20% of the words. The corrected tagging is then run again through the tagger, where special emphasis is placed on words which were erroneously tagged in the first phase. This iterative process, of expert review followed by a tagger rerun, may be repeated until an acceptable error rate is reached.

2.2.3. Parsing and shallow parsing.

Parsing is the process of determining the complete syntactic structure of a sentence or a string of symbols in a language. A parser usually takes as its input a sequence of tokens that were extracted from the original text by a lexical analyzer. The output from the parser is typically an abstract syntax tree, whose leafs correspond to the individual words (lexemes) in the text, and whose internal nodes represent syntactic structures, identified by grammatical tags, such as *Noun, Verb, Noun Phrase, Verb Phrase*, etc. Efficient and accurate parsing of unrestricted text is not within the reach of current techniques. Standard algorithms are too expensive to use on very large corpora and are not robust enough.

A practical alternative is *shallow parsing*. This is a coarser process of breaking documents into non-overlapping word sequences or *phrases*, such that syntactically related words are grouped together. Each phrase is then tagged by one of a set of predefined grammatical tags such as *Noun Phrase, Verb Phrase, Prepositional Phrase, Adverb Phrase, Subordinated clause, Adjective Phrase, Conjunction Phrase,* and *List Marker*. Shallow parsing has the benefit of both speed and robustness of processing, which comes at the cost of compromising the depth and fine-granularity of the analysis. Shallow parsing is generally useful as a preprocessing step, either for bootstrapping—extracting information from corpora for use by more sophisticated parsers—or for end-user applications such as information extraction. Shallow parsing allows the identification of relationships between the object, the subject, and any other spatial or temporal phrase within a sentence.

2.2.3.1. Full versus shallow parsing in IE.

We introduced the concepts of parsing and shallow parsing in the previous section. Based on actual empirical evaluation, it was found that it is enough to focus just on the core constituents of sentences and use shallow parsing augmented by *smart skips*. These skips enable the information extraction engine to skip irrelevant parts, and focus just on the important phrases of each sentence [20]. Researchers have attempted before to use full parsing as a component in their information systems and have concluded that it was not worthwhile to invest the extra effort. Specifically, full parsing was included in the SRI TACITUS system [21] (implemented for MUC-3) and the NYU PROTEUS system [22] (implemented for MUC-6). Both of these systems did not gain any improvement in accuracy due to the full parsing employed. The main problem with using full parsing is that due to the combinatorial explosion of possible parses it is both slow and very error prone.

A full parsing approach has not been used in practical applications on the basis of the following three reasons. First, full parsers in general tend to be slower, and need a larger memory than shallow analysis because they handle the full possible structure of whole sentences even when the full structure is not necessary. Second, it is often argued that the results of full parsers have more ambiguity because full parsers produce the full structure of a sentence whereas shallow methods produce a partial structure by ignoring the part of the sentences that does not match the pattern. Third, full parsers have lower coverage than shallower analyzer because of the complexity of process.

2.2.3.2. Syntactic Role versus Semantic Role.

Syntactic role labeling, done using syntactic parsers (like Link Grammar Parser, Charniak Parser etc.), considers the roles played by the constituent syntactically with respect to the main verb phrase of the sentence. Whereas Semantic role Labeling, uses features derived from different syntactic views and combines them within a phrase based chunking paradigm as described in works by [23]. Semantic role labels are assigned to the constituents of each parse using SVM classifiers. The task of semantic role labeling involves tagging groups of words in a sentence with the semantic roles they play with respect to the particular predicate in the sentence. Identifying Semantic roles needs domain and semantic knowledge. New areas of research in this field are coming up like Semantic tagging (FrameNet) [24] based on frame semantics. Semantic Parsers for English language will be more useful and meaningful for extraction task compared to Syntactic parsers. But constructing semantic parsers is a difficult task and they will be more domain-dependent.

2.3 Information Extraction (IE)

The success of information extraction system depends on the performance of the various subtasks involved. Figure (2-1) gives an overview of the subtasks in information extraction. Information extraction systems that combining NLP tools typically have three to four major components:

1. Tokenization or zoning - splitting the document into words, sentences, or paragraphs.
2. Morphological and Lexical analysis - assignment of part-of-speech (POS) tags, identifying Noun Phrases, Verb Phrases, or disambiguating word sense, Named Entity Recognition.
3. Syntactic analysis - shallow parsing, or full parsing.
4. Domain analysis - anaphora resolution, combining together all the information with respect to the domain on hand.

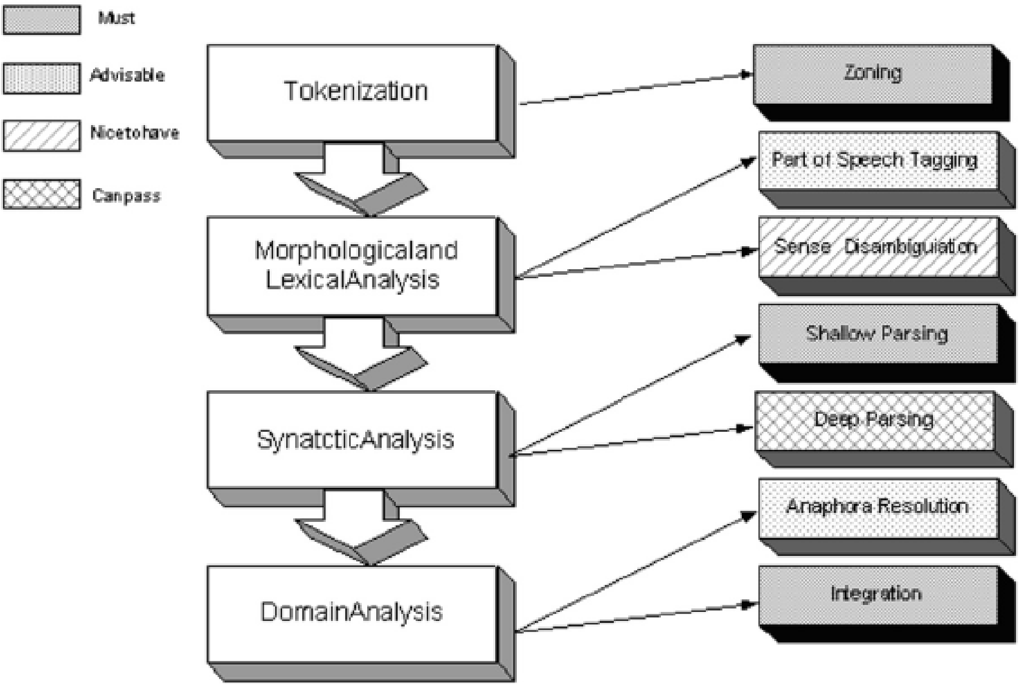


Figure (2-1): Architecture of a typical Information Extraction system. [25]

2.3.1 Information extraction for bioinformatics

Most efforts concerned with biomedical literature mining to date focus on automated *information extraction*. For instance, identifying all the positions in the text that mention a protein or a kinase (entity extraction), or finding all phosphorylation relationships to populate a table of phospohrylated proteins along with the responsible kinase (relationship extraction) are both IE tasks.

Most of the IE systems focused on extracting interactions between genes and proteins. Biologists are also interested in their corresponding protein-protein interaction pathways. Besides, extracting interactions between proteins alone without information such as locations on where the interactions occur can be misleading to biologists. In the case of sentences describing gene location on chromosomes, the constituents forming the sentence are gene and chromosome names, words describing location, and terms denoting experimental methods that validate the location of a gene on a chromosome. Names of genes and chromosomes are identified by Named Entity Recognition. It is a simple heuristics (e.g. terms in all-capital letters which include numbers are regarded as gene names). The experimental methods as well as localization indicators are provided in a predefined list.

After all the mentioned name entities in a text have been identified, we need to recognize their relations. Relation extraction is a task to extract pairs of named entities which have target relations, e.g. pairs of interacting proteins. The sequential event extraction is a task to extract sequences of relations (which represent events in this case), e.g. sequences of protein interactions. Recognition of relations between entities can help us to connect events. A survey of techniques used in protein name extraction is presented in the next section. A survey of techniques used in protein interaction extraction is presented in the next chapter.

2.3.2 Survey of Named Entity Recognition techniques

Text usually contains all kinds of names, for example person names, company names, sports teams, chemicals and lots of other names from a specific domain. Other common units can also fall into this category, such as time expressions, numbers or job titles. These names are referred to as named entities (NE) in Information Extraction. Failing to recognize them as a unit would affect the accuracy of deeper analysis of text, such as chunking or parsing. Therefore named entity recognition becomes a basic component technology for Information Extraction or Natural Language Processing in general. Entity extraction in biomedical domain or named entity identification is the process of identifying the words or phrases of interest such as genes, proteins, protein families, drugs, chemicals and pathways in text.

The simplest and most frequently used approach is a dictionary matching approach where the entity names are compiled as a dictionary and a string match with an entry in the dictionary tags the words or phrases as gene or protein names. A variety of publicly available databases provide the resources for entity names. NCBI"s LocusLink [26] and UniProt [27] are among the databases that provide gene and protein names and their synonyms.

The use of standardized dictionaries containing the names and synonyms of proteins, genes and small molecules has been shown to be an effective way for recognizing these entities in free form text [28]. Although applications of this technique have reported high rates of recall and precision, this technique remains limited as protein, gene, and small molecule names not present in the dictionaries produce large amounts of false negatives. This method is used in PIELG system to identify protein names in the text.

Others have addressed the issue of false negatives by using templates capable of recognizing common naming patterns for genes, proteins, and small molecules [29]. These techniques, which scan potential names by looking for patterns of capitalization, numbering, and use of hyphens have been shown to capture many of the entities missed by the dictionary approach alone, thereby reducing the amount of false negatives. However, these techniques have also been shown to generate a large number of false positives by recognizing words that match the templates but are in fact not protein, gene, or small molecule names.

Entity identification has also been thoroughly researched over the years. Various approaches have been applied to detect named entities in text, such as Decision Tree [30], Maximum Entropy [31] [32] and Hidden Markov Model [33]. The HMM model is simple and effective in capturing the sequential relations between words inside and around a name. Many named entity recognition implementations are based on this model. Recently Support Vector Machines were also applied to this task with good performance reported [34]. Entity Identification systems generally use rule based approaches and machine learning techniques to mark the phrases of interest in text. Rule based approaches rely on regular expressions and heuristic rules to identify gene names. In [29] they follow a combination of regular expressions and expansion rules to identify single word and multi-word gene names. In [35] they also follow a rule based approach to identify biological entities in text.

Alternative approaches have addressed the problems of name recognition through the use of machine learning, and through the use of statistics. Some of the machines learning approaches followed for NER include decision trees, Bayesian classifiers, iterative error reduction, boosted wrapper induction and support vector machines. The ABGene system from Tanabe and Wilbur [36] uses the Brill's tagger [17] to learn transformation rules to tag the gene and protein names in text. The rules are based on the word occurrences, neighboring words and part of speech tags of the words and the neighbors. Although these techniques have reported incremental gains in overall recall and precision over the template and dictionary based approaches, it has been shown that these techniques are also limited by the quality and extent of the training sets used to train the algorithms**.**

Advanced Text Mining (TM) such as semantic enrichment of papers, event or relation extraction, and intelligent Question Answering (QA) have increasingly attracted attention in the biomedical domain. For such attempts to succeed, text annotation from the biological point of view is indispensable. Research in entity recognition has resulted in the development of various corpora for the purpose of providing a benchmark for the entity recognition systems. The GENIA corpus, a hand-annotated corpus of abstracts from over 2000 Medline articles on human blood transcription factors uses the GENIA ontology to tag concepts in text. The recent JBLPBA challenge used the GENIA corpus as the test data for its shared task on Entity recognition. The participants in the task used various machine learning approaches, sometimes using a combination of approaches such as the support vector machines and Hidden Markov Model of Zhou [37]. The results from the task can be obtained from their webpage[2]. The BioCreative[3] corpus is more general in nature, deliberately constructed with challenging false positives by the National Library of Medicine.

BioInfer[4] (Bio Information Extraction Resource) is a new public resource providing an annotated corpus of biomedical English. The annotation scheme captures named entities and their relationships along with a dependency analysis of sentence syntax. They further present ontologies defining the types of entities and relationships annotated in the corpus. Currently, the corpus contains 1100 sentences from abstracts of biomedical research articles annotated for relationships, named entities, as well as syntactic dependencies [38]. Supporting software is provided with the corpus. The corpus is unique in the domain in combining these annotation types for a single set of sentences, and in the level of detail of the relationship annotation.

In [39] they have completed a new type of semantic annotation, event annotation, which is an addition to the existing annotations in the GENIA corpus. As in BioInfer, they do not allow annotators to annotate an event unless an expression mentioning the event type appears in the text. However in their attempt they deliberately dissociate annotation from linguistic structures, and events in their annotation are not necessarily organized around verbs. That is, an event does not necessarily correspond to a constituent such as a clause or phrase, governed by a verb. Expressions which indicate occurrences of an event and expressions which describe its participants (arguments) can be scattered

[2] http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/ERtask/report.html

[3] http://www.mitre.org/public/biocreative/

[4] http://mars.cs.utu.fi/BioInfer/

throughout a sentence without constituting a single constituent in the linguistic structure. The corpus has already been annotated with POS (Parts of Speech), syntactic trees, terms, etc. The new annotation was made on half of the GENIA corpus, consisting of 1,000 Medline abstracts. It contains 9,372 sentences in which 36,114 events are identified. The major challenges during event annotation were (1) to design a scheme of annotation which meets specific requirements of text annotation, (2) to achieve biology-oriented annotation which reflects biologists' interpretation of text, and (3) to ensure the homogeneity of annotation quality across annotators. To meet these challenges, we introduced new concepts such as Single-facet Annotation and Semantic Typing, which have collectively contributed to successful completion of a large scale annotation [40].

2.4 Biological Context

The role of biology in the IE process is to be the specific context in which the language of the documents is processed and analyzed. It does not differ much from any other domain to which information extraction could be applied, but there are, of course, some variation points that cause the challenges presented by the biology domain to differ in their details. It should also be noted that the interest here is not in biology in general, but rather in biology and biochemistry that is applied to medicine, i.e. biomedicine.

2.4.1 Central Biological Concepts

Before venturing any further into the challenges presented by the biology domain, the central biological concepts and terminology that will be encountered in this thesis need to be introduced and defined.

DNA (deoxyribonu-cleic acid): - which is the genetic code of the cell. Cells in all living things contain DNA in their nucleus. It consists of a chain of four types of bases, called adenine (A), guanine (G), thymine (T) and cytosine (C). The DNA chain can be divided into segments called genes that contain the information needed to build proteins.

Proteins: - they are polymer chains of amino acids. There are 20 different kinds of amino acids that can be used to build a protein. All functions in a living organism depend on them; proteins do everything from making muscles move to controlling biochemical processes and transporting materials, such as oxygen (the oxygen binding hemoglobin of red blood cells is a protein), and carrying signals, such as nerve impulses. The process of transforming the instructions in a gene into a protein involves two steps: transcription and translation.

In the transcription phase, the DNA containing the gene produces a messenger RNA (mRNA), which is transported out of the cell nucleus into the cell cytoplasm. The mRNA resembles the original DNA, but with the thymine (T) substituted with Urasine (U). In the translation phase the sequence of the mRNA are read in triplets called codons. A special kind of transport RNA (tRNA) carrying the correct amino acid attaches to each codon. When the tRNAs attach themselves to the mRNA side-by-side, the amino acids they carry form polypeptides that then form the protein.

2.4.2 Protein-protein interactions

Proteomics is aimed at understanding protein-protein interactions. The function of a protein can be characterized more precisely through knowledge of protein-protein interactions. Protein-protein interactions are important for many biological functions. Protein-protein interactions play an important role in vital biological processes (cell cycle control, metabolic and signaling pathways). They link many proteins in the cell into large connected interaction networks. Each protein can have one or more of many roles in the network. Moreover, networks of interacting proteins provide a first level of understanding the cellular mechanism.

Protein interaction information is stored in mostly manually curated databases. However, the amount of biomedical literature is increasing rapidly. Thus, it is difficult for database curators to detect and curate protein interaction information manually. Most of the protein interaction information remains uncovered in the biomedical literature. Development of information extraction and text mining techniques for automatic extraction of protein interaction information from free text is crucial.

Even the simplest process in an organism or a single cell involves many proteins that interact to carry out a specific function or task. Each protein has its own role to play and so the process can be thought of as a network of interactions. These biochemical networks are called **pathways**. There are three different types of pathways, of which the most interesting in the context of this thesis are the signaling pathways, because they represent protein-protein interactions. Each protein can have one or more of many roles in the pathway. For example, some protein might inhibit a biochemical process, while some other protein might bind with another protein to promote the same.

2.5 Biomedical Sources of Information

Human Genome sequencing marked the beginning of the era of large-scale genomics and proteomics, which in turn led to a humongous amount of information. Most of it is unstructured text of published literature. The most used online source of biomedical resource is PubMed[5] database, which is maintained by National Center of Biotechnology Information (NCBI). It contains over 13 million scientific abstracts. Biomedical papers, journals and other publications are the sources from which the information can be extracted. Thus the structure and language used in them is central to the problem. Many of the things that are going to be said apply equally to any other field of science, so these things do have a bearing on the information extraction process in general.

PubMed is accessed by millions of users from all over the world on a daily basis. A typical search for relevant literature within PubMed starts with a Boolean query; the user provides a term or a Boolean term combination (e.g., *OLE1* and *lipid*). The result is the set of *all* the abstracts in PubMed satisfying the query constraints. We note that the lack of uniformity in nomenclature used by authors aggravates the problem of synonymy**.**

The structure of scientific papers is quite the same across disciplines: abstract, introduction, methods, results, discussion [41]; plus supplementary front and back matter, such as heading, acknowledgements, various indexes and bibliography listings. Each of the parts has its own characteristics which make it more or less interesting as the source of factual information. Of these parts, the most interesting in the light of automatic information extraction is the abstract. The abstract contains a brief summary of the key findings of the paper and thus, the basic facts should already be extractable from therein. There are also other properties that make abstracts the most important source for information extraction systems. First of all, abstracts are usually available electronically in plaintext format, as opposed to being in PDF or some other non-plaintext format. Secondly, and this is also very important from the

[5] *www.ncbi.nlm.nih.gov/entrez/query.fcgi*

commercial point of view, abstracts are usually available free of charge, whereas the rest of the subject matter can require a costly subscription. This the reason that causes PIELG system to extract information about protein-protein interaction from abstracts of biomedical papers.

In contrast to the abstract, the discussion part is the least interesting. The reason is that the discussion part is least likely to present new factual findings, and instead it usually contains suggestions for new research [42], generalizations and educated hypothesis about what could be found. It is these guesses, expressed in sentences and wordings that closely resemble the expression of facts, that can mislead the information extraction system to extract uncertainties and mere guesses as truths. Thus, it seems that some of the sections of the papers can be categorically excluded from the IE process without loss of information.

CHAPTER 3

INFORMATION EXTRACTION SYSTEM: A SURVEY

Compared to the last two years, the field of information extraction for biology has made tremendous strides. It has witnessed the emergence of software tools that are able to handle the task. Most of the tools were developed to carry out specific tasks. Each tool seems to have developed its own methodology. It follows different strategy, with different details and usually adapts to the task at hand. Most of the early work on automated understanding of biomedical papers concentrated on analytical tasks such as identifying protein names, or relied on simple techniques such as word co-occurrence, and pattern matching. Then, work based on more general natural language parsers that could handle considerably more complex sentences is involved. Then the emergence of more sophisticated natural language technologies that can handle anaphora as well as extracting a broader range of information is considered.

This situation has motivated us to present a classification scheme for these tools based on the underlying computational technique, and to shed light on the background application that caused this differentiation. We hope that this part of the thesis will help the reader to have a comprehensive and comparative overview of the tools developed until now, to make the most of them, and to evaluate our contribution to the problem of comparing genomic sequences.

3.1 Information Extraction Techniques

There has been a wide range of varying techniques published for extracting proteins relationships from scientific literature. Existing protein-protein interaction works can be roughly divided into two categories: co-occurrence based approaches [42] and rule-based approaches [43]. The simplest way to extract protein relations from the literature is to detect the co-occurrence of protein names in a text [44]. It simply uses co-occurrence statistics of two proteins to predict their relation. However, by its nature, the name co-occurrence detection yields very little or no information about the type of described relation and therefore the co-occurrence data may be misleading. This way, they can only extract well-known PPIs but may not be able to find new emerging PPIs.

On the other hand *rule-based approaches* utilize pre-defined phrase pattern rules. As a result, they are unable to discover new phrase patterns without the known keywords. Once the rule set reaches a certain size, it is very difficult to insert additional rules for further performance improvement. Moreover, rule-based approaches may require redefining of the whole pattern rules when they are applied to a new domain. Therefore other researchers [45] adopt a machine learning method to generate these interaction extraction rules automatically. But, previous machine learning approaches, when applied to this domain, suffer from the trade-off between recall and precision. Typically, when precision is high, recall is very low, and when recall is very high, precision is low.

3.1.1 Co-occurrence based approaches

A simpler approach that relies on co-occurrence of genes/proteins within sentences, rather than on machine learning methods or advanced NLP, was used by [28]. Its goal was to extract information about protein interactions among a predefined set of related proteins from scientific text pertaining to them. Using a list of protein names and a list of interaction words, they look for sentences that have occurrences of two protein names separated by an interaction word, to identify relationships among the proteins. An extension to this work is described by [46], where they use a module for protein name detection (an issue we touch on briefly later) and exclude negations. The latter means that interaction facts are extracted only from sentences that affirmatively report the interaction. The exclusion of negation is an interesting point and merits some discussion. The concern about negation sentences (e.g., “We have found *no evidence* that protein A is involved in the regulation of gene B”) is often expressed in the context of mining the biomedical literature.

The assumption underlying this concern is that we want to avoid, for instance, relating protein A and protein B in a regulatory pathway if according to the literature the two are not related. This is indeed a valid point if we aim to automate the construction of pathways through the literature. However, under different scenarios, for instance, when investigating a set of proteins and genes in which protein A is produced just before gene B is expressed, an edge between A and B marked with a “negative regulation” label and linked to the relevant article stating the negative result is extremely valuable. Hence, the reconsideration of negation, its role, and its treatment is pertinent.

Moreover, since these methods depend on the co-occurrence of terms, within a sentence, a phrase, or an abstract they can only reveal relationships that are *already* reported in the literature and do not attempt to detect new relations. We qualify this with the observation that one could follow Swanson’s methodology [47], and use the “transitive” relations—i.e., the indirect-links among entities—as clues for yet-unknown relationships. For instance, if there is a report relating protein A to B, and another report relating B to C, it may suggest a possible (yet-unreported) relation between proteins A and C. It is also important to note that as large-scale experiments using microarrays and other high-throughput techniques are becoming more popular, the co-occurrence of gene and protein names in the literature may become more of an indicator of their inclusion in large-scale experiment rather than of an actual functional relationship between them. When using the literature to interpret the results of large-scale experiments, it is crucial that the literature-mining engine could actually provide an independent insight into the functional and biological relationships—beyond the mere fact that they participated together in a large scale experiment. Methods that strongly rely on co-occurrence alone are insufficient to address this need.

3.1.2 Rule-based approaches

3.1.2.1 Pattern matching

More sophisticated information extraction approaches rely on **the matching of pre-specified templates** (patterns) or rules (such as precedence/following rules of specific words). The underlying assumption is that sentences conforming exactly to a pattern or a rule express the predefined relationship(s) between the sentence entities. In some cases, these rules and patterns are augmented with additional restrictions based on syntactic categories and word forms in order to achieve better matching precision. The pattern-based systems have been applied to extract protein–protein interaction [48] and pathway information [49].

Another popular approach uses **pattern matching**. As an example, a set of simple word patterns and part-of-speech rules were manually coded, for each verb, to extract special kinds of interactions from abstracts [50]. The method obtains a recall rate of about 85% and a precision rate of about 94% for *yeast* and *Escherichia coli*, which is the best among all reported results. However, manually writing patterns for every verb is not practical for general purpose applications.

In [49] they describe how to generate English expression patterns related to protein–protein interactions. They also present a theory which, focusing on how to improve the patterns. They used **dynamic programming** to extract sentence patterns. A minimum description length (MDL)-based pattern-optimization algorithm is designed to reduce and merge patterns. This has significantly increased generalization power, and hence recall and precision rates, as confirmed by our experiments. They also demonstrated that this proposal of automatically generating and optimizing sentence patterns and using them to mine a targeted area of knowledge is feasible. This approach works in other domains, too. The *F*-score of this approach is 2.98% lower compared other approaches in the training set.

In [50] they propose a system of extracting the relationships between proteins by searching frequently seen keywords, their patterns created by surface clues, and a protein dictionary. This technique used only surface clues based on the word patterns that were presented by the word positions. The patterns representations were defined by the position between the keyword, protein names, and other characteristic words, such as prepositions in the sentences. Each sentence containing the pattern was filtered with the rules based on the grammatical part of speech information. They obtained a recall of 86.8% and a precision of 94.4%. This system may become a powerful tool for creating a database, such as protein interaction, from a huge variety of public databases. This suggests that it can be practically used as support to extract protein interaction data when a protein dictionary becomes available.

3.1.2.2 Natural Language Processing-based systems

A parsing technique that many previous approaches used is shallow parsing [51]. Shallow parsing is more robust than full parsing, but it only separates phrases of sentences, i.e. it only yields merely local syntactic relations. More advanced systems utilizing *shallow parsing* techniques have been described to extract protein interactions [52], enzyme reactions and protein structure information [53], or functional relations between proteins [54]. Unlike word-based pattern matchers, shallow parsers perform partial decomposition of a sentence structure. They identify certain phrasal components and extract local dependencies between them without reconstructing the structure of an entire sentence. The precision and recall rates reported for shallow parsing approaches are 50–80% and 30–70%, respectively. Interestingly, most of the described systems are designed to extract only one specific aspect of protein function information.

The most promising candidates for a practical information extraction system are ones based on *full-sentence parsing* as they deal with the structure of an entire sentence and therefore are potentially more accurate. Systems using full parsing can find deep syntactic relations, e.g. a relation between a passive verb and its semantic subject, from the whole of a sentence. Using this generic NLP tool, extraction patterns in a well generalized format could be obtained. Full parsing is used both in a phase of construction of extraction patterns and in a phase of pattern matching (i.e. the actual IE prediction task). Full parsing constructs more general extraction patterns from a less training corpus, than shallow parsing. Full parsing, can identify both the subject of the whole sentence and the semantic subject that has been shared.

A general *full parser* with grammars applied to the biomedical domain was used to extract interaction events by using bidirectional incremental parsing with combinatory categorical grammar (CCG) in [55]. This method first identifies relevant keywords and localizes the target verbs. They used pattern matching around the keywords for NP candidates. It scans the left and right neighborhood of the verb respectively. Then they validate the noun phrase candidates with CCG. The lexical and grammatical rules of CCG are even more complicated than those of a general CFG. The recall and precision rates of the system were reported to be 48% and 80%, respectively.

Another full parser utilizes a lexical analyzer and *context free grammar (CFG)* [56]. Context-free grammars provide an easily extendible platform for extracting interactions from free text and are powerful enough to describe most natural language structure while being able to be restricted enough to allow for efficient parsing. They also describe a methodology for creating a corpus for analyzing techniques that can be extended and potentially used to do comparative analysis between techniques in the future. It extracts protein, gene and small molecule interactions with a recall rate of 63.9% and a precision rate of 70.2%. This approach provides a level of abstraction for adding new rules for extracting other types of biological relationships beyond protein, gene and small molecule relationships. The potential is to be able to mine the larger set of scientific literature available in order to populate structured representations for capturing interaction data for further computational analysis.

A general *full parser* with grammars applied to the biomedical domain was used to extract interaction events by filling sentences into argument structures in [57]. They used a parser that converts the variety of sentences that describe the same event into a canonical structure (*argument structure*) regarding the verb representing the event and its arguments such as (semantic) subject and object. In this work, they introduce two preprocessors that resolve the local ambiguities in sentences to improve the efficiency. One of the preprocessors is a term recognizer that glues the words in a noun phrase into one chunk so that the parser can handle them as if it is one word. The other is a *shallow parser* [58] that reduces the lexical ambiguity. An HPSG-based parsing system (XHPSG) is used as a full parser. As a shallow parser, they adopt ENGCG. Information extraction itself is done using pattern matching on the canonical structure. Event information is then extracted by domain-specific mapping rules from argument structures to frame representations. Using a general-purpose grammar for syntactic analysis makes it possible to modularize the system, so that the IE system as a whole becomes easy to be tuned to specific domains, and easy to be maintained and improved. No recall or precision rate was given.

Similar methods such as preposition-based parsing to generate templates were proposed [59]. They developed a medical parser that extracts information, fills basic prepositional-based templates, and combines the templates to capture the underlying sentence logic. They tested their parser on 50 unseen abstracts and found that it extracted 246 templates processing only abstracts with a template precision of 70%. In comparison with many other techniques, more information was extracted without sacrificing precision. Future improvement in precision will be achieved by correcting three categories of errors.

3.1.3 Machine learning Approaches

The above researchers essentially need some linguistic rules to extract the biological interactions, and most of them use many different hand-crafted rules. But it is time-consuming to construct hand-crafted rules which require much human effort, and these systems are difficult to be applied to other domains. Therefore other researchers adopt a machine learning method to generate these interaction extraction rules automatically. But, previous machine learning approaches, when applied to this domain, suffer from the trade-off between recall and precision. Typically, when precision is high, recall is very low, and when recall is very high, precision is low.

In [60] they proposed an evaluation conducted by NIST to measure IE technologies. They used Maximum Entropy Model to integrate lexical, syntactic and semantic features for relation detection and characterization (RDC) task containing 24 relation types on news articles with Automatic Content Extraction (ACE1, 2004). It shows a better performance than Culotta and Sorensen, 2004 on ACE corpus. Furthermore, although supervised learning has been reported by [61] for PPI extraction, only preliminary pattern induction has been implemented, which is basically corpus statistics on POS patterns without any pattern generation to cover new similar patterns which are not available in corpus.

In [60] they used sentence classification approach for sub-cellular location relations. It’s not suitable for PPI extraction, since there may be more than one PPI and judgment needed when there’re more than two proteins existing in a sentence. On the other hand, Marcotte EM, et al 2001’s supervised learning text classification can only decide PPI information which is only mentioned in the text without the extraction function. Palakal M, et al, 2002 only use HMM to decide the direction of PPI provided, which is a much simpler task than PPI extraction itself.

In [61] they have proposed a supervised learning approach to extract protein-protein interaction using *Maximum Entropy (ME)* from the output of a shallow parser. This model achieves promising performance of a 90.9 F-score, 93.9% recall and 88.0% precision on IEPA corpus provided. This method overcomes the limitation of the state-of-the-art co-occurrence based and rule-based approaches. It incorporates corpus statistics of various lexical, syntactic and semantic features. They find that the use of shallow lexical features contributes a large portion of performance improvements in contrast to the use of parsing or partial parsing information. Yet such lexical features have never been used before in other PPI extraction systems. Furthermore it can be easily adapted to extract other relations among biomedical entities given in the training corpus instead of re-writing phrase pattern rules. In summary, this approach is the first systematic study of supervised learning and the first attempt of feature-based supervised learning for PPI extraction.

In [62] they propose a two-phase *machine learning-based* biological interaction extraction method. First, the system focuses on improving a recall in extracting interactions between biological entities by using a supervised interaction learning method. Second, the system removes the incorrect biological interactions by verifying the extracted results with a *Maximum Entropy (ME)* classification method. They obtained 53% recall and 25% precision in the first place. Despite of the low performance in the initial extraction, they could successfully verify the incorrect interactions with the ME classifier and raise the precision up to 56% with tolerable degrading of the recall from 53% to 48%. Accordingly, this system splits compound or complex sentences into simple sentences with a syntactic parser, and then this system transfers those simple sentences to finite LSP sequences for more efficient process. Finally this system uses full articles, instead of abstracts, to extract more detailed information using more rich contextual information. Because the syntactic structure and expression styles are different among these four articles, some performance has been lost in a cross-validation test.

3.2 Most common PPI's Extraction Systems

In the following section we will introduce some of the most popular protein-protein interaction extraction systems.

3.2.1 PIES, a Protein Interaction Extraction System

The Protein Interaction Extraction System (PIES) [63] aims to automate a large portion of the tasks of extracting, manipulating, managing, and visualizing protein interaction pathways. PIES[6] is constructed on top of three main technologies: Kleisli, BioNLP, and Graphviz. Kleisli is a broad-scale data integration system that is used for downloading Medline abstracts and for general manipulation and management of pathway/interaction databases. BioNLP is a natural language-based information extraction system. Graphviz is a graphical layout package developed for directed graphs that we use for visualization of the extracted pathways. PIES can be augmented with various means for extracting protein interaction information from sequence databases, for example, by using Kleisli's power to integrate sequence comparison tools to detect gene fusion events in sequence databases.

PIES uses pattern matching rules to determine actor-patient roles in order to determine which protein plays the role of the "actor" (or subject) and which protein plays the role of the "patient" (or object) in the interaction mentioned in this sentence. It is worth stressing that PIES extracts the direction of interactions; that is, who inhibits whom and who activates whom. This level of information is in contrast to co-occurrence-based methods that simply say two proteins interact but without giving the direction of the interaction.

PIES automates the task of creating and visualizing pathways on-the-fly, as well as supports sophisticated large-scale manipulations of pathways including automatic integration of interaction pathway databases. It is fully operational and web access can be arranged on a case-by-case basis with the author. PIES has good functionalities when it comes to manipulation of pathways. However, currently it does not support explicit annotations by the user on individual protein interaction. Such annotations are a useful addition to the evidence sentences extracted automatically by PIES. It would be useful for PIES to support this.

[6] http://www.comp.nus.edu.sg/~wongls/talks/psb01-talk/

Currently, this support is only an indirect one. User can then add his annotations and re-export the annotation database back to PIES. The textual display of PIES does provide information on the context of the extracted information in the form of evidence sentences and links to the original Medline abstracts. However, the graphical display does not provide this information. Further research should be carried out on the graphical presentation of the context information in a visually appealing and explicit manner. Finally, the BioNLP module currently specializes in extracting inhibit vs. activate type of interactions. While its specificity on abstracts discussing this type of interaction appears high, a formal accuracy study remains to be done.

3.2.2 MedScan

MedScan[7] is a completely automated natural language processing-based information extraction system, which interprets these semantic structures using a pathway-oriented ontology and extracts protein function information. The NLP module deals with the domain-independent sentence structure decomposition, while the information extraction module can be reconfigured towards different tasks [64]. NLP component of a MedScan is a biomedical domain oriented NLP engine that processes sentences from MEDLINE abstracts and produces a set of semantic structures representing the meaning of each sentence [65]. It is based on a *context-free grammar* and a lexicon developed specifically for MEDLINE. Processing is done in two steps.

First, a syntactic parser constructs a set of alternative syntactic structures of an input sentence. Since syntactic knowledge is ambiguous in its nature, a single sentence usually yields many alternative parses. Next, semantic processor transforms each of them into a corresponding semantic tree. In MedScan, information extraction is controlled by a set of explicit declarative rules that

[7] http://www.ariadnegenomics.com/products/pathway-studio/medscan/

specify which parts of an input semantic tree should be taken into consideration and what information should be retrieved. The MedScan information extraction mechanism follows the input tree structure in a top-down manner, applying a set of context-free and context dependent *transformation rules*.

MedScan is used to extract 2976 interactions between human proteins from 3.5 million sentences from MEDLINE abstracts dated after 1988. The precision of the extracted information was found to be 91%. Comparison with the existing protein interaction databases BIND[8] and DIP[9] revealed that 96% of extracted information is novel. The recall rate of MedScan was found to be 21%. MedScan is a high precision information extraction system capable of extracting various types of protein function information encoded in a form of extendable ontology. Utilization of ontology provides an ability to change the scope of extracted information, making the entire system more flexible, and along with high performance, favorably differentiates it from the other systems.

The context free grammar is hard to construct and to manipulate. This system needs a large amount of memory hence for lexicon or ontology and so on. However, the volume of data can be increased several times by implementing a reasonable set of improvements to the system, extending the ontology towards the description of experimental data and application of logical inference methods in order to convert the experimental result into the protein function information. It might still represent a considerable interest to the users of the technology. They therefore envision two major goals of further improvement of MedScan: the improvement of the NLP grammar and the enrichment of the ontological rules to include some of the information presented in a form of raw experimental data.

[8] http://bond.unleashedinformatics.com/
[9] http://dip.doe-mbi.ucla.edu/

3.2.3 PreBIND and Textomy

PreBIND and Textomy [66] is an information extraction system that was designed to locate protein-protein interaction data in the literature and present these data to curators and the public for review and entry into BIND database. Its approach hypothesizes that the formidable task-size of backfilling the database could be reduced by using Support Vector Machine technology to first locate interaction information in the literature. PreBIND and Textomy are two components of a literature mining system designed to find protein-protein interaction information and present this to curators or public users for review and submission to the BIND database. PreBIND and Textomy differ from other methods by a combination of four factors:-

1. Support Vector Machine (SVM) technology is used to identify articles about bimolecular interactions and confirm sentences that mention specific protein-protein interactions.
2. Protein names and their gene-symbols are derived from a non-redundant sequence database.
3. This information extraction (IE) system is coupled to a human-reviewed data-entry queue for a publicly available bimolecular interaction database (BIND).
4. PreBIND and Textomy allows user feedback into the SVM training set that can constantly improve the performance of the system's ability to detect abstracts that describe bimolecular interactions.

It provides a reasonable classifier for finding interaction data in the over 14 million PubMed abstracts that are available to us. The SVM method performed better than a naïve-Bayesian classifier. SVM is a statistical approach, which appears to perform well in recognition and classification of phrases, without focusing on actual meaning. Cross-validation estimated the support vector machine's test-set precision, accuracy and recall for classifying abstracts describing interaction information as 92%, 90% and 92% respectively.

The system would be able to recall up to 60% of all non-high throughput interactions present in another yeast-protein interaction database. The system was applied to a real-world curation problem and its use was found to reduce the task duration by 70% thus saving 176 days. Machine learning methods are useful as tools to direct interaction and pathway database backfilling; however, this potential can only be realized if these techniques are coupled with human review and entry into a factual database such as BIND. Backfilling interaction data from the biomedical literature is an ongoing task that will not be completed for some time.

3.2.4 BioRAT

BioRAT[10] is a new information extraction tool, specifically designed to perform biomedical IE, and which is able to locate and analyze both abstracts and full-length papers. BioRAT [67] is a Biological Research Assistant for text mining, and incorporates document search ability with domain-specific IE. BioRAT can be regarded as a research assistant that is given a query and, autonomously, finds a set of papers reads them and highlights the most relevant facts in each. BioRAT uses natural language processing techniques and domain-specific knowledge to search for patterns in documents, with the aim of identifying interesting facts. These facts can then be extracted to produce a database of information, which has a higher ‘information density’ than a pile of papers.

The heart of BioRAT is an IE engine, based on the GATE toolbox, produced at Sheffield University [68]. *GATE* is a general purpose text engineering system,

[10] http://bioinf.cs.ucl.ac.uk/biorat/

whose modular and flexible design allows us to use it to create a more specialized biological IE system. BioRAT performs as well as existing systems, when applied to abstracts; and that significantly more information is available to BioRAT through the full-length papers than via the abstracts alone. Typically, less than half of the available information is extracted from the abstract, with the majority coming from the body of each paper. However, Extra time is required to obtain the full length papers, and there are difficulties in converting them into a usable plain text format. These costs are outweighed by the fact that more than twice as much relevant information can then be extracted automatically. However, scalability of automated information extraction systems requires that all steps in the process are automated. The recall performance of BioRAT on the abstracts alone is 20%. Overall, BioRAT achieved 43% recall and over 50% precision on full-length papers.

3.2.5 GeneScene

GeneSene[11] is a toolkit that provides an overview of published literature content. They combined a linguistic parser with Concept Space, a co-occurrence based semantic net. Both techniques extract complementary biomedical relations between noun phrases from MEDLINE abstracts. The parser extracts precise and semantically rich relations from individual abstracts and prepositions as entry points into phrases in the text. The parser also recognizes coordinating conjunctions and captures negation in text, a feature usually ignored by others. Concept Space extracts relations that hold true for the collection of abstracts.

The Gene Ontology, the Human Genome Nomenclature, and the Unified Medical Language System, are also integrated in GeneScene. Currently, they

[11] http://www.genescene.org/

are used to facilitate the integration of the two relation types, and to select the more interesting and high-quality relations for presentation. GeneScene [69] fills in a set of basic templates of patterns of prepositions around verbs and nominalized verbs. It also has a set of rules for combining these templates to extract information from more complex sentences.

Genescene stores Medline abstracts relevant to several biomedical topics, e.g., AP-1, p53, yeast, together with the relations extracted from these abstracts. Cascaded finite state automata structure the relations between individual entities. The automata are based on closed-class English words and model generic relations not limited to specific words. A user study focusing on p53 literature is discussed. All MEDLINE abstracts discussing p53 were processed in Genescene. Two researchers evaluated the terms and relations from several abstracts of interest to them. The results show that the terms were precise (93%) and relevant, as were the parser relations (precision 95.5%). The Concept Space relations were more precise when selected with ontological knowledge (precision 78%) than without (60%).

Genescene provides biomedical researchers with research findings and background relations automatically extracted from text and experimental data. These provide a more detailed overview of the information available. The extracted relations were evaluated by qualified researchers and are precise. A qualitative ongoing evaluation of the current online interface indicates that this method when used to search the literature is more useful and efficient than keyword based searching. In GENIES, more complicated patterns with syntactic and semantic constraints are used [44]. GENIES even uses semantic information. However, GENIES' recall rate is low. In the above methods, patterns are hand-coded without exception. Because there are many verbs and their variants describing protein interactions, manually coding patterns for every verb and its variants is not feasible in practical applications.

The most promising candidates for a practical information extraction system are ones based on *full-sentence parsing* as they deal with the structure of an entire sentence and therefore are potentially more accurate. An example of such a system is *GENIES* [44], which utilizes a parser and a semantic grammar consisting of a large set of nested semantic patterns (incorporating some syntactic knowledge) reflecting most frequently used sentence structures. Unlike other systems, GENIES is capable of extracting a wide variety of different relations between biological molecules as well as nested chains of relations. However, the downside of the semantic grammar-based systems like GENIES is that they may require complete redesign of the grammar in order to be tuned to a different domain.

3.2.6Link Grammar Parser – Based systems

In the last few years, natural language processing has become a rapidly-expanding field within bioinformatics, as the literature keeps growing exponentially [70] beyond the ability of human researchers to keep track of, at least without computer assistance. Natural language processing techniques rely on syntactic and semantic knowledge that is often manually encoded for a particular domain. Initially NLP is used for machine translation, speech recognition and also knowledge representation. NLP-based methods perform a substantial amount of sentence parsing to decompose the text into a structure from which relationships can be readily extracted.

Many natural language processing approaches at various complexity levels have been used successfully to extract various classes of data from biological texts, including protein-protein interactions. Recently, extraction systems have also used *Link Grammar* to identify interactions between proteins. Their approach relies on various linkage paths between named entities such as the gene and protein names. Ding et al. proposed an interaction extraction method based on Link Grammar Parser [71]. They made a great leap in biomedical information extraction area because link grammar itself is a robust and powerful framework. It can handle lots of irregularities and attempt to interpret sentences even when they are ungrammatical or contain some unknown words. However, their work is limited to counting the length of link paths only, neglecting the abundant grammatical information along the paths. In fact, the grammatical information is most valuable for interaction extraction. Basically, we cannot extract accurate information of interactions until the grammatical information is exhaustively exploited.

3.2.6 .1 ProtExt

The ProtExt system [72] extends the idea of Ding et al., 2003. They proposed a novel template language (PETL) for extracting protein-protein interactions. Their system extract protein-protein interactions embedded in sentences more accurately and customizable. It produces satisfactory results and the template language can be further extended to extract regulation of biological pathways. Their information extraction approach relies on the matching of *pre-specified templates* (patterns) or rules. The underlying assumption is that sentences conforming exactly to a pattern or a rule express the predefined relationship(s) between the sentence entities. They didn't report the values of Precision and Recall. They need to consider a template optimizer to speed the matching which can pack numerous templates into one template using a more sophisticated data structure. Manually writing patterns for every verb is not practical for general purpose applications.

3.2.6.2 IntEx

The IntEx [73] system splits complex sentences into simple clausal structures made up of syntactic roles. Their extraction system handles complex sentences and extracts multiple and nested interactions specified in a sentence. IntEx system achieves better performance without the labor intensive pattern engineering requirement. However, researchers are also interested in contextual information such as the location and agents for the interaction and the signaling pathways of which these interactions are a part. They don't extract the detailed contextual attributes (such as bio-chemical context or location) of interactions might give extra information to the biologist. They don't identify the relationships among interactions extracted from a collection of sentences (such as one interaction stimulating or inhibiting another) to construct “Protein Interaction Pathways” from abstracts and full text articles. They didn't Attempt to improve the parse output of the Link Grammar System by augmenting the dictionaries of the Link Grammar Parser with medical terms with their linking requirements. Every paper evaluates on a different test set, and so it is quite difficult to compare systems. The comparison between the Precision and Recall of IntEx and our system (PIELG) will be present in chapter 8.

3.2.6.3 BioPPIExtractor

The BioPPIExtractor system [74] applies Conditional Random Fields model to tag protein names in biomedical text, then uses a Link Grammar Parser to extracts complete interactions. Their main aim is to introduce CRFs-based protein name recognition method and evaluate its contribution to the overall protein – protein interaction performance. Their experimental results show that introduction of this method indeed helps to improve the PPI performance. However, the recall errors of BioPPIExtractor are due to the complicity of the protein interaction expression so they faced a difficulty to compile the appropriate extraction rules and, therefore, many interactions are missed out. The leading cause of precision errors of BioPPIExtractor is the nonexistence of not perfect extraction rules.

3.3 Conclusions

We have surveyed the prominent methods used for information extraction. We have demonstrated its application in the context of biomedical literature mining and protein-protein interactions. Several have shown that template and simple rule based algorithms can be used to recognize interactions achieving high rates of recall and precision ([28]; [75]). However, this technique has been found to be overall limited in the set of interactions that can be extracted by the extent of the recognition rules that are implemented, and also by the complexity of sentences being processed. Specifically, complicated cases such as interaction descriptions that span several sentences of text are often missed by these approaches. Others have addressed the issue of complex sentence structures and some limited work has been done on extracting interactions spanning several sentences through the use of parts of speech analysis [57], and natural language based approaches [78].These approaches, like the rule-based systems, have also reported high levels of recall and precision.

The protein-protein interaction extraction is a relation extraction task. In the relation extraction with news domain, some work has also been reported. In [79] they utilize a kernel-based classification approach to extract relations by computing kernel functions between parse trees. In [80] they use a similar approach as [81] method and further extend it to estimate kernel functions between augmented dependency trees. Due to the computation complexity, speed is still a serious problem for kernel approaches to be used in practical applications. The abundance of biomedical literature motivates an intensive pursuit for effective text-mining tools. Such tools are expected to help uncover the information present in the large and unstructured body of text, while addressing three main problems:

- The sheer magnitude of the available text collections;
- The ambiguity and non-uniformity of the nomenclature used in the context of genomics and proteomics.
- The linguistic-complexity of the scientific documents, stemming from the diversity of the authors in terms of expertise, style, and native languages.

As literature mining challenges in the context of bioinformatics vary widely in aspects such as scope, data sources, and ultimate goals, no single tool can currently perform all the required tasks. However, a combination of methods is likely to address many of the problems. To successfully mine the biomedical literature, it is important to realize the merits and the limitations of the different literature-mining methods. Moreover, it is essential to coherently state the actual biomedical problems we expect to address by using such methods.

Most of the previous mentioned biomedical information extraction systems focus on verbs which represent target events by themselves (i.e. “activate”), there are many cases that combinations of verbs, prepositions and certain nouns form proper IE forms. PIELG investigates and classifies forms which are needed to extract interacting protein pairs to see what forms are required in addition to ones that consist of only one verb. PIELG coves many linguistic variations of the interaction words in various contexts. The system covers nine classes based on constituents of the verbs including the nominal form as shown latter. Also, PIELG success to extract of detailed contextual attributes of interactions by interpreting modifiers like: location/position modifiers (*in, at, on),* agent/accompaniment modifiers (*by, with*), purpose modifiers (*for*, and theme/association modifiers (*of*).

CHAPTER 4

SYSTEM ARCHITECTURE

The structure or architecture of most Information Extraction systems has a common theme as described early in chapter 3. This theme is probably due to the nature of the problem, but also a result of the use of common components and building blocks, such as linguistic parsers. PIELG System extracts protein-protein interactions from biomedical text. Our Information Extraction system is organized in cascaded modules such that the output of one module is the input of the next module.

A typical session in using PIELG involves the user providing an initial search specification (keywords). The keywords may be one protein name or pairs of protein names wanted to detect their interaction properties. Then PIELG downloads PubMed abstracts satisfying that specification. Each abstract is analyzed to identify sentences that mention interaction of proteins. These sentence clauses are then processed to obtain the interactions between proteins using syntactic roles of the sentence and their linguistically significant combinations.

The *actor* and *patient* of each interaction are identified. These interaction evidence sentences are then grouped by actor and patient. Then PIELG extracts interaction information from abstracts and titles of scientific papers, and presents the extracted information in textual forms. PIELG is purely implemented with Perl under Linux platform. The architecture of the PIELG system is shown in Figure (4-1). The following sections briefly explain the workings of its modules.

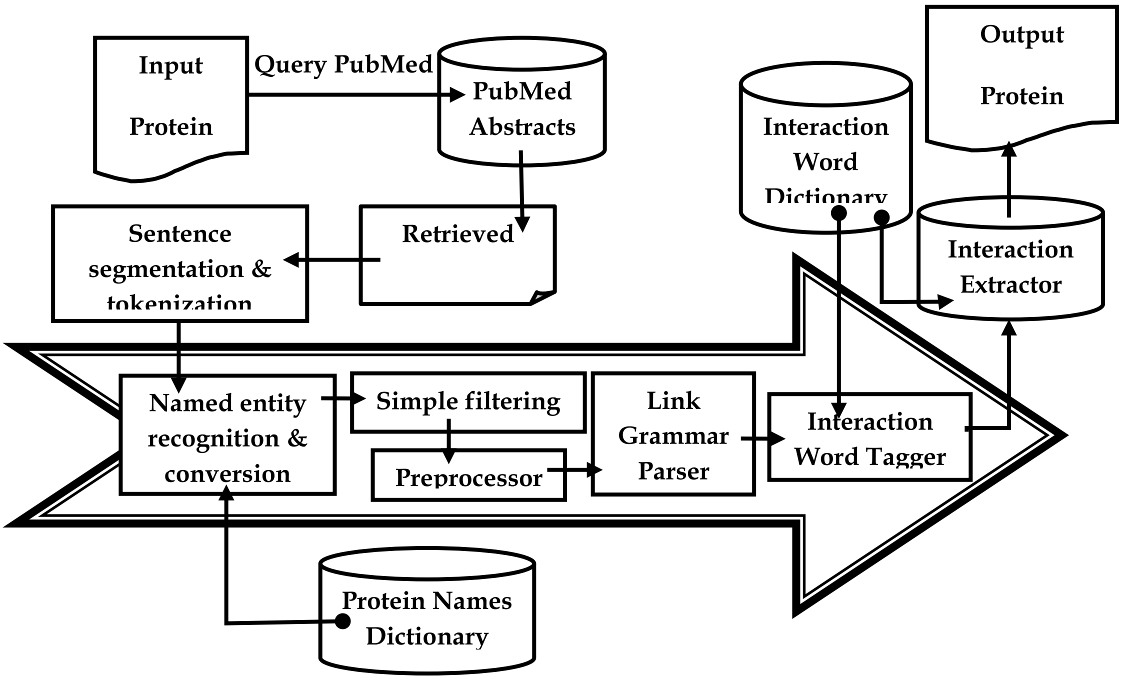


Figure (4-1): PIELG System Architecture.

4.1 Sentence Segmentation and Tokenization

The first phase of an Information Extraction system is usually the lexical analysis, which consists of dividing the input text into sentences and tokens, i.e. tokenization, and doing a lexical analysis. Each token represents the smallest linguistic unit; it can be a word (e.g. "run"), a numeric expression (e.g. "21st"). The lexical analysis is usually based on the use of morphological analyzers. In English, the morphological analysis can be based simply on the use of a word list.

PIELG system for extracting interactions requires sentence segmentation since only the proteins within a sentence are considered when identifying interactions. This module identifies sentence and word boundaries. It splits the retrieved abstracts into sentences including titles of each paper. The title of a paper may include important information like the title of this paper: - *Dentin matrix protein-1 regulates dentin sialophosphoprotein gene transcription during early odontoblast differentiation.* This is done by using simple regular expressions, to identify sentence boundaries, assuming any period followed by a space and an uppercase letter is a sentence boundary. The word and sentence segmentation step is simplified. The result of the morphological analysis is usually basic linguistic features. Tokens can be also tagged for other information. For example, in the context of bioinformatics, a token could receive a tag identifying it as a biomedical term that could be the part of gene or protein name.

4.2 Named Entity identification and conversion

Most efforts concerned with biomedical literature mining to date focus on automated *information extraction*, using crated lexica for identifying relevant phrases and facts in text. Named entity identification or Entity extraction is the process of identifying protein names in the text. The simplest and frequently used approach to entity identification is a dictionary matching approach. Entity names are compiled as a dictionary. A string match with an entry in the dictionary tags the words or phrases as protein names. A variety of publicly available databases provide the resources for entity names.

Some of the major current sources for gene-related terms: genome and proteome databases such as LocusLink[12] , UniProt[13], and the HUGO[14] gene nomenclature contain many of the names and synonyms denoting known genes in various organisms. These databases provide gene and protein names and their synonyms. PIELG distills its dictionary of protein names from EXpaxy[15] and iHOP[16] databases. The dictionary of PIELG carries about 1000 entries. However, we do not do any synonym grouping or name clustering. Since our main goal is aimed at proposing a method for extracting protein-protein interactions, the current named entity recognizer is sufficient for this purpose.

Named entity conversion process is important for entity extraction. It is the process of converting each protein name into a *personal name*. Before conversion we need to make sure that each protein name has one identical representation. It is noticed that a protein name may have different appearances and lots of identical representations. For example, the protein name *Dentin matrix protein-1* may appear as *Dentin matrix protein 1*. Also, its abbreviation may appear in the text as *DMP-1* or *DMP 1*. This module tries to normalize protein names using a dictionary so that different names of the same protein are mapped to a standard name.

The conversion process aimed to get the Link Grammar Parser handles texts with protein names of multiple words. This is done by converting each protein

[12] http://www.ncbi.nlm.nih.gov/sites/ entrez?db=gene.

[13] http://beta.uniprot.org.

[14] http://www.genenames.org/

[15] http://www.expasy.ch.

[16] http://www.ihop-net.org.

name into a *personal name*. This is necessary because link parser does not have an unbounded dictionary which may hold the vocabulary of all protein names. Common personal names are already known to the Link Grammar parser and doing this can prevent it from guessing the biochemical names. For example, *Bone morphogenetic proteins* will be replaced by *BMPs* and *Electron probe micro-analysist* will be replaced by *EPMA*. If we do not do the conversion, then perhaps few sentences can be well parsed by the parser. Besides, doing this usually can reduce the number of words in sentences, which is helpful to processing. This will reduce the total processing time of the total system. If we take the following sentence as an example *Dentin matrix protein-1 is verified by real-time reverse transcruption-polymerase chain reaction* it will be converted to *DMP-1 is verified by real-time RT-PCR*.

4.3 Simple Filtering and Transformation

Simple filtering is the process used to reduce the processing time for an abstract. It filters out sentences that do not contain any interactions. Sentences are again searched for the protein pairs. The sentences that contain at least two protein names are alone chosen for processing.

The *Transformation* process is needed to make the Link parser able to handle text with some expressions including protein names. The expressions of multiple words properly would have required a wrapper around the parser. This wrapper is the transformer that will transfer those expressions into *personal names* from the text before passing it to the parser. The re-transformer is then inserted in the name back after parsing, for example, *gene expression of Alp* and *expression of the transcription factor RUNX2*. Besides, doing this usually can reduce the number of words in sentences, which is helpful to processing. This will reduce the total processing time of the total system.

4.4 Preprocessor

The details of preprocessing vary from one system to another but certain steps are considered by all system designers. Preprocessor allows removing of numerous structure ambiguities, which clearly benefits the parsing quality and execution time. The tagged sentences need to be pre-processed to replace syntactic constructs, such as parenthesized nouns and domain specific terminology that cause the parser to produce an incorrect output. This problem is overcome by replacing such elements with alternative formats that is recognizable by the parser. The preprocessor forces the Link Grammar parser to recognize the biological names as noun forms. Since the parser recognizes words that start with an uppercase letter as a noun therefore, the pre-processor converts each protein personal name to a word starting with an uppercase letter.

The parser is also not designed for parentheses in the sentences. The sentences in the abstracts are analyzed, and it is found that the text inside parentheses often referred to alias names of the entities mentioned. So, the words in the parentheses are removed to improve the parse output as they provide no additional information in many sentences. However, there is some loss of information regarding the interactions due to this process which bring down the recall of the extraction system.

The pre-processor performs minor punctuation corrections on the spacing of commas and semi-colons in the text. It filters out some adverbs such as *however*, *hence*, *also*, *furthermore* etc. The preprocessor removes some information that is unrelated to biochemical interactions, such as a window of time: (1994-2008), probabilities, mathematical notations: (p _ 0.03), special characters, and so forth. The rationale of doing this is that it can save some computational effort during parsing without losing crucial information related to interactions and make sentences more understandable to Link Grammar Parser. The tagged sentences need to be preprocessed to replace domain specific terminology that causes the Link Grammar Parser to produce an incorrect output. This problem is overcome by replacing such elements with alternative formats that are recognizable by the parser.

4.5 Link Grammar Parser and Link Grammar

Link grammar (LG) introduced by Sleator and Temperley [82] is a dependency-based grammatical system. The Link Grammar Parser is a syntactic parser of English based on link grammar, an original theory of English syntax. The basic idea of link grammar is to connect pairs of words in a sentence with various syntactically significant links. Given a sentence, the system assigns to it a syntactic structure, which consists of a set of labeled links connecting pairs of words. The Link parser is freely available from the internet. While the parser with full source code could be downloaded from the Link parser homepage, there was no clear statement of the license under which it could be used commercially. However, since no notion of this could be seen on the official homepage, the use of the parser under such obscure promise of a license was rejected.

Link Grammar Parser is available as ANSI C program and Perl module. As most of PIELG system was coded in Perl language, a wrapper was written for

LG Parser in Perl to handle its input and output. The wrapper overrides the input and output buffers of LG parser. The Perl module Lingu::LinkParser runs and the wrapper feeds the input buffer with the sentence and collects the output parse from the output buffer. Then the output is parsed using regular expressions to extract the linkages, and these linkages can be accessed through the wrapper's Perl API. A detailed description of this module is covered in Chapter 5.

4.6 Interaction Word Tagger

Once protein names have been found, the relationships between them need to be ascertained. The words that convey a biologically significant action between two protein names are labeled as *interaction words*. For example in sentence ‘‘DMP-1 regulates DSPP during early odontoblast differentiation**.’’**, the main verb “regulate”, describes the action performed by “DMP-1” on “DSPP”, is an example of interaction word. Some other example of interaction words are “bind”, “down-regulation”, “phosphrylation”, *bind, associate* and *complex* etc. This can be done in a number of ways depending on the Information Extraction (IE) task. The system uses dictionary look-up method to identify interaction words in the sentences.

We use a category/keyword dictionary for identifying terms describing interactions. The category/keyword dictionary is adapted from [44] with additional categories and keywords found to be prevalent in our corpus. A list of interaction words, which consists of 45 noun and 53 verb roots, was compiled from the literature. In order to broaden the list of potential interaction words, all inflected variants of known interaction words are also considered. Further, also all predictable spelling and derivational variants are considered.

Table (4-1) Direct and Indirect Interaction Words

Direct interaction verbs	**Indirect interaction verbs**
bind (bound)	induc(-es,-ed)
interact (-s,-ed)	trigger(-s,-ed)
stabilize (-s,-d)	block(s),
phosphorylate(-s,-d)	enhance(s)
ubiquinate(-s,-d)	synergize(s)
sumoylate(-s,-d)	cooperate(s)
degrade(-s,-d)	localizes
block(s).	regul(-ates,-ion)
	activate(s)
	inhibit(s)
	control(s)
	translocate(s)
	antagonize(s)
	amplif(-y,-ies)
	transduce(s)
	degrade(s)
	trigger(s).

The dictionary is enriched manually with additional verbs that are known to refer to interactions. The *direct* and *indirect* physical interaction words are split into as shown in Table (4-1).

Example If the word *labeled* appears in the corpus as an interaction word, we also consider the words *label, labels, labeling, labeled* to be potential interaction words. Similarly, for the word *rebinds* we also consider the words *re-binds, rebind, re-bind, rebound, re-bound, rebinding, rebinding*. Table (4-2) shows some examples of interaction words.

Table (4-2): Examples of interaction words.

Category	**Keywords**	**Category**	**Keywords**	**Category**	**Keywords**
Activate	accumulat (e,ed,es,ion) activat (e,ed,es,or, ion) elevat (e,ed,es,ion) hasten (ed,es) Incite (ed,es) increas (ed,es) Induc (e,ed,es,tion) promot (e,ed,es) stimulat (e,ed,or,ion) transactivat	**Break Bond**	cleav (e,ed,es) demethylat (e,ed,es,ation) Dephosphorylat (e,ed,es,ation) sever (e,ed,es)	**Inactivate**	inhibit (s,ed,ion) reduc (e,ed,es,tion) repress (ed,es,ion) supress (ed,es,ion)
		Cause	influenc (e,ed,es)	**Modify**	modifi (ed,cation)
		Contain	contain (s,ed,es)	**Process**	apoptosis myogenesis
		Create Bond	methylat (e,ed,es,ation		

	(e,ed,es,ion) up-regulat (e,ed,es,or,ion) Upregulat (e,ed,es,or)		) phosphorylat (e,ed,es,ation)		
Association	associat (e,ed,es,ion)				
				Release	disassembl (e,es,ed) discharg (e,es,ed)
Attach	add (s,ition) bind (s),bound catalyz (e,ed,es) Complex				
		Generate	express (ed,es,ion) overexpress (ed,es,ion) produc (e,ed,es,tion)		
				Signal	mediat (e,ed,es) modulat (e,ed,es,ion) participat (e,ed,es,ion) regulat (e,es,ed,ion)
		Inactivate	block (s,ed) decreas (e,ed,es) deplet (e,ed,es,ion) down-regulat (e,ed,es,ion) down-regulat (e,ed,es,ion) impair (s,ed) inactivat (e,ed,es,ion)		
				Substititute	replac (e,ed,es) substitut (e,ed,es,ion)

4.7 Interaction Extractor (IE)

Interaction Extractor (IE) extracts interactions from simple sentence clauses produced by the Link Grammar parser. Sentences are made up of different syntactic constituents like Noun Phrases, Verb phrase, Modifying Phrases etc. Each of this syntactic "roles" has some meaning and context in the sentence they are talked about. Each of these constituent plays a role (e.g. subject of main verb) based on the theme or the event the sentence is talking about. To keep the Interaction Extractor as generic as possible, we used only three basic constituents types based on the 'roles' they play:

Subject - subject of main verb.

Object - one or more object of main verb for the given subject.

Modifying Phrase (MVP) - One of more MVP for both subject and object.

Given these syntactic constituents we identity the roles based on the information they contain. For example in sentence ''DMP-1 regulates DSPP during early odontoblast differentiation.'' subject "DMP-1" contains one protein name, object "DSPP" contains one protein name, and modifying phrase "during early odontoblast differentiation" contains one protein name. Detailed description of this module is covered in Chapter 6. The highly technical terminology and the complex grammatical constructs that are present in the biomedical abstracts make the extraction task difficult, even a simple sentence with a single verb can contain multiple and/or nested interactions.

For example: "Phosphophoryn signals DSPP by directly stimulating DMP-1. Here the sentence has two interactions "Phosphophoryn, signals, DSPP" and "Phosphophoryn, stimulating, DMP-1". That’s why our IE system is based on a deep parse tree structure presented by the LG and it considers a thorough case based analysis of contents of various syntactic roles of the sentences. Detailed description of this module is covered in Chapter 6.

CHAPTER 5

LINK GRAMMAR

Many approaches to NLP have been pursued in the past few decades, but few are as popular as the Link Grammar parser. Link grammar (LG) is an original theory of English syntax. It was written by Davy Temperley, Daniel Sleator, and John Lafferty of Carnegie Mellon University [81] to simplify English grammar with a context-free grammar. Link grammar [82] is a theory of syntax which builds relations between pairs of words, rather than constructing constituents in a tree-like hierarchy. Link grammar is a dependency based grammatical system; its basic idea is to connect pairs of words in a sentence with various syntactically significant links. Rather than examining the basic context of a word within a sentence, the Link Grammar is based on a model that words within a text form "links" with one another. It considers words as blocks with connectors coming out. There are different types of connectors and may point to the right or to the left. A left-pointing connector connects with a right-pointing connector of the same type on another word. The two connectors together form a *link*.

The link grammar consists of sets of words, each of which has a linking requirement. This linking requirement can be seen as a block with connectors above each word. A connector is satisfied by matching it with a compatible connector. In Link Grammar, a *linkage* is a single successful parse of a sentence: a set of links in which none of the connecting arcs cross. The words of a syntactic structure are connected in such way. The links satisfy the linking requirements for each word of the sentence (satisfaction). The links do not cross and all words form a connected graph. These links are used not only to identify parts of speech (nouns, verbs, and so on), but also to describe in detail the *function* of the word within the sentence. If a phrase consists of two adjectives and two nouns you really want to know which adjective modifies which noun. The LG does that.

In Link Grammar vernacular, a *linkage* is a single successful parse of a sentence: a set of links in which none of the connecting arcs cross. The following diagram Figure (5-1) shows how linking requirement for the sentence “The dog chased a cat” is satisfied.

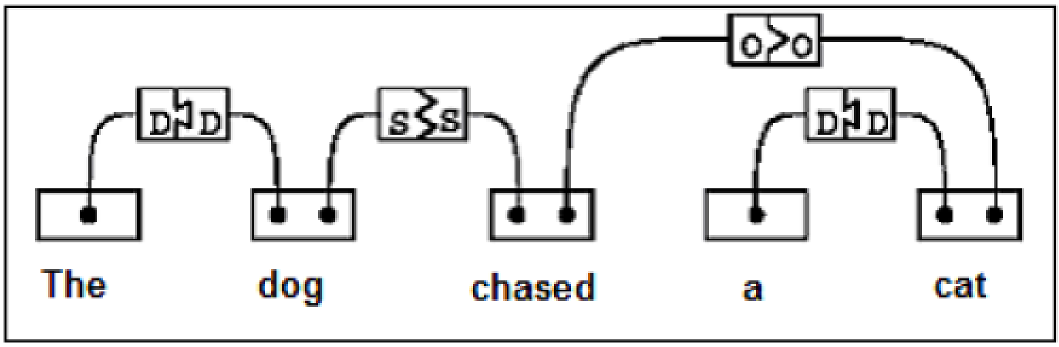


Figure (5-1): Link Grammar Representation of a Sentence

The arcs between the words are “links” and the labels show the link type. In the example below the link between “dog” and “chased” is “S” (connects Subject-noun to verbs), the link between “chased” and “cat” is “O” (connects verbs to direct or indirect Objects) and the link between “the” and “dog” is “D” (connects determiners to nouns). A sample parse of the sentence, "A camel is a horse designed by a committee" is depicted in Figure (5-2).

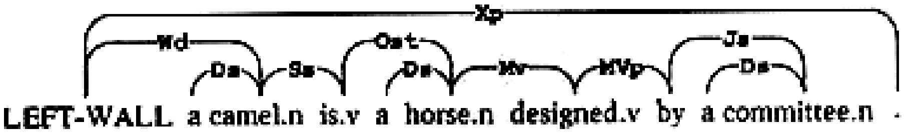

Xp
Wd
Ds
Ss
Ost
Ds
Mv
MVp
Js
Ds
LEFT-WALL a camel.n is.v a horse.n designed.v by a committee.n .

Figure (5-2): A sample parse, with links.

The primary parts of speech are labeled with .n and .v to indicate that these words are nouns and verbs, respectively. The labels of the links between words indicate the type of link. For example, the J connector in this sentence indicates a connection between prepositions and their objects; in this case, the verb designed is connected to by a committee, identifying a prepositional phrase.

Each word in the lexicon of link grammar must satisfy the linking requirements [81], which is stored in a dictionary. These requirements are specified by means of a formula of connectors combined by binary associative operators. When a link connects to a word, it is associated with one of the connectors of the formula of that word, and it is said to satisfy that connector. No two links may satisfy the same connector. A sequence of words is a sentence of the language defined by the grammar if there exists a way to draw links among the words so as to satisfy each word's formula, and the following meta-rules:

1. **Planarity:** The links are drawn above the sentence and do not cross.
2. **Connectivity:** The links suffice to connect all the words of the sequence together.
3. **Ordering:** When the connectors of a formula are traversed from left to right, the words to which they connect proceed from near to far. In other words, consider a word, and consider two links connecting that word to words to its left. The link connecting the nearer word (the shorter link) must satisfy a connector appearing to the left (in the formula) of that of the other word. Similarly, a link to the right must satisfy a connector to the left (in the formula) of a longer link to the right.
4. **Exclusion:** No two links may connect the same pair of words.

5.1 Some Important Links

The following is a list explaining the significance of some of the important linkages of the link grammar system which are used in our scheme:

- ***A and AN:*** Connects pre-noun modifiers like adjectives or nouns to the following noun. e.g. - the huge man sat there, the ***tax*** proposal is to be revised.
- ***B:*** Connects transitive verbs back to their objects in relative clauses and questions. e.g. - the man he killed, what did you eat? Also, connects the main noun to the finite verb in subject-type relative clauses. e.g. the teacher who taught me was tall.
- ***DP:*** Connects possessive determiners to gerunds in cases where the gerund is taking its normal complement. e.g. your telling Jane to leave was a mistake.
- ***I:*** Connects infinitive verb forms to certain words such as modal verbs and “to”. e.g. he has to be present, they should do their work.
- ***J:*** Connects prepositions to their objects. e.g. the man with the dog is here.
- ***M:*** Connectsnouns to various kinds of post-noun modifiers like prepositions and participles. e.g. the man with the umbrella, the lady to whom I proposed.

- ***MV:*** connects verbs and adjectives to modifying phrases that follow e.g. the man slept in the room, it was hotter yesterday.
- ***MX:*** Connects nouns to post-nominal noun modifiers surrounded by commas. e.g. the man, who killed him, was arrested.
- ***O, OD and OT:*** Connects transitive verbs to their objects, direct or indirect. e.g. he played cricket, I gave you a book.
- ***P:*** Connects forms of the verb “be” to prepositions, adjectives and participles. e.g. he is playing; the boys are in the field, she was angry.
- ***PP:*** Connects forms of “have” to past participles e.g. he has gone.
- ***R:*** Connects nouns to relative clauses.

- ***RS**:* Connects the relative pronoun to the verb. e.g. – the man **who chased us.**
- ***S, SI, SX and SXI**:* Connects subject nouns to finite verbs. e.g. - a **child likes** sweets.
- ***TO:*** Connects verbs and adjectives which take infinitival complements to the word 'to". e.g. - they **planned to** party.

5.2 The Link Grammar Parser

The Link Grammar Parser (LGP) [75] is a syntactic parser of English, based on link grammar. Given a sentence, Link Grammar Parser assigns to it a syntactic structure, which consists of a set of labeled links connecting pairs of words. These links are used not only to identify parts of speech (nouns, verbs, and so on), but also to describe in detail the *function* of that word within the sentence. LGP has around seven hundred definitions that capture many phenomena of English grammar. It can handle: noun-verb agreement, questions, imperatives, complex and irregular verbs (wanted, go, denied, etc.), different types of nouns and many other things. The dictionary of LG parser has about 60000 word forms, with wide coverage of syntactic constructions.

The parser is robust and can skip the portions of sentences it cannot understand and assign some structure to the rest of sentence. Its ability to handle unknown vocabulary is remarkable and this feature is most useful when parsing unknown alphanumeric gene/protein names and sentences which very compound and complex, as most of the sentences from the abstracts biomedical domain are. As we were looking for a dependency-tree based parser, we selected LG parser. The LG parser gives the links to extract the constituents like Subject, Object and modifier in a sentence.

The Link Grammar Parser also produces a *constituent* representation of a sentence (showing noun phrases, verb phrases, etc.). For example, in a Subject-

Verb-Object (S-V-O) language like English, the verb would look left to form a subject link, and right to form an object link. Nouns would look right to complete the subject link, or left to complete the object link. A sample parser output is depicted in Figure (5-3) for the sentence; *DGI is associated with mutations in DSPP.*

```
++++Time 0.00 seconds (51.89 total). Found 3 linkages
+------------------------------------Xp----------------------------+
|                     +----------------MVp-------------+          |
+-Wd-+-Ss-+---Pv---+---MVp--+---Jp--+       +-Js+          |
|        |    |        |          |        |        |    |        |
WALL DGI is.v associated.v with mutations.n in DSPP .
```

Figure (5-3): A sample parser output with links.

The primary parts of speech are labeled with .n and .v to indicate that these words are nouns and verbs, respectively. The labels of the links between words indicate the type of link. For example, the *Mv* connector in this sentence indicates a connection between the verb and its modifying phrase. In this case, the verb *associated* is connected to *with mutations*, identifying a modifying phrase.

The parser has an internal timer. If the timer runs down before a complete or partial linkage has been found, the parser will output whatever it has found so far (termed a fragmented linkage). Link Grammar Parser has many Applications such as: - AbiWord [83] checks, information extraction of biomedical texts and events described in news articles, as well as experimental machine translation. Another sample parser output is depicted in Figure (5-4) for the sentence; *DMP-1 regulates DSPP during odontoblast differentiation.*

```
++++Time 0.00 seconds (51.89 total).Found 1 linkage (1 with no P.P.
violations)Unique linkage.cost vector = (UNUSED=0 DIS=0 AND=0 LEN=11)
+--------------------------------------Xp----------------------------------+
|                               +-----------------Jp-----------------+     |
|                +------MVp-----+      +--------------A--------------+     |
+---Wd--+---Ss---+---Os---+     |      |            +--------A-------+     |
|       |        |        |     |      |            |                |     |
LEFT  DMP-1 regulates.v DSPP during early.a odontoblast.a differentiation.n.
```

Figure (5-4): The linkage given by the Link Grammar Parser for the sentence "DMP-1 regulates DSPP during odontoblast differentiation."

5.2.1 Link Grammar Parser's Dictionary

The parser uses a dictionary that contains the linking requirements of each word. For example, the words *the*, *chased*, *dog*, and *cat* are shown below with their linking requirements. The D within the box below in Figure (5-5) "*the*" indicates that another word must connect with D to the right of the in order for the link requirements to be satisfied for that word.

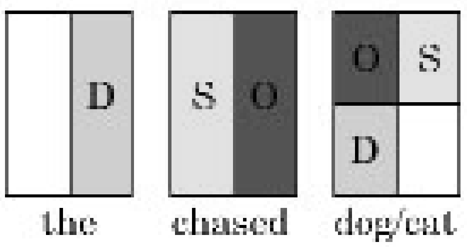


Figure (5-5): Some linking requirements

For these words to form a sentence, the parser must find them in an order which satisfies the above three requirements. When a word has more than one row of connectors, only one side (left or right) of each row may be connected (e.g. *cat* has a row *D* and a row *O/S*, so *D* must be connected along with either *O* or *S*). When only one row exists on a single level (e.g. *cat* has *D*), one connector must be linked. The meaning of each link used here is indicated above. Thus, the following arrangement is correct: *The cat chased the dog*. The unused connectors are grayed out in this example in Figure (5-6). Since our second *"the"* connects to *dog* as a determiner, *chased* actually spans the length, connecting to *"dog"*. You can mentally shuffle these words to see that *cat* and *dog* could be swapped, and likely would be if our program had any semantic knowledge. Moving other words around, however, will break the link criterion and deem the parse ungrammatical.

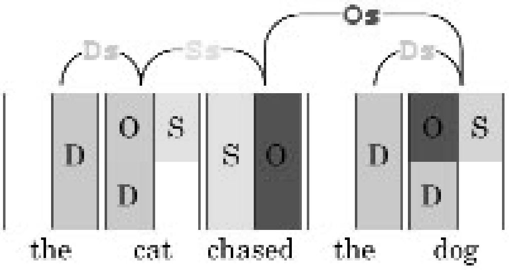


Figure (5-6): Linking requirements and inferred links.

All of these requirements are stored in the Link Parser's dictionary files. The files use a "link dictionary language" to list the requirements for each word, and are themselves an interesting study in pattern representation. A highly optimized custom algorithm processes the data in these files, analyzing the possible links. This algorithm is yet another fascinating study in and of itself. Because the researchers at CMU had the generosity and intelligence to make their project research open to developers like us, we can examine the ingenuity of their methods. We can use and modify their well-conceived Application Program Interface (API). We can extend and combine the functionality of their system with that of other language processing technologies. And, of course, Perl makes it all possible, practical, and inevitable.

The parser uses a dictionary that contains the linking requirements of each word and the possible part of speech assignments for the entries. It has a dictionary of about 60000 word forms. Also, it has coverage of a wide variety of syntactic constructions. The parser is robust; it is able to skip over portions of the sentence that it cannot understand, and assign some structure to the rest of the sentence. It is able to handle unknown vocabulary, and make intelligent guesses from context and spelling about the syntactic categories of unknown words. It has knowledge of capitalization, numerical expressions, and a variety of punctuation symbols.

5.2.2 LGP's Dictionary Enhancement

The word dictionaries of the Link Grammar Parser are from conversational English which do not include the biological named entities. The LG parsers' lexicon can be easily enhanced to produce better parses for biomedical text [84]. We use two methods to extending the lexicon of the Link Grammar Parser. The first method is to use the LinkGrammar-WN [85] which aims to import lexical information from WordNet. WordNet [86] is an online lexical reference system that in recent years has become a popular tool for Artificial Intelligence (AI) researchers. The LinkGrammar-WN v1.0 release contains 14,392 noun word forms not available within the original LGP lexicon, thus increasing the size of the LGP lexicon by 25%.

The second extension method is to use the extended Link Grammar Parser [87] where the lexicon is extended by the lexicon from UMLS' [88] Specialist lexicon enabled to general-purpose language processing tools. That enables Link Grammar Parser to manipulate medical text. The typically non-technical vocabularies must be augmented with a large medical lexicon. It applies a heuristic method to import lexical definitions of about 200,000 word senses into the LG dictionary, more than tripling its size from the UMLS's Specialist lexicon. This extension of Link Grammar's dictionary [89] effects on its performance. This extension can significantly improve efficiency, parsing performance and significantly reduced ambiguity. The extended parser manipulates biomedical text well.

5.3 Lingua::LinkParser

The Link Grammar Parser itself is a complex piece of software implementing a complex theory of language. The Perl module Lingua::LinkParser [90] directly embeds the parser API, providing an object-oriented interface that you can use from your Perl programs. Objects may be created to represent sentences, linkages, links, individual words, and the parser itself. The PIELG system uses the Perl module Lingua::LinkParser [90]. It is a Perl module implementing the Link Grammar Parser under Linux platform. This module is available at CPAN [91] directly, embeds the parser. As an example, consider the following code:

```
use Lingua::LinkParser;
use strict;

my $parser = new Lingua::LinkParser;        # create the parser
my $text   = "Moses supposes his toses are roses.";

my $sentence = $parser->create_sentence($text); # parse the sentence
my $linkage  = $sentence->linkage(1);        # use the first linkage

print $parser->get_diagram($linkage);         # print it out
```

This code will output as shown in Figure (5-7).

```
    +-------------------------Xp--------------------------------------+
    |                +------Ce------+                                 |
    +-----Wd-----+---Ss---+            +--Dmc--+-Spx-+-Opt-+    |
    |                 |           |             |          |         |        |    |
LEFT-WALL Moses supposes.v his toses[!].n are.v roses.n .
```

Figure (5-7): The output of the code

Without delving into all the details, this diagram reveals some interesting things about the parser. First, *supposes* and *are* have *v* labels, indicating that they're verbs. The word *"roses"* is labeled n for noun, as is *"toses"*. The [!] tag next to *"toses*" indicates that the parser isn't familiar with this word, which usually means that it isn't a word at all. So even with a word it's never seen before, the Link Grammar can identify the part of speech.

5.4 The Link Parser Application Program Interface (API)

The original version of the parser was designed around a standard interface, where the user types in a sentence, and the parser displays the linkages that it finds. This is fine for showing how the grammar and parser work, but in order to make actual *use* of the information that the parser provides, it is necessary to have access to its inner workings.

The Link Parser Application Program Interface (API) [92] was written to give users flexibility in using the parser in their applications. The Lingua::LinkParser module provides access to the parser API using Perl objects to easily analyze linkages. The API makes it easy to incorporate the parser into other applications. The API provides a set of basic data structures and function calls that allow the programmer to easily design a customized parser. The module organizes data returned from the parser API into an object hierarchy consisting of, in order, sentence, linkage, sub-linkage, and link.

Examples of the kind of capability the API provides include:

- Open up more than one dictionary at a time.
- Parse a sentence with different dictionaries or parsing parameters, and compare the results.
- Limit the time and memory that the parsing process takes.

- Use different "cost functions" for ranking linkages.
- Save linkages from a sentence, and access individual links.
- Post-process a sentence with more than one set of post-processing rules.
- Extract the domains that links participate in, to perform transformations on a linkage.
- Recover the constituent structure corresponding to a phrase-structure grammar.

The API provides a set of basic data structures and function calls that allow the programmer to easily design a customized parser. The API is written in ANSI C, and runs in both Linux and Windows environments. The following Example helps us to understand the Link Grammar API. To use the information within a program requires access to the links themselves. Continuing with the program a**bove**, we will extract from the *$linkage* object an array of *$word* objects. These will spring into existence, along with a *links()* method to return an array of *$link* objects. Well, just watch:

```
my @words = $linkage->words;
   foreach my $word (@words) {
      print "\"", $word->text, "\"\n";
      foreach my $link ($word->links) {
         print "  link type '", $link->linklabel,
           "' to word '", $link->linkword, "'\n";
      }
   }
```

An excerpt from the output:

```
  "Moses"
    link type 'Wd' to word '0:LEFT-WALL'
    link type 'Ss' to word '2:supposes.v'

  "supposes.v"
    link type 'Ss' to word '1:Moses'
    link type 'Ce' to word '4:toses[!]'

  "his"
    link type 'Dmc' to word '4:toses[!]'

  "toses[!]"
    link type 'Ce' to word '2:supposes.v'
    link type 'Dmc' to word '3:his'
    link type 'Spx' to word '5:are.v'
```

CHAPTER 6

INTERACTION EXTRACTOR MODULE

6.1Introduction

Interaction Extractor is the main component of the PIELG system**.** The aim here is to do deep analysis of the sentence to extract multiple and nested interactions from the sentence. Our IE system is based on a deep parse tree structure presented by the Link Grammar. It considers a thorough case based analysis of contents of various syntactic roles of the sentences like their subjects (S), verbs (V), objects (O) and modifying phrases (M) as well as their linguistically significant and meaningful combinations like S-V-O or S-V-M. Each of syntactic roles has some meaning and context in the sentence they are talked about.

The sentences are made up of different syntactic constituents like Noun Phrases, Verb phrase, Modifying Phrases etc. Each of these constituent plays a role (e.g. subject of main verb) based on the theme or the event the sentence is talking about. To keep the Interaction Extractor as generic as possible, we used basic constituents types based on the ‘roles’ they play:

Subject - subject of main verb.

Object - one or more object of main verb for the given subject.

Modifying Phrase (MVP) - One of more MVP for both subject and object.

Given these syntactic constituents we identity the roles based on the information they contain. For example in sentence *‘‘DMP-1 regulates DSPP during early odontoblast differentiation.*” subject *“DMP-1”* contains one protein name, object *“DSPP*” contains one protein name, and modifying phrase *“during early odontoblast differentiation”* contains one protein name. For each syntactic role of the sentence, the role type matcher identifies the type of each role based on its matching content.

6.2 Information Extractor Algorithm

The interaction extraction scheme uses a series of mapping rules to extract information about protein- protein interactions. Those mapping rules could be applied to first identify all the main verbs. Then, determining if those verbs are truly representing the interaction between two protein names (interaction words), in the text or not. If the main verb is not an interaction word then the algorithm detects all verbs in the sentence until detecting an interaction word.

The algorithm (Algorithm 1) as shown in Table (6-1), is based on generic templates constructed using English Grammar syntax, looks into all parts of the sentence. The input to IE is the preprocessed and role typed simple clause structures. The IE algorithm (Algorithm 1) progresses bottom up, starting with each syntactic role subject, verb or modifying phrases, and expanding them using the lattice provided in until all “Complete” singleton or composite role types are obtained.

6.2.1 The main verb is an interaction word

If the main verb is an interaction word, the system applies a set of rules to predict the subject for each of these. The scheme also helps to find out the object of the verb, when present, as well as the modifiers of all verbs and nouns. The prediction scheme begins once the sentence has been passed through the link parser and the linkage for that sentence has been obtained. As the link grammar requires that no two links cross each other, no two links connect the same pair of words and all the words form one unit, the linkage structure can be represented in the form of a tree. The elements of the tree are then analyzed to first find the main verbs and then if possible, find their subjects (S), verbs (V), objects (O) and modifying phrases (M) as well as their linguistically significant and meaningful combinations like S–V–O, S–V–M. Then finding and extracting protein–protein interactions only if a syntactic role (or meaningful combination) has at least two protein names and an interaction word.

Table (6 -1): (Algorithm 1): Algorithm for Interaction Extractor

(Algorithm 1) Algorithm for Interaction Extractor

INPUT: Simple clausal Structures of the sentence.

OUTPUT: Protein-Protein Interactions, Example :(Protein1, Interaction Word, Protein 2).

1. Identifying the main verb of the sentence.
2. Using the linkage given by the Link Grammar parser, for the given sentence, obtaining the constituents Subject, Object and the Modifying Phrase (S, O and MP respectively).
3. If the main verb is an interaction word and the Subject, Object or Modifying Phrase is a protein name. Then extract interaction from the combination of Subject, Verb and Object (S-V-O) roles. Similarly extract interaction from the combination of Subject, Verb and Modifying Phrase (S-V-O) roles**.**
4. If the main verb is not an interaction word the system detects the place of the interaction word. Then the system extracts interaction from the combination of (S-V-O) and (S-V-O) roles**.** We have taken various possible cases in which interaction can occur in a sentence.
5. If the sentence contains combinations of prepositions the system uses Preposition-based patterns to find agent, theme and action to extract the interaction, for both active and passive voices.

6.2.1.1 Identifying the main verbs

The system uses the procedure proposed in [93] for identifying the main verb. The link parser itself tags the verbs of the sentence with a ‘v’ tag but all of them are not main verbs and all of them do not require subjects. Here, a main verb is considered to be the word in the verb phrase which actually represents the action done, *i.e.*, words like infinitives (e.g. - to, will), modal verbs (e.g. -

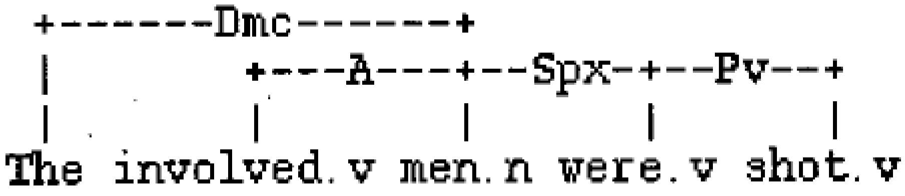
must, should) and sometimes forms of *be* are neglected. Also, verbs do not need subjects when they are acting as an adjective. In order to identify the main verbs, all the words tagged with ‘v’ are considered first. Then verbs are pruned out based on the conditions presented in [93]. After identifying the main verb, if it is marked as an interaction word from the interaction word tagger the system will continue to predict the subject and the object.

After all the main verbs have been identified, the subject, the object (if it exists) and the modifying phrases of both the verb and the object will also have to be predicted based on the rules presented in [93]. The rules are applied in hierarchical to identify the subjects (S), and objects (O) as well as all available modifying phrases (M) of the sentences. After identification, the interaction extractor algorithm progresses bottom up, starting with each syntactic role subject, verb or modifying phrases, and expanding them until all composite interaction role types are obtained. If subject, object or modifying phrase role itself is a protein name, then the system will extract interaction from the combination of subject-verb-object (*S-V-O)* or subject-verb- modifying phrase *(S-V-M).* We have taken various possible cases in which interaction can occur in a sentence. So, almost all information about protein - protein interactions could be extracted from the text, for both active and passive voices.

6.2.1.2 Rules for Verb Prediction

In order to identify the main verbs, all the words tagged with ‘v’ are considered first [93]. Then verbs are pruned out based on the following conditions:-

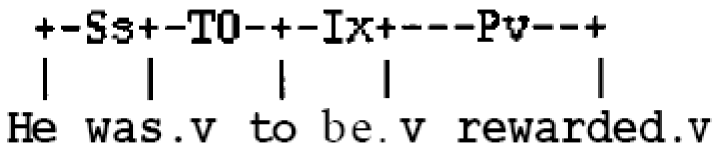
1. Verbs which make an ‘A’ link with some noun to their right or make a **‘M’** link with some noun to their left without making any other link act as adjectives and so they do not need a subject. (Refer Figure (6-1))



```
+-------Dmc-------+
|       +---A---+--Spx-+--Pv--+
|       |       |      |      |
The involved.v men.n were.v shot.v
```

Figure (6-1): Verb as adjective

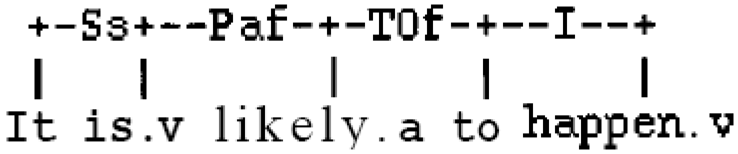
2. Infinitives, modal verbs and forms of “be”, when followed by a verb are neglected. This is done by neglecting all words which make a **‘P,** ‘PP or ‘I’ link with some word to their right. Also, if a verb makes a ‘TO link with “to” which in to makes an ‘I’ link with some word, then both are neglected. (Refer Figure (6-2))



```
+-Ss+-TO-+-Ix+---Pv--+
|   |    |  |        |
He was.v to be.v rewarded.v
```

Figure (6-2): Pruning verb phrase

3. In some cases, adjectives are also treated as verbs because they too form ‘P links with forms of “be” and, ‘MV’ and ‘TO’ links with modifying phrases just l i.e. verbs. This is necessary to predict the subjects of verbs occurring in modifying phrases. (Refer Figure (6-3))



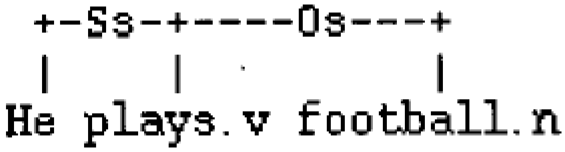
```
+-Ss+--Paf-+-TOf-+--I--+
|   |      |     |     |
It is.v likely.a to happen.v
```

Figure (6-3): Adjectives as verbs

6.2.1.3 Rules for Subject Prediction

After all the main verbs have been identified, the subject and object (if it exists) for each of them is predicted based on the following rules. First, let's go through the rules for subject prediction. The rules are applied in hierarchical fashion with the next rule being applied only if the subject is not found with all the rules before it. The only exception is that rule 4 is applied only if the subject is found in a rule before it. Also, each rule is applied not only to the main verb identified but also to each word occurring in the verb phrase.

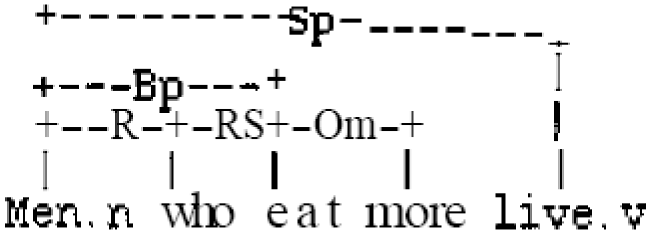
1. The most basic and obvious way of identifying the **sub**ject is by finding a word which makes either a **‘S’,** ‘SI’, **‘SX or ‘SXI’** link with the verb. (Refer Figure (6-4))**.**



```
+-Ss-+----Os---+
|    |         |
He plays.v football.n
```

Figure (6-4): He + plays

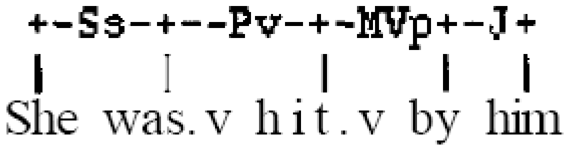
2. If a verb is connected to a noun by a ‘B’ link and the verb also bears a ‘RS’ link then the noun with which it has the “B’ link is its subject. (Refer Figure (6-5)).



```
+---------Sp-------+
+---Bp---+         |
+--R-+-RS+-Om-+    |
|    |   |    |    |
Men.n who eat more live.v
```

Figure (6-5): Men --t **eat**

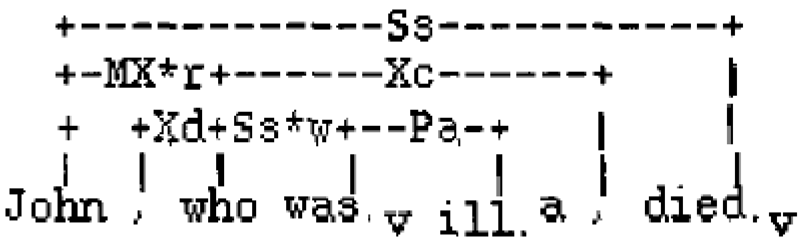
3. The above rules do not work in the case of passive sentences as the word with the ‘S’ link is actually the object. A sentence is deduced as passive if a **‘Pv’** link is present in the verb phrase. In such sentences, the subject is usually present in the form of the phrase “by subject”. Or else, the object is identified as done for normal cases and classified as the subject. (Refer Figure (6-6)).



```
+-Ss-+--Pv-+-MVp+-J+
|    |     |    |  |
She was.v hit.v by him
```

Figure (6-6): him + **hit** + **She**

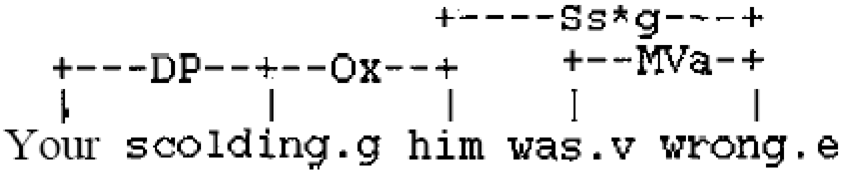
4. In some cases, the actual subject may be connected by a **‘MX*r’** link to the subject found by any one of the above three rules. (Refer Figure (6-7)).



```
+------------Ss-----------+
+-MX*r+------Xc------+    |
+  +Xd+Ss*w+--Pa-+   |    |
|  |  |    |     |   |    |
John , who was.v ill.a , died.v
```

Figure (6-2): John + was

5. When the verb occurs in the form of a gerund, the subject may he attached to the verb with the ‘DP link. (Refer Figure (6-8)).

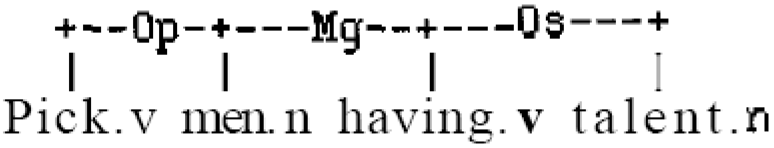


```
                     +----Ss*g---+
+---DP--+--Ox--+     +--MVa-+
|       |      |     |      |
Your scolding.g him was.v wrong.e
```

Figure (6-8): Your + scolding

The above five rules are the basic rules for finding the subject directly.

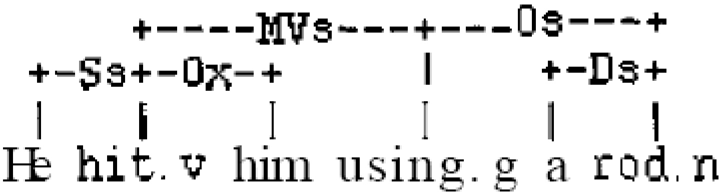
6. If a verb is connected to the object of some other verb with ‘Mg’ link then that object is the subject for this verb. (Refer Figure (6-9)).



```
+--Op-+---Mg--+---Os---+
|     |       |        |
Pick.v men.n having.v talent.n
```

Figure (6-9): men -i having

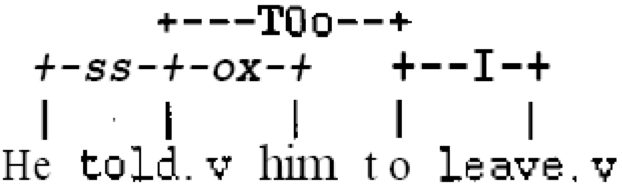
7. If a verb occurs in the phrase modifying a verb, wherein the phrase is connected to the verb with 'MV' link, then its subject is the subject of the verb it modifies. (Refer Figure (6-10))**.**



```
    +----MVs---+---Os---+
+-Ss+-Ox-+     |    +-Ds+
|   |    |     |    |   |
He hit.v him using.g a rod.n
```

Figure (6-10): He + **using**

8. If a verb occurs in the phrase modifying a verb, wherein the phrase is connected to the verb with 'TO link, then its subject is the object (if it exists) of the verb it modifies. If the verb which is modified does not have an object then its subject is the required subject. (Refer Figure (6-11)).



```
      +---TOo--+
+-ss-+-ox-+    +--I-+
|    |    |    |    |
He told.v him to leave.v
```

Figure (6-11): him --t **leave**

9. In the extreme case of all the above rules failing, the subject of the verb is taken as **any** noun to which the verb is connected with a 'M link. This rule need not be correct at all times.

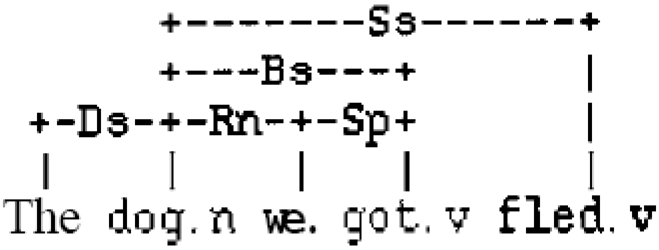
From the above rules it is clear that to find the subject, the object of the verb (if it exists) and the modifying phrases of both the verb and the object will also have to be found.

6.2.1.4 Rules for Object Prediction

The rules for finding the object are as follows:

1. Here too, the most basic way of finding the object is to find the word which makes either an 'O', 'OD or 'OT' **link** with the verb.

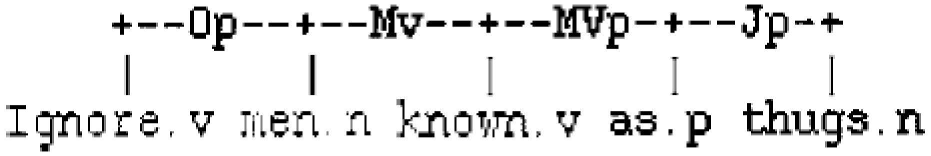
2. If the verb makes a 'B' link with a noun and the verb does not have a **'RS'** link then that noun is the object of the verb. (Refer Figure (6-12)).



```
          +--------Ss-------+
          +---Bs---+        |
  +-Ds-+-Rn-+-Sp+           |
  |    |    |   |           |
The dog.n we. got.v fled.v
```

Figure (6-12): we + got + dog

3. **If** a verb makes a 'Mv' link with the object of some other verb then that object is the object of this verb as well. (Refer Figure (6-13))**.**



```
    +--Op--+--Mv--+--MVp-+--Jp-+
    |      |      |      |     |
Ignore.v men.n known.v as.p thugs.n
```

Figure (6-13): known + men

4. Also, as already mentioned, in the case of passive sentences, the subject and object are interchanged.

6.2.1.5 Rules for Modifying Phrase Prediction

After finding the verb, subject and object, their modifiers have to be found as they are required to find the subject and object of verbs occurring later. Any phrase which forms a complete linkage structure on its own and is connected to a verb by a ‘MV’ or ‘TO’ link is classified **as** a verb modifying phrase. Similarly, for subjects and objects, in fact for any noun, a phrase is said to modify them if it forms a complete linkage structure on its own and is connected to the noun by means **of** a "M" link. For instance, in the sentence ***“He is said to have killed him.”*** it is not possible to deduce who is the subject for the verb ***said*** from the article alone. Such verbs are called ‘agentless passives’.

Example 1 (active main verb):- The sentences from the biomedical abstracts are parsed using the Link Grammar Parser (LGP). For example, if the input of the system is a clausal structure of the sentence: *DMP-1 regulates DSPP during odontoblast differentiation.* The LG parser gives the output in the form of links between words as shown in Figure (6-14). The output will be in the form: (PROTEIN1, Interaction word, PROTEIN2) as explained in the following algorithm for Interaction Extractor.

1. The main verb *regulate* is identified.
2. The algorithm uses the links given by the LG parser to predict and obtains subject, object and modifying phrase as shown: Subject (S): *DMP-1* Object (O): *DSPP* which are both protein names. Modifying Phrase (MV): *odontoblast differentiation.*
3. The main verb is an interaction word.
4. The system tries to extract interaction between subject, verb and object combination (S-V-O).
5. Since the main verb is tagged as an interaction word, the interaction extractor uses the S-V-O composite role from to find and extract the following complete interaction: [DMP-1, regulate, DSPP]

```
    +--------------------------------Xp------------------------------------------------------+
    |                              +------MVp-----+--------------Jp------------+              |
    +-----Wd-----+---Ss--------+---Os---+       |        +---------A---------+              |
    |               |              |          |       |        |                      |              |
LEFT-WALL DMP-1 regulates.v DSPP during odontoblast[?].a differentiation.n   .
```

Figure (6-14): The linkage given by the link grammar parser.

Example 2 (modifying of the interaction word):- For extracting interaction between subject and modifying phrase combination let us consider another example for the sentence *"OPN interacts with cell surface CD44 through their ino termini.*" The LG parser gives the output in the form of links between words as shown in Figure (6-15). The boundaries of the subject and the modifying phrase are identified as explained in the following algorithm for Interaction Extractor.

1. The main verb *interact* is identified.
2. The algorithm uses the links given by the LG parser to obtain subject, object and modifying phrase as shown below: Subject (S): “*OPN*” and Modifying Phrase (MV): “*with cell surface CD44*”.
3. The main verb is an interaction word. The main verb ‘‘interact" is identified and the system tries to extract interaction between subject and **Modifying Phrase combination (S-V-MP).**
4. Then the roles are found for each one of them, here subject is a protein name and modifying phrase is a protein name.
5. Since the main verb is tagged as an interaction word, information extractor uses the S-V-M composite role to find and extract the following complete interaction: [OPN, interact, cell surface CD44].

```
+---------------------------------------Xp-------------------------------------------------------------+
   |                        +----------------MVp---------------------+------------Jp---------+         |
   |                        |         +----------Js----- ---------+        |       +--------D*u----+       |
  +------Wd-----+---Ss--+--MVp-+     +---AN--+---GN--+        |       |      +-----A----+        |
   |                |        |         |     |         |           |       |       |    |            |        |
LEFT-WALL OPN interacts.v with cell.n surface.n CD44 through their ino[?].a termini[?].n .
```

Figure (6-15): The linkage (parse) given by the link grammar parser.

6.2.2The main verb is not an interaction word

The system has searched for all occurrences of the ***interaction words*** where they occur as main verbs. If the main verb is not an interaction word each occurrence of the interaction word or one of *its* synonyms and hyponyms is to be one occurrence of the required interaction. *So,* by finding the subject, object as well as all available modifiers, almost all information about that instance of the event can be extracted from the document. Now use the rules enumerated in the previous section to identify the subject and object (if present) of the verb as well as the modifiers of all three (verb, subject and object).

The PIELG system will apply different approach if the main verb is not an interaction word or if there more than one interaction in the sentence. We need to detect the interaction word whatever its place in the sentence. Then the system predicts its subject, object and modifying phrase for each interaction word. Then the program starts to check if they are a protein name or not. And so on to extract the relation between two protein names in the sentence whatever its place. In the following section we will display various sentences and the output of the sentence.

Example 1 (nested interactions):- For example for the sentence *"DSPP binds DMP-1 and activates DPP*". The LG parser gives the output in the form of links between words as shown in Figure (6-16). The extracted information will be [DSPP, bind, DMP-1] and [DSPP, activate, DPP] as explained in the following algorithm for Interaction Extractor.

1. There are two interaction words. One is the main verb which is *bind* the other is *"activate"*. All interaction words in the sentence whatever its place are identified.
2. The algorithm uses the links given by the LG parser to predict and obtain subject, object and modifying phrase for each interaction word in the sentence.

3. The Subject (S) is: *DSPP*. The Object (O) of the first interaction word *bind* is: *DMP-1*. The Object (O) of the second interaction word *activate* is: *DPP*. Both the Subject and the object of each interaction word is a protein name.
4. The system tries to extract interaction between verb, subject and object.
5. Since the main verb is tagged as an interaction word, the interaction extractor uses the S-V-O composite role from to find and extract the following complete interaction: [DSPP, bind, DMP-1].
6. The second verb is tagged as an interaction word, the interaction extractor uses the S-V-O composite role from to find and extract the following complete interaction: [DSPP, activate, DPP].

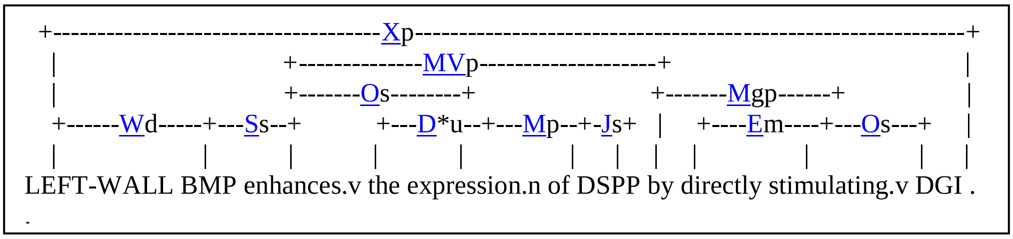
```
  +--Ss-+--Os--+
  |     |      |
DSPP binds.v DMP-1 and activates.v DPP


  +------------Ss-----------+---Os--+
  |                         |       |
DSPP binds.v DMP-1 and activates.v DPP
```

Figure (6-16): The parse given by the link grammar parser.

Example 2:- Another example "BMP enhances the expression of DSPP by directly stimulating DGI". The LG parser gives the output in the form of links between words as shown in Figure (6-17). There are two interaction words the output of the system in that case will be as follows [BMP, enhance, DMP-1] and [BMP, stimulate, DGI] as explained in the following algorithm for Interaction Extractor.

1. There are two interaction words. One is the main verb which is *enhance* the other is *stimulate*. All interaction words in the sentence whatever their places are identified.
2. The algorithm uses the links given by the LG parser to predict and obtain subject, object and modifying phrase for each interaction word in the sentence.
3. The Subject (S) of both interaction words is: *BMP*. The Object (O) of the first interaction word *enhance* is: *expression of DSPP*. The Object (O) of the second interaction word *stimulate* is: *DGI*. Both the Subject and the object of each interaction word is a protein name.
4. The system tries to extract interaction between verb, subject and object.
5. Since the main verb is tagged as an interaction word, the interaction extractor uses the S-V-O composite role from to find and extract the following complete interaction: [BMP, enhance, DMP-1].
6. The second verb is tagged as an interaction word, the interaction extractor uses the S-V-O composite role from to find and extract the following complete interaction: [BMP, stimulate, DGI].



```
 +----------------------------------Xp-----------------------------------------------------------+
  |                      +-------------MVp-------------------+                                   |
  |                      +-------Os--------+                 +-------Mgp------+                  |
  +------Wd-----+---Ss--+          +---D*u--+---Mp--+-Js+  |    +----Em----+---Os---+     |
  |             |        |         |         |         |    |    |  |           |         |     |
LEFT-WALL BMP enhances.v the expression.n of DSPP by directly stimulating.v DGI .
.
```

Figure (6-17): The linkage (parse) given by the link grammar parser.

6.2.3 Phrasal-prepositional Verbs Patterns

Phrasal-prepositional verbs are a small group of multi-word verbs made from a verb plus another word or words. Many people refer to all multi-word verbs as phrasal verbs. There are three types of multi-word verbs: prepositional verbs, phrasal verbs and phrasal-prepositional verbs. Here, we are interested in **phrasal-prepositional verbs**. Phrasal-prepositional verbs are made of: Verb + adverb + preposition. Many verbs in English are followed by an adverb or a preposition (also called a particle), and these two-part verbs, also called phrasal verbs, are different from verbs with helpers. A phrasal verb can contain an adverb and a preposition at the same time. Again, the verb itself can have a direct object:

- *No direct object:* The driver *got off to* a flying start.
- *Direct object:* Onlookers *put* the accident *down to* the driver's loss of concentration.

Phrasal-prepositional verbs could be viewed as: Verb + Particle + Preposition Combinations (Phrasal Verbs + Prepositions)**.** In this part we treated the case of preposition combinations. There are a small number of preposition combinations, such as *by-of*, *from-to* etc., which occur frequently within the clauses. Those prepositional combinations are used to distinguish the agent, the predicate and the theme of the interactions. To solve this problem, the system uses **phrasal-prepositional verbs** patterns to find agent, predicate, theme and action to extract the interaction, for both active and passive voices. This is an example of phrasal-prepositional verbs "*Gene expression* ***of*** *TGF-beta1 was sharply down-regulated by LTA in odontoblasts*."

Phrasal-prepositional verbs pattern Matching is the phase in an information extraction process that does the main job of finding interesting information bits from the output of the Link Grammar Parser and extracting that information for further processing. Depending on the complexity of the requested information model and the way in which the information is presented in the source data, the information extracted by this phase is combined and transformed to produce the final output of the information extraction process. For preposition based deep extraction the system uses a pseudo code. The algorithm is repeated for each sentence of the text. This code starts by finding pattern corresponding to the prepositional combinations in the string. If the prepositional combinations exists the pattern (predefined patterns), then extract protein - protein interactions using the pattern.

Example 1:- This is an example of prepositional combinations in phrasal-prepositional verbs "*Gene expression* ***of*** *TGF-beta1 was sharply down-regulated by LTA in odontoblasts*." In this example, there is a preposition combination between *by* and *in*. There are two modifier phrases. The first one is *LTA* which is the subject of the passive voice. The second one is *odontoblasts* which is the modifier of the main verb. The system used the **Phrasal-prepositional verbs** pattern to distinguish the modifier of the verb from the subject of the passive voice. In this sentence, the main verb (action) is an interaction word which is *down-regulated*. The agent is *LTA* which is a protein name. The predicate is *gene expression of TGF-beta1* which is also a protein name. The theme is *odontoblasts*. Odontoblasts are cells lining the inner surface of the tooth. The predefined patterns for this sentence is the *by-in* pattern [(PROTEIN1 (predicate)) (is/are) or (was/were) (Interaction-Word (action)) by ... (PROTEIN2 (agent)) ... in... (Theme) ...]. The interaction extractor is able to extract the correct interaction (LTA, down-regulate, TGF-beta1, in, odontoblasts). The final step in the interaction extraction module is re-transformation. The main job of the re-transformer is to insert multiple words of protein names back after manipulation.

Example 2:- The following sentence is in the passive voice *"DSP is cleaved* ***into*** *DPP* **in** *odontoblasts."* Here the main verb (action) is an interaction word which is *cleaved*. This sentence is an example of the *into-in* combination in the passive voice. Here there is no agent (subject of the active voice). The predicate is *DSP* which is a protein name. The patient is *DPP* which is also a protein name. The theme is *odontoblasts*. The predefined patterns for this sentence is the *into-in* pattern [(PROTEIN1 (predicate)) (is/are) or (was/were) (Interaction-Word (action))...into ... (PROTEIN2 (patient))... in... (Theme)...].The interaction extractor is able to extract the correct interaction (DSP, cleaved, into, DPP, in, odontoblasts).

6.2.4 Nominal form

Biochemical interactions described in PubMed abstracts are rarely stated as simply as “protein A activates protein B.” Various syntactic structures are used to compact several interactions, as well as other information, into a single sentence. Among the most frequently used are nominalization (converting a predicate to a noun phrase) and coordination (combining two or more predicates with coordinating conjunctions). Examples for nominalization are "interaction of, interaction between, the association of, "phosphorylation of, dephosphorylation of, activation of, and so on". While many previous information extraction projects have concentrated only on the verbal forms of interactions, patterns for the nominal form in the case of ‘phosphorylate’ interactions is needed. Here we present a series of examples to illustrate how the rules operate and identify the desired information. The following examples illustrate nominalization.

1. **The theme can appear before ‘up-regulation’ as in ‘DMP-1 up-regulation’:-**
    - When an argument (protein) appears before ‘up-regulation,’ this protein is likely to be the agent. Its role is normally indicated clearly; such as with the theme appearing after ‘up-regulation’ as in the following pattern: **[_AGENT_ up-regulation] NP by _THEME_].**
    - Let us take the following sentence as an example :
        - "DMP-1 up-regulation by Cbfa in human dental pulp stem cells was activated ."
            - THE RESULT IS : [--Cbfa up-regulated DMP-1----in— human dental pulp stem cells]
        - "The phosphophoryn activation of Smad1 implies this is a direct effect".
            - THE RESULT IS : phosphophoryn.n--activation.n--of--Smad1----

2. **The agent and theme can also appear after ‘phosphorylation’ as captured by the following pattern:**
    - [Phosphorylation of _THEME_ (by _AGENT_)? (in /at _SITE_)]. The arguments are attached through the *"of"* and "*by"* prepositional phrases, where the latter identifies the agent role.
    - Let us take the following sentences as examples :

- "Phosphorylation of Smad1 by phosphophoryn was enhanced ".
    - THE RESULT IS : [phosphophoryn--phosphorylated Smad– 1]
- "The Up-regulation o f DMP-1 promoter by Cbfa in HDPSC was activated".
    - THE RESULT IS : [--Cbfa -- Up-regulated -- DMP-1-- in --HDPSC -]

3. The system is also able to identify *dephosphorylation* relations, as exemplified by the following nominalization example, from which we extract that DSPP—are dephosphorylaed by DMP-1. Let us take the following sentence as an example :
    - Dephosphorylation of DSPP by DMP-1 was carried out.
        - THE RESULT IS: [--DMP-1- dephosphorylated--- DSPP].
4. Another example as captured by the following pattern: **[Nominalized interaction word between _THEME_ and _THEME_].** The arguments are attached through "*between*" prepositional phrase where they identify the theme role. Let us take the following sentences as examples :
    - Interactions between DMP1 and DSPPP provide that DMP1 regulates the expression of the DSPP gene.
        - THE RESULT IS :DMP1--interactions.n----DSPPP—
        - THE second RESULT IS :DMP1--regulates---- expression of the DSPP gene--.
5. Another example as captured by the following pattern: **[Nominalized interaction word of _THEME_ and _THEME].** The arguments are attached through the *"of"* prepositional phrase, where the latter identifies the theme role. Let us take the following sentences as examples :

- Up-regulation of ITGA1 and CD44 has been reported separately.
    - THE RESULT IS : upregulation.n--of--ITGA1--and---CD44--

CHAPTER 7

RESULTS AND EVALUATION

The first part of this chapter presents the results produced by the PIELG system. The remaining part presents the evaluation of the results of the PIELG system and an analysis based on the evaluation. The evaluation is divided into two phases. The first phase of the evaluation process for PIELG system was the evaluation of the information extraction performance by measuring the metrics Precision and Recall. And so, perform experimental evaluations with two other state-of-the-art extraction systems – the BioRAT and IntEx comparison indicate that PIELG system achieves better performance. The second phase of the evaluation process for PIELG system was the evaluation of the PIELG system as compared to the visualizing software requirements.

7.1 Results

Surface Variations of the Same Information, IE can be seen as a process that reduces diverse surface forms in text into a fixed standard representation when they express the same information. Whether two forms in text express the same information or not depends on the perspectives or interests researchers have.
For example, "*Entity1* interacts with non-polymorphic regions of *Entity2*" can be considered to express the same information as "*Entity1* interacts with *Entity2*" if one is interested in general protein-protein interaction regardless of their modes of interaction, but cannot be for others whose interests are the modes. In short, the application-specific nature of IE resides in this kind of perspective-dependency in the definition of information.

However, there are other types of surface variations that express the same information regardless of users' perspectives, such as "*Entity1* activates *Entity2*" and "*Entity2* is activated by *Entity1*". In some cases, a surface form can be considered to contain as its part the same information that another form expresses, regardless of users' perspectives. "*Entity1* can activate *Entity2*" vs. "*Entity1* activates *Entity2*" are such examples. We mean that by a Link Grammar parser, a program which assigns standard forms to surface sentences, the same information of these kinds is represented in the same formats. In this format, all the surface forms in Table (7-1) share the same information "*Entity1* activates *Entity2*" as their part.

What is important here is that computation from surface sentences to linkage representation can be carried out regardless of users' perspectives and that linkages represented by single forms the same information that appear in very different sequences of words. Due to such reduction in complexity, we can expect that the construction algorithm of IE rules that works on linkage representations needs a much smaller training corpus than those working on surface word sequences. Furthermore, due to the reduction of surface diversity at the linkage representation level, an IE system with extraction rules at this level should show improved performance in terms of recall.

7.1.1 Classification of treated forms

Although many previous biomedical IE system focus on verbs which represent target events by themselves (i.e. "activate", "bind"), there are many cases that combinations of verbs, prepositions and certain nouns form proper IE patterns. We investigated and classified forms which are needed to extract interacting protein pairs to see what forms are required in addition to ones that consist of only one verb. We found nine classes based on constituents of the verbs. Table (7-1) shows Syntactical variation of the interaction words in various contexts.

The PIELG system covered nine classes based on the syntactical variation of the interaction words in various contexts as shown below:

- Class (1) consists of the simplest sentence form, which includes only one verb and interacting proteins (entities). This kind of sentences was the main part of early works.

- Class (2) includes the passive case where the order of the subject and the object is reversed.

- Class (3) consists of the simplest sentence form, which includes only one verb and interacting proteins (entities) and a modifying phrase of the verb.

- Class (4), includes verbs after auxiliary verbs.

- Forms in class (5) include verbs in the past participle forms.
- Class (6) is the infinitive case.
- Ones in class (7) are based on nouns representing interaction themselves (ex. “interaction”, “Phosphorylation”).

- Class (8) includes the phrasal-prepositional verb Patterns. This case where there is a combination between the prepositions.

- Class (9), includes the case where there are more than one interaction word in the sentence. In most cases, one of the verbs in the patterns is used to modify a noun phrase.

Table (7-1):- Linguistic variation of the interaction words in various contexts.

Class (1):- Active main verb

- Entity 1 recognizes and **activates** Entity 2.
- Our results indicate that Entity 1 **inhibits** the activated Entity 2.
- Entity 1 **activates** Entity 2 **and binds** Entity 3**.**
- Entity 1 up-regulates the expression of Entity 2.
- *Entity 1* prevents the decrease of *Entity 2* and inhibits *Entity 3*.

Class (2):- Passive

- Entity 2 is activated by Entity 1.
- The expression of Entity 1 is induced by Entity 2 in primary cultured dental pulp cells not in calvaria osteoblasts.
- A gene encoding Entity 1 is processed into two proteins Entity 2 and Entity 3.

Class (3):- Modifying phrases of verbs

- Both *Entity1* and *Entity 2* interact with cell surface *Entity 3* through their amino termini.
- *Entity 1* associates with the *Entity 2*.
- *Entity 1* consists of *Entity 2* and *Entity 3*.
- *Entity 1* binds strongly to *Entity 2*.
- *Entity 1* is able to bind specifically with the *Entity 2*.

Class (4):- After an Auxiliary Verb

-
- *Entity 1* may bind large amount of *Entity 2*.
- *Entity 1* must be proteolytically processed to form these two *Entity 2* proteins.
- The sites of *Entity 1* may also contain *Entity 2*.

Class (5):- Past particle

- *Entity 1* activated *Entity 2*.

Class (6):- Infinitive

- *Entity 1* is able to inhibit *Entity 2*.

Class (7):- Nominalization

- The Up-regulation of *Entity 1* by *Entity 2* in *Entity 3* was activated.
- *Entity 1* up-regulation by *Entity 2* in *Entity 3* was activated.
- Dephosphorylation of *Entity 1* by *Entity 2* was carried out.
- The phosphophoryn activation of *Entity 1* implies this is a direct effect .
- Phosphorylation of *Entity 1* by *Entity 2* was enhanced .

Class (8):- **Phrasal-prepositional Verbs Patterns**

- *Entity 1* was expressed in *Entity 2*.
- *Entity 1* was expressed by *Entity 2* throughout *Entity 3* in the *Entity 4* .
- *Entity 1* was performed for *Entity 2*.
- *Entity 1* is primarily synthesized as *Entity 2* .
- *Entity 1* is cleaved into *Entity 2* and *Entity 3* in *Entity 4.*
- *Entity 1* is associated with mutations in *Entity 2*.
- *Entity 1* is probably regulated by *Entity 2* during dentinogenesis .

Class (9):- Nested interactions

- Entity 1 signals Entity 2 by directly stimulating Entity 3.
- *Entity 1* prevents the decrease of *Entity 2* and inhibits *Entity 3.*

7.2 Evaluation

The evaluation process for the PIELG system is divided into two phases. The first phase of the evaluation process for PIELG system is the evaluation of the information extraction performance by measuring the metrics Precision and Recall. Then, experimental evaluations with two other state-of-the-art extraction systems – the BioRAT and IntEx indicate that PIELG system achieves better performance. The second phase of the evaluation process for PIELG system is the evaluation of the PIELG system as compared to the visualizing software requirements set in chapter 9.

7.2.1The First Phase of the Evaluation Process

Information extraction systems are evaluated on the basis precision and recall measures. Precision and Recall for PIELG system are calculated using the equations 7.1 and 7.2. Then, we perform experimental evaluations with two other state-of-the-art extraction systems – the BioRAT and IntEx. Information Extraction researchers can use their systems to extract protein-protein interactions, and then compare these with the records in protein- protein interactions databases like: DIP, BioGRID[17] and so on. Each record of the database has a pair of proteins that interact with each other. Each protein defined with entry keys to different protein databases. And so, in our evaluation phase we choose BioGRID database.

7.2.1.1 Algorithm

First, we evaluate our system by selecting pairs of proteins which are known to be interacting with each other from BioGRID, the protein-protein interaction

[17] http://www.thebiogrid.org/

databases. We choose six queries currently considered to have applications in dental medicine: *osteopontin (SPP1)*, *CD44 molecule (CD44)* (Indian blood group), *Dentin matrix acidic phosphoprotein*; *dentin matrix protein-1 (DMP-1), Collagen, type I*; *alpha 1 (COL1A1), decorin (DCN), biglycan* (*BGN).* Then we look up their interaction properties using BioGRID database. Then, we send those six queries to PubMed separately retrieving 30 abstracts. After manually reviewing all these abstracts, we found that 89 (82%) among them are correct.

The second queries are arbitrary pairs of proteins. Then, we evaluate our system by determining pairs of unknown proteins. We didn't know their interaction properties. The proposed system starts to extract all the information about interaction properties of both proteins from the linkage representations of the retrieved abstract. Then we evaluated the obtained interactions by referring to BioGRID. Then we start to compare their interaction properties to measure the metrics Precision and Recall. Then, we perform experimental evaluations with two other state-of-the-art extraction systems – the BioRAT and IntEx.

7.2.1.2 Recall Analysis

Recall is a measure of sensitivity of the system, giving an account of how often the system is able to extract the right results. It is calculated as the ratio of true positives to the sum of true positives and false negatives. The true positive is the interactions extracted correctly. The false negative is the interaction extracted incorrectly. The summation of the true positive and false negative is the total interactions present in the text.

$$\text{Recall} = \frac{|\text{Interactions extracted correctly}|}{|\text{Interactions present in text}|} \qquad (7.1)$$

For recall comparison with BioGRID database, we compared our extracted results with BioGRID entries manually. If an interaction (both the protein names) matches with a BioGRID entry for a given abstract, we take this as ‘Match’. If no BioGRID entry matches an extracted interaction for a given abstract, then we take it as ‘No Match’. These numbers are given in Table (7-2). We have 250 interactions as matches and it gives a recall of 47.4%**.** Low figure for recall is due to the fact that BioGRID database has interactions from both abstract and full text of the paper, and for our evaluation we extracted interactions only from abstracts.

Table (7-2): Recall of PIELG when compared with BioGRID database

PIELG	**Cases**	**Percent %**
Match	250	47.4%
No Match	277	52.56%
Totals	527	100.00

Like BioRAT and IntEx, We manually re-analyzed these records with no reference to BioGRID but instead we counted how many of **PIELG**’s predictions were correctly extracted from the text. Table (7-3) shows the recall from these abstracts by **PIELG**, namely 47.4%, which is much higher than BioRAT (20.31%) and IntEx (26.94%).

Table (7-3): Recall comparison of IntEx and BioRAT from 229 abstracts when compared with BioGRID database

Recall Results	**PIELG**		**IntEx**		**BioRAT**	
	Cases	**Percent %**	**Cases**	**Percent %**	**Cases**	**Percent %**
Match	250	47.4%	142	26.94	79	20.31
No Match	277	52.56%	385	73.06	310	79.69
Totals	527	100.00	527	100.00	389	100.00

7.2.1.3 Precision Analysis

The Precision value of a system is a measure of the specificity of the system. It gives an idea of the correctness of the system by measuring the number of times the results are extracted correctly in comparison with the total number of results.

$$\text{Precision} = \frac{|\text{Interactions extracted correctly}|}{|\text{Interactions extracted}|} \qquad (7.2)$$

We evaluated precision of extracted interactions manually, and we also compared these interactions with BioGRID database. A total of 399 interactions were extracted from 229 abstracts and each one of the interactions was manually checked for correctness. Out of 399 interactions, we found 250 of these as "Match" with BioGRID entries for the same abstract. Table (7-4) shows these results. Table (7-5) shows the precision from these abstracts by PIELG, 62. 65%, which is a bit higher than BioRAT (55.07%) and but lower than IntEx (65.66%).

Table (7-4): Precision results for PIELG system, when compared with BioGRID database.

	PIELG compared with BioGRID	
	Cases	**Percent %**
Match	250	62. 65%
Totals	399	100.00

Table (7-5): Precision comparison of IntEx and BioRAT from 229 abstracts

Precision	**PIELG**		**IntEx**	**BioRAT**		
Results	**Cases**	**Percent %**	**Cases**	**Percent %**	**Cases**	**Percent %**
Correct	250	62. 65%	142	35.58	239	55.07
Incorrect	149	47.45%	257	34.34	195	44.93
Totals	399	100.00	399	100.00	434	100.00

7.2.1.4 The corpus

The first phase of the evaluation process for PIELG system was performed on the selected corpus. The scope of our experiments is limited to abstracts describing human protein function. The corpus of the PIELG is selected in order to evaluate the proposed protein-protein interaction validation method. This corpus is selected to be about proteins currently considered to have roles in *dentine formation* process and involved in dentinogenesis.

Amounts of the non-collagenous proteins in dentin; *decorin (DCN), biglycan (BGN) , osteonectin (SPARC), osteocalcin, osteopontin (SPP1), bone sialoprotein,* and *Dentin matrix protein-1 (DMP-1),* which are detected in the bone matrix, are also found in the dentin. However, two extracellular matrix proteins have been shown to be specific for the dentin matrix: the *Dentin Sialoprotein (DSP*) and the *Dentin Sialophosphoprotein (DSPP).* Furthermore, dentin is a reservoir of growth factors such as *Transforming Growth Factor Beta (TGF3*), *Bone Morphogenetic proteins (BMPs),* and *Fibroblast Growth Factors (FGFs*), since these molecules are captured in the dentin matrix

7.2.1.5 Results for the first evaluation step

In The first step we evaluated our system by selecting pairs of proteins which are known to be interacting with each other from BioGRID. The result of the first step is that our system has extracted successfully that *osteopontin (SPP1),* interacts with *CD44 molecule (CD44). Dentin matrix acidic phosphoprotein*; *dentin matrix protein-1 (DMP-1)* interacts with *CD44 molecule (CD44). Collagen, type I*; *alpha 1 (COL1A1)* interacts with *decorin (DCN), biglycan* (*BGN)* and *CD44 molecule (CD44*). Also we notice that *Dentin sialophosphoprotein (DSPP*) consists of two proteins *Dentin phosphoprotein (DPP*) and *Dentin sialoprotein (DSP).*

For the second queries which are arbitrary pairs of proteins the system downloaded abstracts related to that pairs of proteins. The selected corpus consists of 229 abstracts out of 1000 sentences, including abstract titles, with annotated proteins and interactions. Those 1000 sentences are sentences which contain one pair of proteins and one interaction word. If a sentence includes more than one interaction, all interactions are counted as answers. Additionally, the presented system tried to extract all. The parser processed 880 sentences, and did not process 120 sentences. The percentage of parsed sentences is 88%. The percentage of failed sentences is 12%. After lexicon expansion, the parser could parse additional 89 sentences, and only 31 of 120 sentences are left out. The parser success rate is higher after personal name conversion and transformation phases. And so, the percentage of failed sentences becomes 1.2%.

The extracted interactions correspond to 229 abstracts from the PubMed. Using abstracts ID's (PubMed ID's) of these 229 abstracts; we downloaded 527 records form BioGRID[18] database those interactions represented in the 229 abstracts. BioGRID database entries were downloaded as a flat file from. PIELG system extracted 399 interactions from these 229 abstracts. Among of those 399 interactions 250 interactions are extracted correctly (matched BioGRID). For fair comparison, we have also limited our protein name dictionary used for tagging genes to the iHOP[19] entries.

The extracted interactions were compared with BioGRID entries manually. If an interaction extracted by PIELG is not found in BioGRID, it can be that (a) it is a false-positive example, reducing the precision of PIELG; or (b) the interaction is missing from BioGRID. The latter case consists of interactions that are mentioned in papers, but have not been added to BioGRID. We manually re-analyzed these records with no reference to BioGRID but instead we counted how many of PIELG**'s** predictions were correctly extracted from the text. The results of the first phase of the evaluation process gives a rate of recall and precision of extraction by PIELG are 47.4% and 62. 65%. BioGRID contains protein interactions from both abstracts and full text. Since our extraction system was tested only on the abstracts, the system missed out on some interactions that were only present in the full text of the abstract.

[18] http://www.thebiogrid.org/
[19] http://www.ihop-net.org

7.2.1.6 Error Analysis

A detailed analysis of the sources of all types of errors associated with all the protein-protein interaction extraction processing stages by PIELG system is shown in Figure (7-1),.Different sources of errors identified are: Link Grammar Parse, Protein Name Tagging, Interaction Word (Iword) tagging, Interaction Extractor and Preprocessing sentences for Link Grammar Parser. For each error occurred, we identified its source and increased its quota toward error count. As seen in Figure (7-1), protein name tagging is prime source of most of the errors (almost 45%). To improve protein name tagging better a named entity recognition module is needed. Most of the other sources of error are between 5% - 10%, but they need some improvements too.

Among others, the number of errors generated in interaction extraction stage is the biggest. The reason is that due to the complicity of the protein interaction expression it is rather difficult to compile the appropriate extraction rules and, therefore, many interactions are missed out.

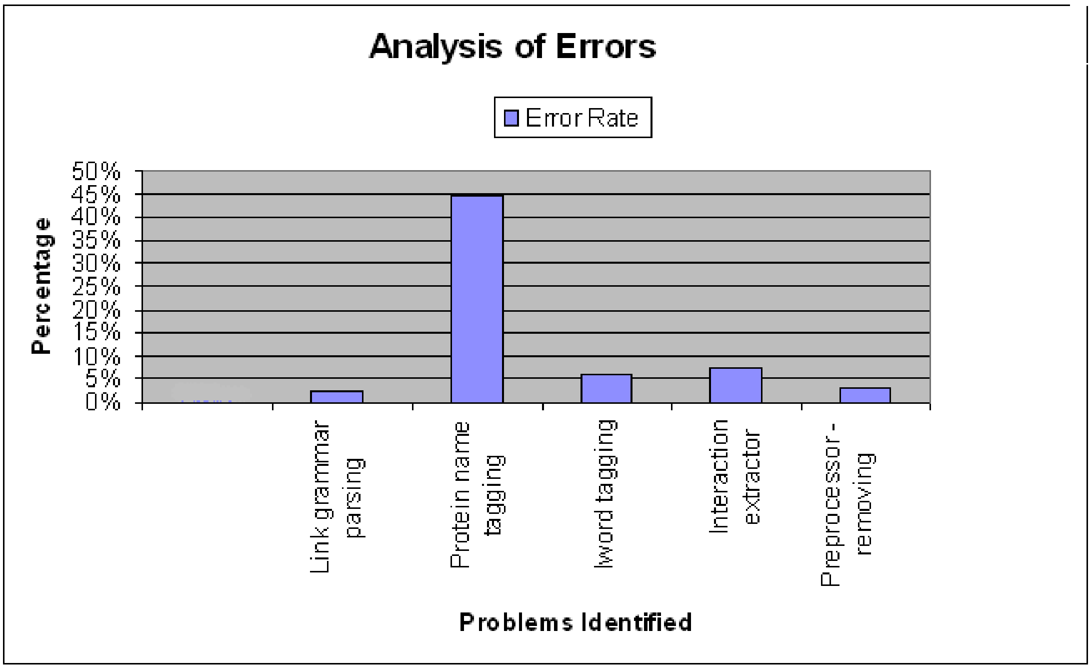


Figure (7-1): Analysis of different types of errors encountered.

7.2.1.7 Link Grammar Parser Errors

The errors generated in link grammar parsing because *Link Grammar Parser* itself may make some mistakes. For example, when dealing with too long sentences Link Grammar Parser will get into *panic* model in which the parser can parse even very long sentences quickly, but with considerably reduced accuracy. For example the sentence: *DSPP is an extracellular matrix protein that is cleaved into DSP and DPP with a highly restricted expression pattern in tooth and bone.* It is a too long ambiguous sentence.

Other reasons of parsing errors or failures may be for example, failure to recognize and correctly assign categories to all words in a sentence. These errors are caused by the presence of domain-specific concept notations including residue substitutions, chromosome positions, concentrations, cell line names, measurements of various parameters, etc.

For example the sentence: *COL1A1, BSP, DMP1, the marker for odontoblast DSPP and DSP were detected in these cells by immunohistochemistry RT-PCR and in situ hybridization.* This sentence conteins many domain-specific concept notations as *immunohistochemistry,* and *situ hybridization* which are not familiar to the lexicon of the Link Grammar Parser. A significant portion of these terms can be described using regular expression formalism which is implemented in the transformation phase. Lexicon errors constitute the major portion of parsing failures. Lastly, parsing failures may occur due to the incomplete grammar.

Ambiguity of the syntactic processing is a critical issue in practical applications of NLP systems. Due to the general ambiguity of syntactic knowledge, each sentence usually yields a number (sometimes very large) of potential sentence structures, but only one of them is generally considered correct. The source of ambiguity is investigated on parsed sentences by observing the structure of each alternative parse tree and correlating it with the compositional structure of the corresponding sentence. This analysis revealed that the major sources of ambiguity are variations in prepositional phrase attachment, and structures of coordinate conjunctions. For example the sentence: *DSPP and DMP-1 was induced by TGF-beta3 in primary cultured dental pulp cells.* Another example the sentence *DSP has enhanced DSPP*.

The parser gives two linkage representations. The first representation considers *enhanced DSPP* is the object of the verb *has* while *enhanced* here is an adjective. The second linkage representation considers the sentence in past participle tense so DSPP is the object of the verb enhanced.

7.2.1.8 Discussion about the First Phase

The heart of the system lies in the working of the rules for prediction of subject, object and their modifiers. The rules for the PIELG system are derived by running the link parser on abstracts of scientific papers including abstract and titles. Our experimental results show that the PIELG system presented here achieves better performance without the need of manual pattern creation (by user) which is required for these other systems .

The highly technical terminology and the complex grammatical constructs that are present in the biomedical abstracts make the extraction task difficult, Even a simple sentence with a single verb can contain multiple and/or nested interactions. That’s why PIELG is based on a deep parse tree structure presented by the Link Grammar and it considers a thorough case based analysis of contents of various syntactic roles of the sentences as well as their linguistically significant and meaningful combinations.

Most missed interactions are caused by semantic problems. Currently it is not necessarily the case that more powerful grammars lead to better biochemical interaction extraction. Until recently, most Information Extraction systems for mining semantic relationships from texts of technical sublanguages avoided full parsing [82]**.** Semantic Parsers for English language will be more useful and meaningful for the extraction tasks compared to Syntactic parsers. But constructing semantic parser is a difficult task and this parser will be more domains dependent. It is important to note, that using the Link Grammar in the proposed information extraction system makes it applicable to a large number of areas ranging from pathway analysis to clinical information and protein structure-function relationships. The time took for full parsing is also a problem for Information Extraction systems.

Although we have demonstrated that the Link Grammar Parser has the potential to be a useful part of a system for extracting biochemical interactions, its current limitations are also evident, as highlighted by the moderate performance gain in our experiment. A list of further developments that would enhance the importance of link grammar parsing in the biomedical domain is listed below.

1. Extending its dictionary to include technical terms.
2. Extending its unknown-word-guessing rules, so that, for example, the parser can guess that a word ending with -*ase* is a protein name and not a verb.
3. Developing other algorithms, such as template matching, to further process link paths extracted from the parser's output**.**
4. Modifying the grammar of the parser.

The PIELG system success to extract detailed contextual attributes of interactions by interpreting modifiers like: location/position modifiers (*in, at, on, into, up, over…),* agent/accompaniment modifiers (by, with…), purpose modifiers (*for*…), and theme/association modifiers (*of*...). Finally, several issues can make extracting interactions and relationships as a difficult job due to:

1. The task involves free text – hence there are many ways of stating the same fact.
2. The genre of text is not grammatically simple.
3. The text includes a lot of technical terminology unfamiliar to existing natural language processing systems.
4. Information may need to be combined across several sentences.
5. There are many sentences from which nothing should be extracted.
6. The abstracts of some papers are also used to take into consideration *technical style of writing*.

7.2.2 The second Phase of the Evaluation Process

PIELG can be augmented with various means of graphical packages. For further evaluation of the PIELG system, it is augmented with a graphical package for extracting protein interaction information from sequence databases. We used Cytoscape[20] which is a good tool for drawing directed graphs that can be adapted for drawing interaction extracted. Cytoscape is a graphical layout package developed for directed graphs. The augmentation process is done for two reasons. The first reason is to visualize the extracted interactions. The second reason is to evaluate the extracted interaction by drawing the pathways

[20] http://www.cytoscape.org/

for the extracted interaction. Then we compare those pathways with the stored pathways in Cytoscape.

Second evaluation for the PIELG system was done to extract relationships between interactions extracted from a collection of sentences (such as one interaction stimulating or inhibiting another) to construct (Protein Interaction Pathways) from abstracts. This could be done by directed graphs that are used for visualization of the extracted pathways.

7.2.2.1 Cytoscape

Cytoscape is an open source bioinformatics software platform for ***visualizing*** molecular interaction networks and ***integrating*** these interactions with gene expression profiles and other state data. Cytoscape was initially made public in July, 2002 (v0.8); the second release (v0.9) was in November, 2002. and v1.0 was released in March 2003. Version 1.1.1 is the last stable release for the 1.0 series. Latest version of Cytoscape is 2.6.0. Cytoscape Core developer team continue to work on this project and near future, they are going to release next major version, Cytoscape 3.0. It will be more modularized and scalable version of Cytoscape.

Although Cytoscape was originally designed for biological research, now it is a general platform for complex network analysis and visualization. Cytoscape *core* distribution provides a basic set of features for data integration and visualization. Additional features are available as ***plugins***. Plugins are available for network and molecular profiling analyses, new layouts, additional file format support, scripting, and connection with databases. Plugins may be developed by anyone using the Cytoscape open API based on Java™ technology and plugin community development is encouraged. Most of the plugins are freely available.

7.2.2.2 Algorithm

The visualization process (Drawing Pathway Diagram) for a **specific protein** composed of three stages for **Creating Networks** using Cytoscape:

1. Creating an empty network and manually adding nodes and edges. *The first stage* is to create an empty network and manually add nodes and edges. We gathered the extracted interaction prosperities for a specific protein from the extracted interactions by the PIELG system. We used Cytoscape to draw Networks for the extracted interactions. Proteins are represented by nodes, and interactions (or other biological relationships) are represented by edges between nodes. For compactness, a gene also represents its corresponding protein. Nodes may also be used to represent compounds and reactions (or anything else) instead of genes.

2. Importing pre-existing formatted network files. **The second stage** is retrieving the interaction prosperities of the previously mentioned protein from BioGRID database. The interactions of a specific protein are downloaded as a flat file from BioGRID database. Then we use Cytoscape to Creating Networks by importing pre-existing, formatted network files. Network files can be specified in any of the formats. The network file can either be located directly on the local computer, or found on a remote computer. Here we Load Networks from Local Computer. We retrieve the interaction prosperities of a specific protein from BioGRID database. The interactions of a specific protein are downloaded as a flat file from BioGRID database. Then we use Cytoscape to Creating Networks by importing pre-existing, formatted network files.

3. Importing networks from Web Service. **The third stage** is to use Cytoscape to Creating Networks by importing networks from Web Service. From version 2.6.0, Cytoscape has a new feature called **Web Service Client Manager**. Users can access various kinds of databases through this function. A web service is a standardized, platform-independent mechanism for machines to interact over the network. These days, many major biological databases publish their system with web service API. This enables developers to write a program to access these services. Cytoscape core developer team has developed several sample web service clients using this framework. Currently, Cytoscape supports the following web services:

    - IntAct: an open source database of protein interaction data, hosted at EMBL-EBI.
    - Pathway Commons: an open source portal, providing access to multiple integrated data sets, including: Reactome, IntAct, HPRD, HumanCyc, MINT, the MSKCC Cancer Cell Map, and the NCI/Nature Pathway Interaction database.
    - NCBI Entrez Gene: a public database of genes, including annotation, sequence and interactions.

We retrieve Protein-Protein Interaction Networks from NCBI Entrez Gene.

7.2.2.3 Examples implementing the three steps

The visualization process (Drawing Pathway Diagram) for Collagen, type I (COL1A1), transforming growth factor, beta 1 **(TGFB1**) and DMP-1 will be represented as follows. .

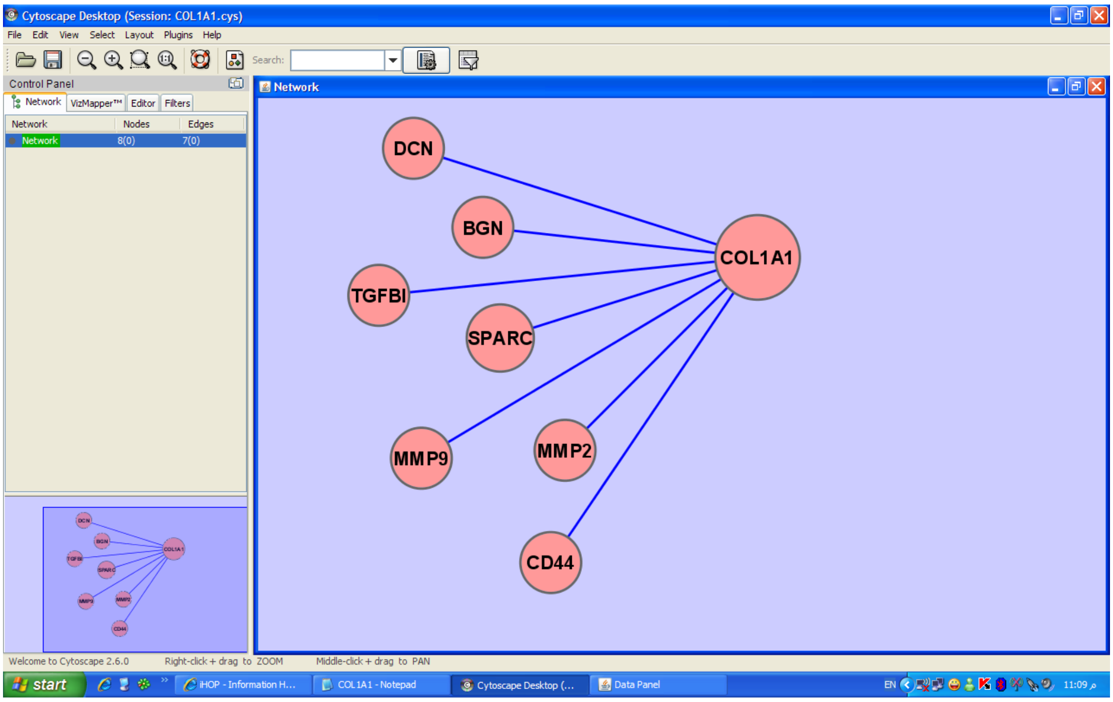
7.2.2.3.1 Collagen, type I (COL1A1)

The first stage: - Gathering the extracted result of the PIELG system for Collagen, type I (COL1A1) we found that COL1A1 interacts with several proteins as shown in Table (7-6).

Table (7-6): Protein interactions identified by PIELG for COL1A1

Name	Description
DCN	decorin
TGFBI	transforming growth factor, beta-induced, 68kDa;
SPARC	osteonectin
BGN	biglycan
MMP9	matrix metallopeptidase 9
MMP2	matrix metallopeptidase 2
CD44	CD44 molecule (Indian blood group)

We used Cytoscape to create an empty network and manually add nodes and edges. **We draw Networks** for the extracted interactions of Collagen, type I (COL1A1). Proteins are represented by nodes, and interactions (or other biological relationships) are represented by edges between nodes. The final network is represented in Figure (7-6)


Cytoscape Desktop (Session: COL1A1.cys)
File Edit View Select Layout Plugins Help
Search:
Control Panel
Network VizMapper™ Editor Filters
Network
DCN
BGN
TGFBI
SPARC
COL1A1
MMP9
MMP2
CD44
Welcome to Cytoscape 2.6.0
Right-click + drag to ZOOM
Middle-click + drag to PAN
start
iHOP - Information H...
COL1A1 - Notepad
Cytoscape Desktop (...
Data Panel
EN
11:09

Network	Nodes	Edges
Network	8(0)	7(0)

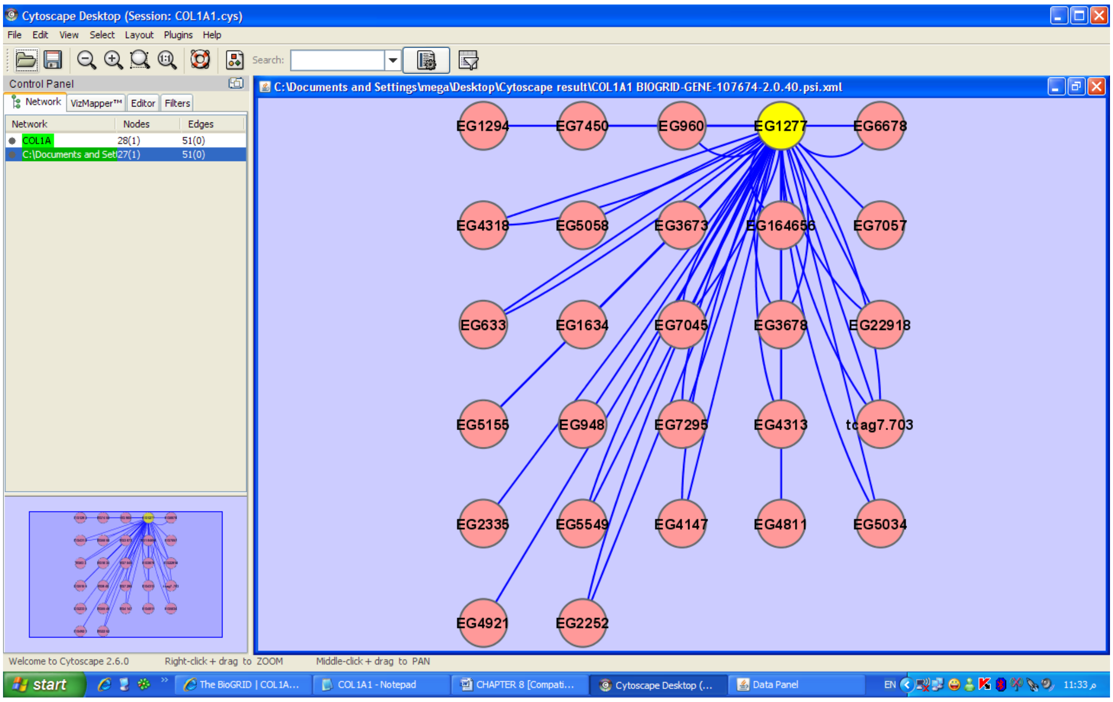
Figure (7-1): COL1A1 Network generated by creating network manually.

The second stage: - retrieving the interaction prosperities of Collagen, type I (COL1A1) from BioGRID database. The interactions of Collagen, type I (COL1A1) are downloaded as a flat file from BioGRID database. Then we use Cytoscape to Create Networks by importing pre-existing, formatted network files. **COL1A1 was identified with 51 protein interactions as shown in** Table (7-7). COL1A1 pathways by Importing Fixed-Format Network Files will be explained in Figure (7-2).

Table (7-7): Protein interactions of COL1A1 identified by BioGRID

Name	Description
ITGA2	Integrin, alpha 2
DCN	decorin
TGFBI	transforming growth factor, beta-induced, 68kDa;
IGFBP3	Insulin-like growth factor binding protein 3

ITGA5	Integrin, alpha 5 (fibronectin receptor, alpha polypeptide)
SPARC	osteonectin
TXN	thioredoxin
BGN	biglycan
TMPRSS6	transmembrane protease, serine 6
MMP9	matrix metallopeptidase 9
C1QR1	CD93 molecule
MMP2	matrix metallopeptidase 2
PDGFB	platelet-derived growth factor beta polypeptide
CD44	CD44 molecule (Indian blood group)
P4HB	procollagen-proline, 2-oxoglutarate 4-dioxygenase
MATN2	matrilin 2
FGF7	fibroblast growth factor 7|keratinocyte growth factor
PRELP	proline/arginine-rich end leucine-rich repeat protein
VWF	coagulation factor VIII VWF; von Willebrand factor
COL7A1	collagen, type VII, alpha 1
CD36	CD36 molecule (thrombospondin receptor)
PAK1	p21/Cdc42/Rac1-activated kinase 1
THBS1	thrombospondin-1p180; thrombospondin 1
DDR2	discoidin domain receptor family, member 2
NID	enactin|entactin; nidogen 1
FN1	fibronectin 1

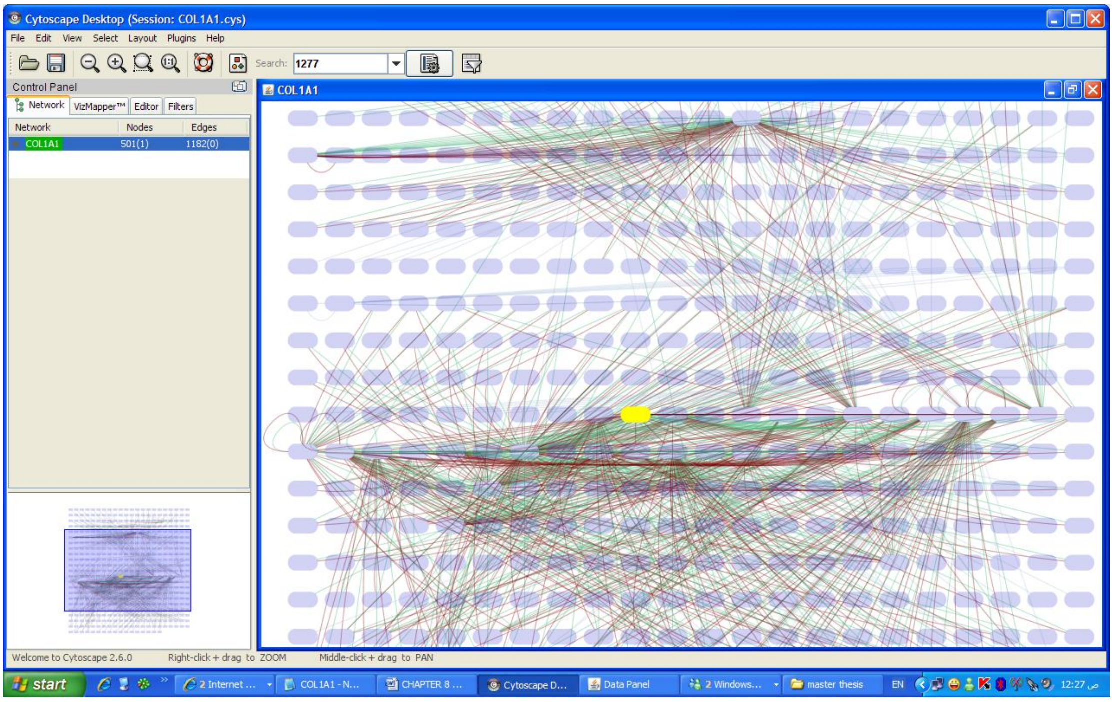


Network	Nodes	Edges
COL1A	28(1)	51(0)
C:\Documents and Set	27(1)	51(0)

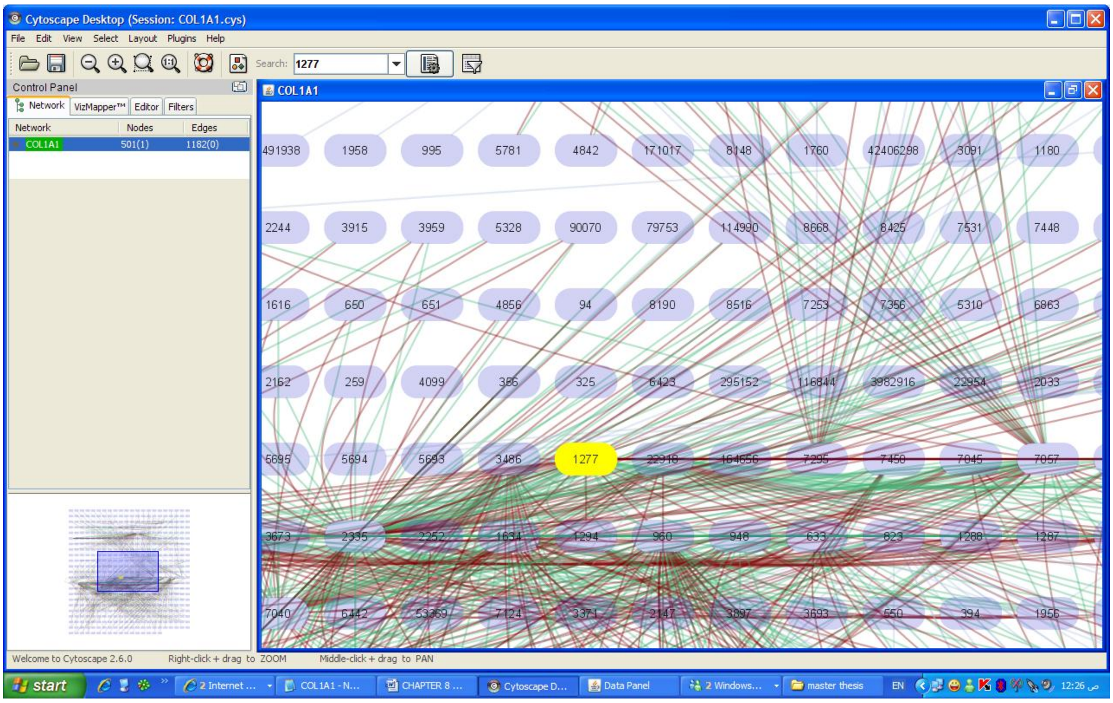
Figure (7-2): COL1A1 Network generated by Importing Fixed-Format Network Files

The Third stage: - using Cytoscape to Creating Networks by importing networks from Web Service. We will retrieve Protein-Protein Interaction Networks from NCBI Entrez Gene. NCBI web service client uses this section to build networks. Network generated from Entrez Gene data: The network below Figure (7-3) is generated from interaction data matching the keyword *Homo sapiens*. Edge color represents data source type (BIND, BioGRID, or HPRD). NCBI client extracts interaction data from a huge dataset,

Downloading Pathways and Interaction Networks:-


Cytoscape Desktop (Session: COL1A1.cys)
File Edit View Select Layout Plugins Help
Search: 1277
Control Panel
Network VizMapper™ Editor Filters
COL1A1
Welcome to Cytoscape 2.6.0
Right-click + drag to ZOOM
Middle-click + drag to PAN
start
2 Internet ...
COL1A1 - N...
CHAPTER 8 ...
Cytoscape D...
Data Panel
2 Windows...
master thesis
EN
12:27 ص

Network	Nodes	Edges
COL1A1	501(1)	1182(0)



Network	Nodes	Edges
COL1A1	501(1)	1182(0)

Figure (7-3): COL1A1 Network generated by Entrez Gene data

7.2.2.3.2 Transforming Growth Factor, beta 1 (TGFB1)

The first stage: - Gathering the extracted result of the PIELG system for transforming growth factor, beta 1 **(TGFB1**) we found that **TGFB1**interacts with several proteins as shown in Table (7-8).

Table (7-8): Protein interactions identify by PIELG for TGFB1

Name	**Description**
DCN	decorin
BMP3	**bone morphogenetic protein 3 (osteogenic)**
BGN	biglycan
MMP2	matrix metallopeptidase 2
ITGB8	**integrin, beta 8**

We used Cytoscape to create an empty network and manually add nodes and edges. **We draw Networks** for the extracted interactions of transforming growth factor, beta 1 **(TGFB1**). Proteins are represented by nodes, and interactions are represented by edges between nodes. The following (Figure (7-4)) represented the final network.

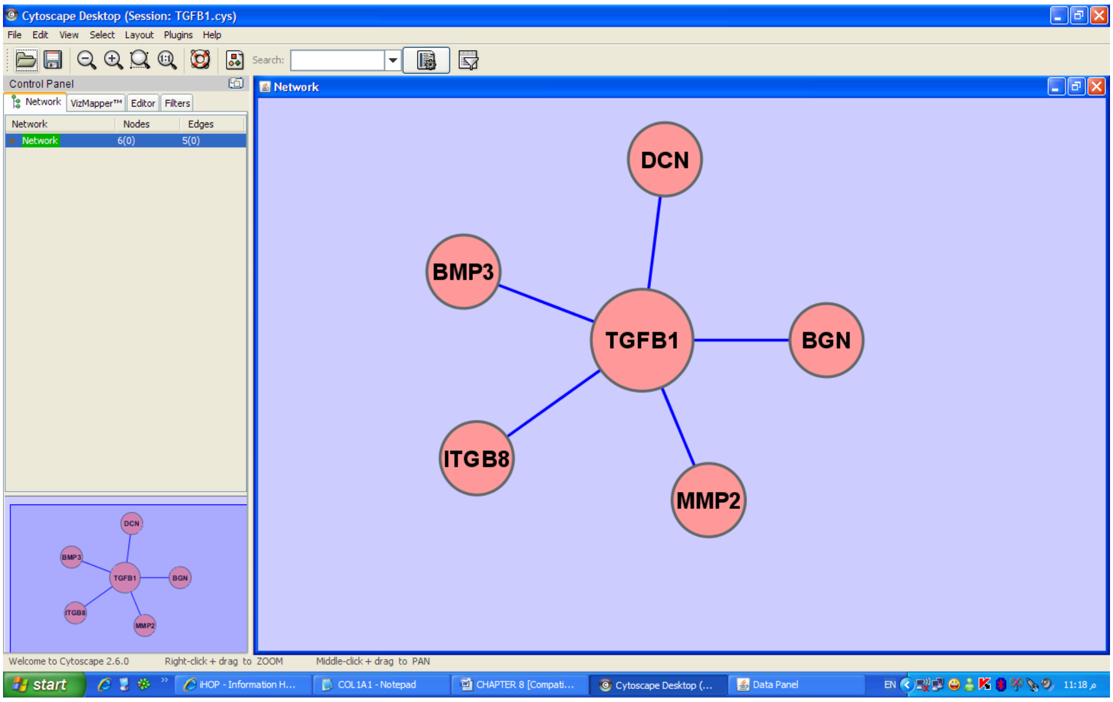


Figure (7-4): TGF-beta-1 Network generated by creating network manually.

The second stage: - retrieving the interaction prosperities of transforming growth factor, beta 1 (TGFB1) from BioGRID database. The interactions of transforming growth factor, beta 1 (TGFB1) are downloaded as a flat file from BioGRID database. Then we use Cytoscape to Creating Networks by importing pre-existing, formatted network files. TGFB1 was identified with 60 protein interactions as shown in Table (7-9).COL1A1 pathways by Importing Fixed-Format Network Files will be explained in Figure (7-10).

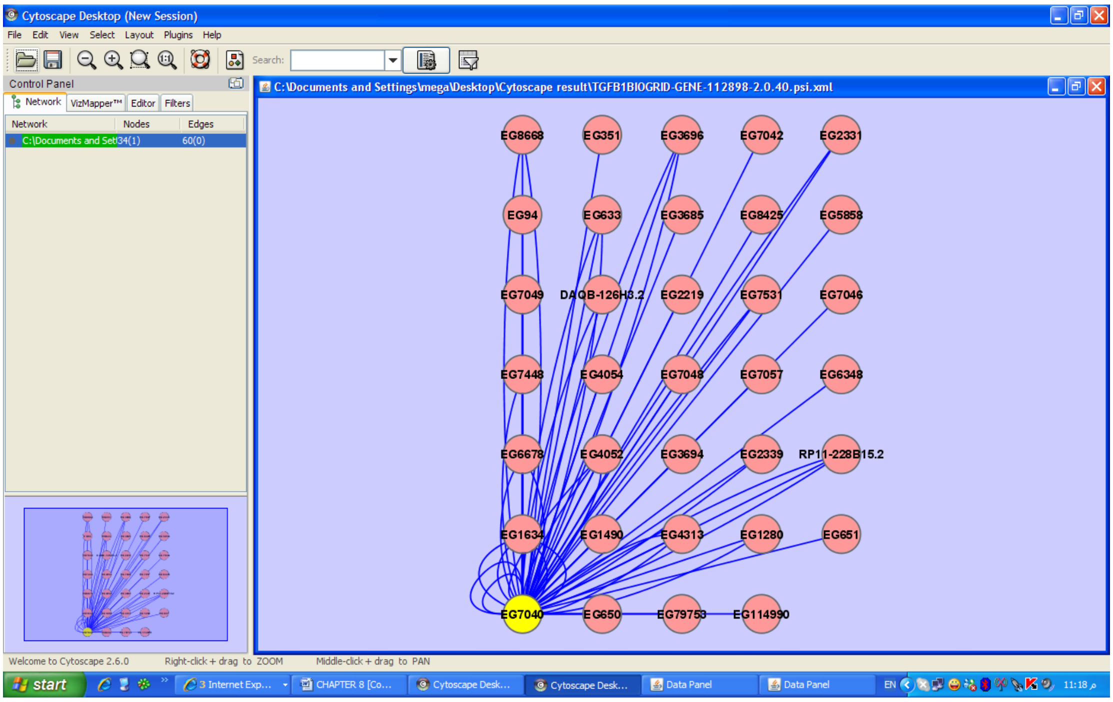


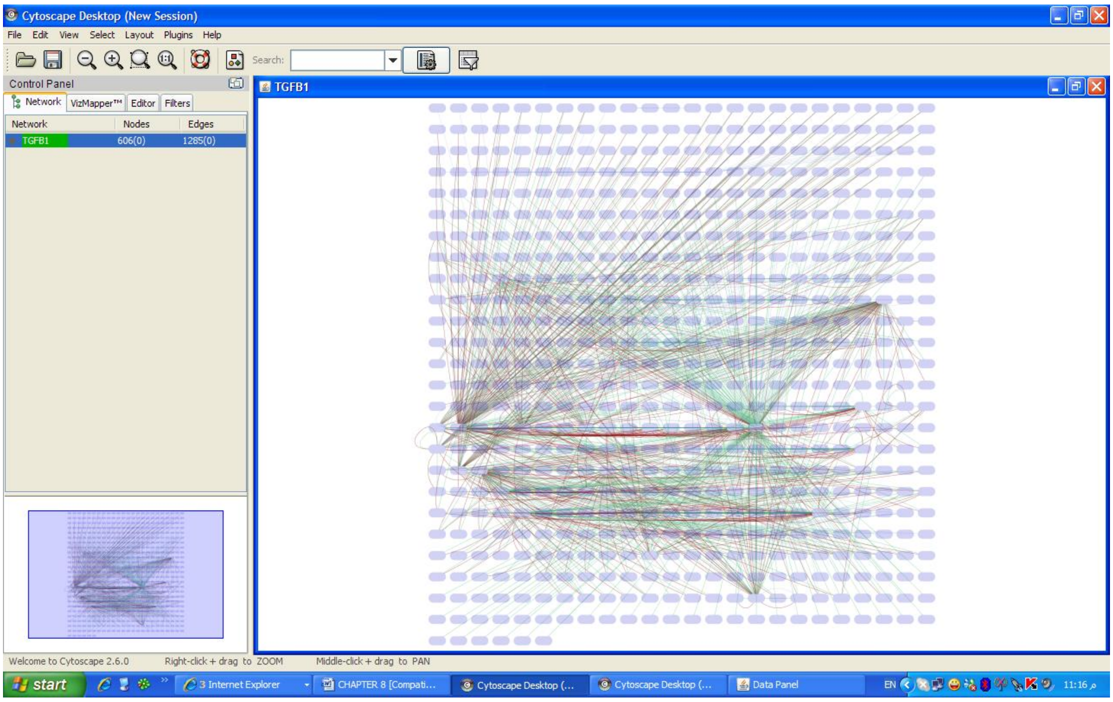
Figure (7-5)**:TGFB1 Network generated** by Importing Fixed-Format Network Files

Table (7-9): Protein interactions of **TGFB1** identified by BioGRID

Name	Description
DCN	decorin
LTBP1	TGF-beta1-BP-1
BGN	biglycan
SPARC	secreted protein, acidic, cysteine-rich (osteonectin)
EIF3S2	eukaryotic translation initiation factor 3, subunit I
ENG	endoglin (Osler-Rendu-Weber syndrome 1); endoglin
MMP2	matrix metallopeptidase 2
ITGB8	integrin, beta 8
DAXX	death-associated protein 6
TGFB1	transforming growth factor, beta 1

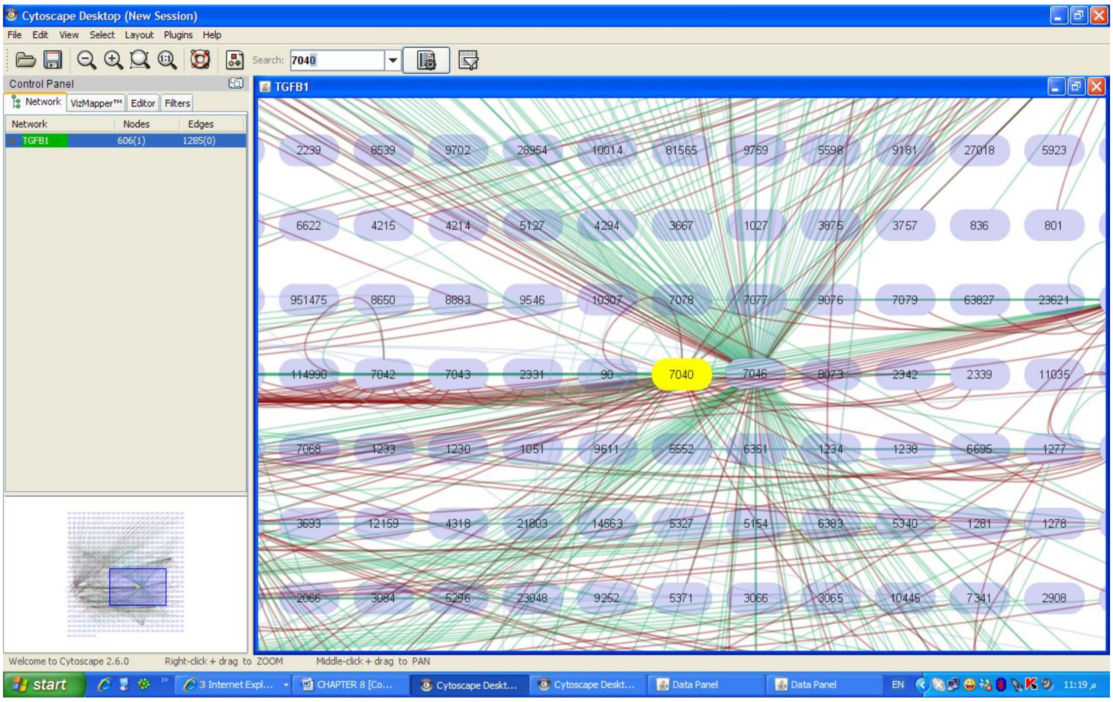
FMOD	fibromodulin proteoglycan; fibromodulin
YWHAE	tyrosine 3-monooxygenase/tryptophan
VTN	vitronectin
COL2A1	collagen, type II, alpha 1
FNTA	farnesyltransferase, CAAX box, alpha
ACVRL1	activin A receptor type II-like 1
CTGF	connective tissue growth factor
TGFB2	transforming growth factor, beta 2
BMP3	bone morphogenetic protein 3 (osteogenic)
SNIP1	Smad nuclear interacting protein 1
SLITL2	slit-like 2; vasorin
APP	amyloid beta (A4) precursor protein
ITGAV	integrin, alpha V
ITGB6	integrin, beta 6
FCN1	ficolin (collagen/fibrinogen domain containing) 1
BMP2	bone morphogenetic protein 2
TGFBR1	transforming growth factor, beta receptor I
LTBP3	latent TGF beta binding protein 3
THBS1	thrombospondin-1p180; thrombospondin 1
TGFBR3	betaglycan proteoglycan
PZP	pregnancy-zone protein; Pregnancy zone protein
LTBP4	latent transforming growth factor beta binding protein 4
CCL3	chemokine (C-C motif) ligand 3
TGFBR2	transforming growth factor, beta receptor II (70/80kDa)

The third stage: - using Cytoscape to Creating Networks by importing networks from Web Service. We will retrieve Protein-Protein Interaction Networks from NCBI Entrez Gene. NCBI web service client uses this section to build networks. Network generated from Entrez Gene data: The network below Figure (7-6) is generated from interaction data matching the keyword *Homo sapiens*. Edge color represents data source type (BIND, BioGRID, or HPRD). NCBI client extracts interaction data from a huge dataset.



Network	Nodes	Edges
TGFB1	606(0)	1285(0)

Welcome to Cytoscape 2.6.0   Right-click + drag to ZOOM   Middle-click + drag to PAN



Search: 7040

Network	Nodes	Edges
TGFB1	606(1)	1285(0)

Welcome to Cytoscape 2.6.0   Right-click + drag to ZOOM   Middle-click + drag to PAN

Figure (7-6):**TGFB1 Network generated** by **Entrez Gene data**

7.2.2.3 Dentin Matrix Protein -1 (DMP-1)

The first stage: - Gathering the extracted result of the PIELG system for dentin matrix protein 1 (DMP-1) we found that DMP-1interacts with several proteins as shown in Table (7-10).

Table (7-5): Protein interactions identify by PIELG for DMP-1

Name	Description
CFH	**complement factor H**
DSPPP	**DSPP Promoter**
CD44	**CD44 molecule (Indian blood group)**

We used Cytoscape to create an empty network and manually add nodes and edges. **Then, we draw Networks** for the extracted interactions of dentin matrix protein 1 (DMP-1). Proteins are represented by nodes, and interactions (or other biological relationships) are represented by edges between nodes. The following figure (Figure (7-7)) represented the final network.

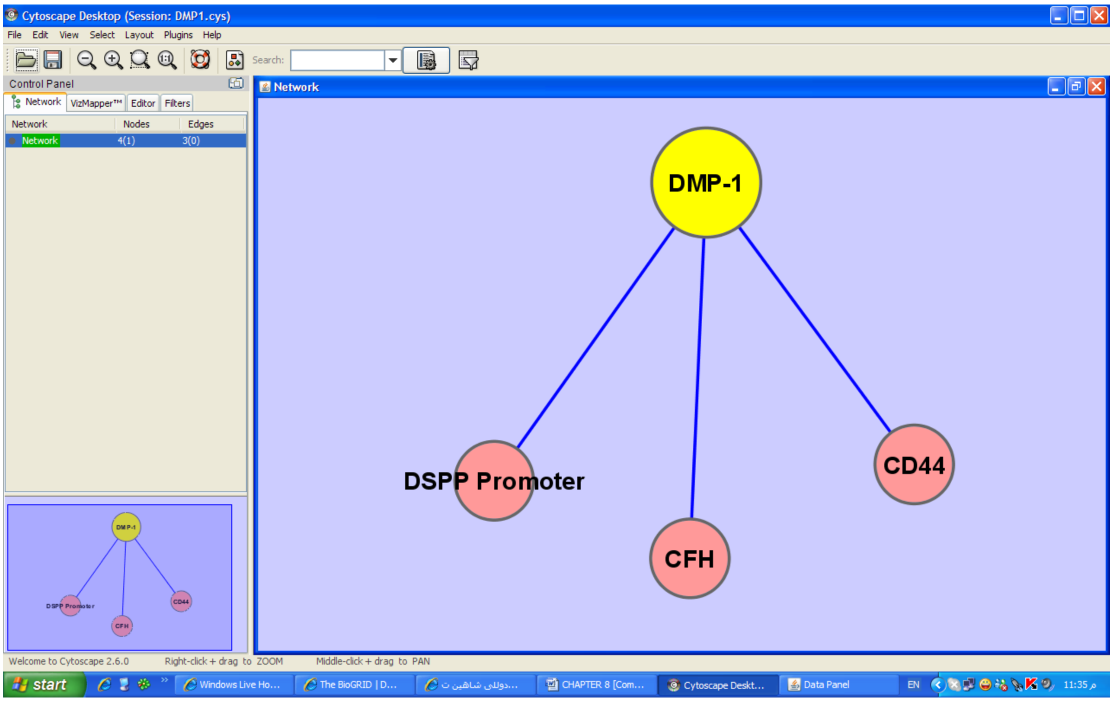
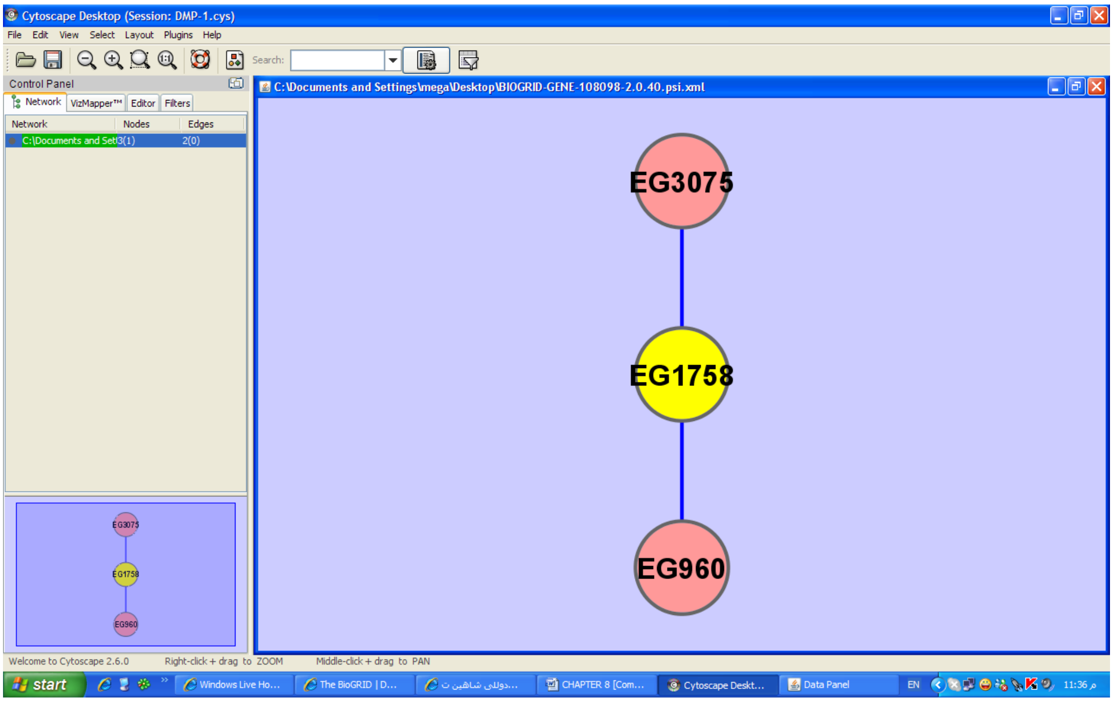

Cytoscape Desktop (Session: DMP1.cys)
File Edit View Select Layout Plugins Help
Search:
Control Panel
Network VizMapper™ Editor Filters
Network Nodes Edges
Network 4(1) 3(0)
Network
DMP-1
DSPP Promoter
CFH
CD44
Welcome to Cytoscape 2.6.0
Right-click + drag to ZOOM
Middle-click + drag to PAN
start
Windows Live Ho...
The BioGRID | D...
CHAPTER 8 [Com...
Cytoscape Deskt...
Data Panel
EN
11:35

Figure (7-7): **DMP-1 Network generated** by creating network manually.

The second stage: - retrieving the interaction prosperities of dentin matrix protein 1 (DMP-1) from BioGRID database. The interactions of dentin matrix protein 1 (DMP-1) are downloaded as a flat file from BioGRID database. Then we use Cytoscape to Creating Networks by importing pre-existing, formatted network files. DMP-1 **was identified with 2 protein interactions as shown in** Table (7-11)**.** DMP-1 pathways by Importing Fixed-Format Network Files will be explained in Figure (7-8).

Table (7-11): Protein interactions of DMP-1identified by BioGRID

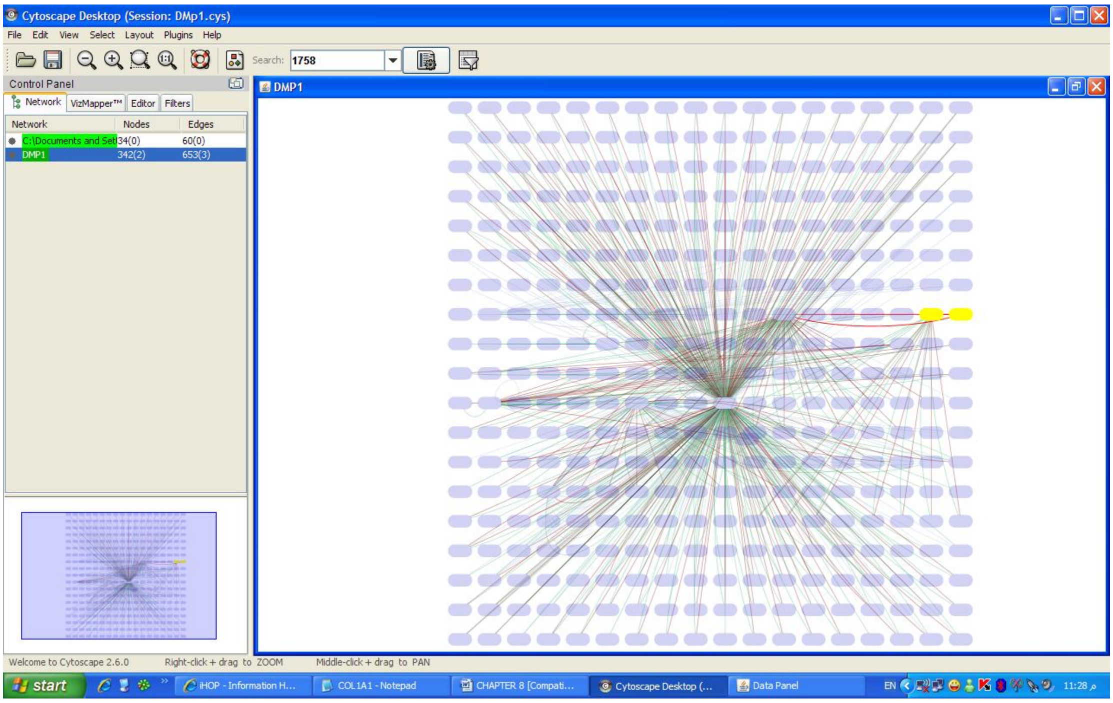
Name	Description
CFH	complement factor H
CD44	CD44 molecule (Indian blood group)


Cytoscape Desktop (Session: DMP-1.cys)
File Edit View Select Layout Plugins Help
Search:
Control Panel
Network VizMapper™ Editor Filters
C:\Documents and Settings\mega\Desktop\BIOGRID-GENE-108098-2.0.40.psi.xml
EG3075
EG1758
EG960
Welcome to Cytoscape 2.6.0
Right-click + drag to ZOOM
Middle-click + drag to PAN
start

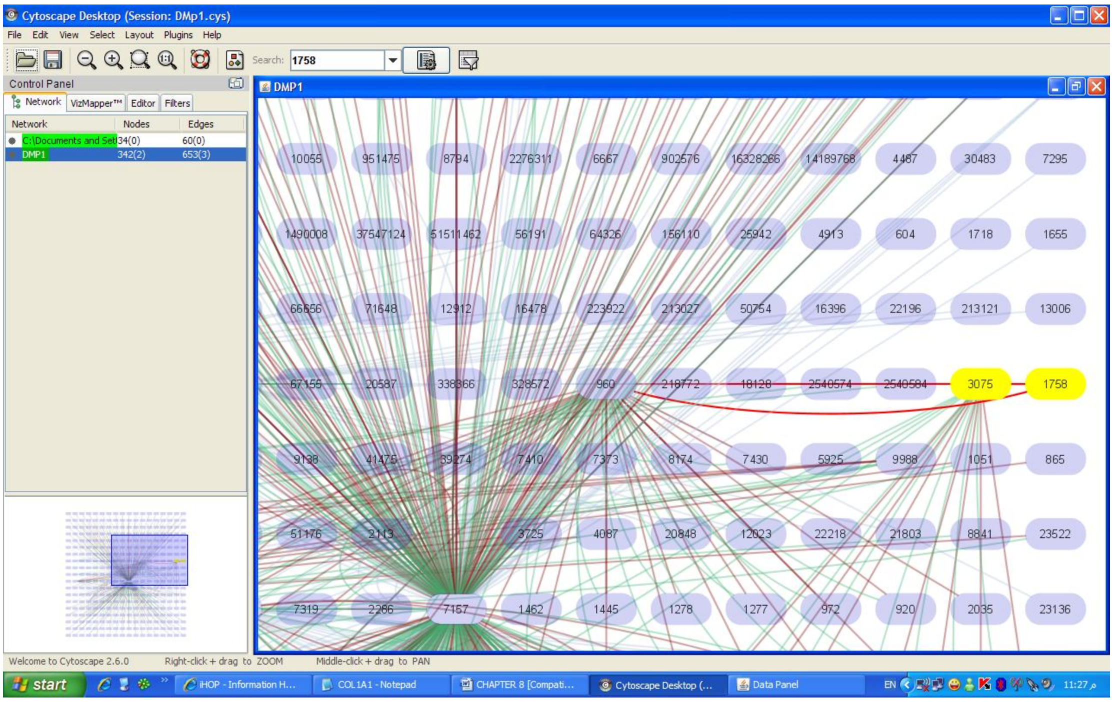
Network	Nodes	Edges
C:\Documents and Set	3(1)	2(0)

Figure (7-8): DMP-1 Network generated by Importing Fixed-Format Network Files

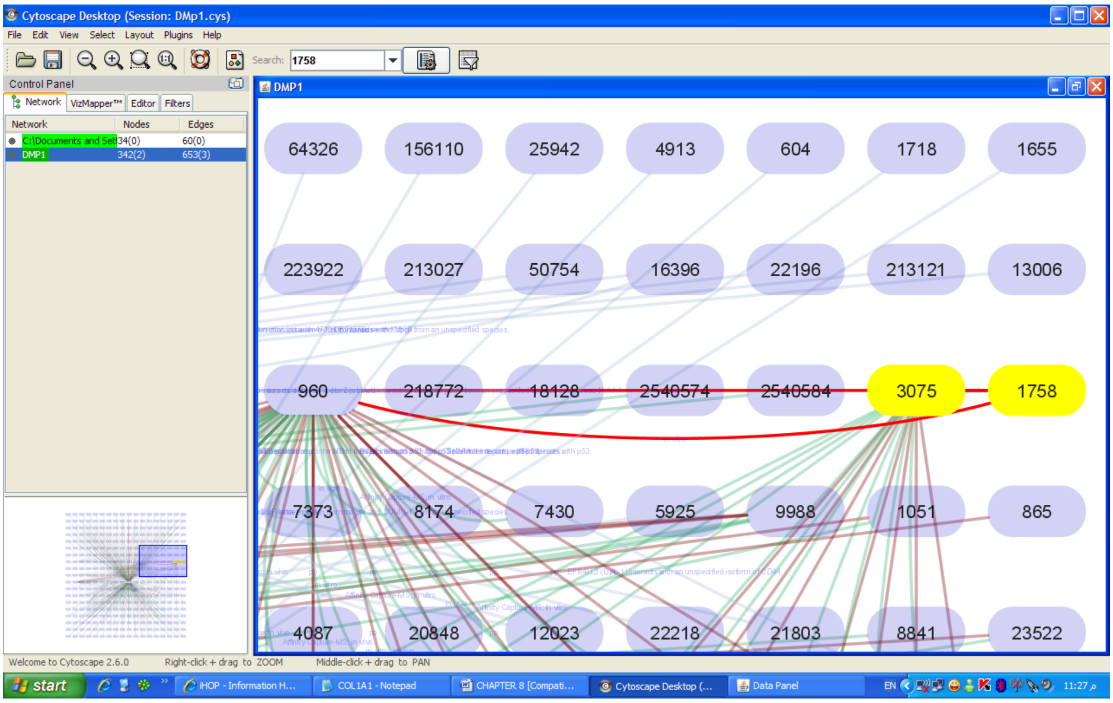
The Third stage: -


Cytoscape Desktop (Session: DMp1.cys)
File Edit View Select Layout Plugins Help
Search: 1758
Control Panel
Network VizMapper™ Editor Filters
DMP1
Welcome to Cytoscape 2.6.0
Right-click + drag to ZOOM
Middle-click + drag to PAN
start

Network	Nodes	Edges
C:\Documents and Set	34(0)	60(0)
DMP1	342(2)	653(3)



Network	Nodes	Edges
C:\Documents and Sett	34(0)	60(0)
DMP1	342(2)	653(3)



Network	Nodes	Edges
C:\Documents and Sett	34(0)	60(0)
DMP1	342(2)	653(3)

Figure (7-9): DMP-1 Network generated by Entrez Gene data

7.3 Discussion about the second Phase

BioGRID contains protein interactions from both abstracts and full text. Since PIELG system is tested only on the abstracts, the system misses out on some interactions that are only present in the full text of the abstract. If those interactions are excluded, PIELG can have a higher recall. However, a lack of a standard common corpus and a lack of standard techniques and equations for reporting recall and precision have made comparative analysis of different techniques a difficult problem [94]**.**

CHAPTER 8

Conclusion and Future work

8.1 Conclusions

The goal of this thesis was to design a tool for automatically extracting information about protein-protein interactions from biomedical papers written in natural language and retrieved in electronic format from databases such as the PubMed. The solution was to use a method called information extraction (IE) that deployed sophisticated natural language processing (NLP) techniques to analyze the syntax of the text and to extract information using rule-based scenario patterns. This thesis presents a protein–protein interaction extraction system specially designed to process biomedical literature– PIELG. The distinguishing feature of PIELG is that it introduces extraction of interactions via Link Grammar Parser. PIELG is based on a deep parse tree structure presented by the Link Grammar and it considers a thorough case based analysis of contents of various syntactic roles of the sentences as well as their linguistically significant and meaningful combinations.

Confined to the complicity of natural language, extracting protein-protein interactions from biomedical literatures is a challenging task and it is difficult to achieve a good performance. However, we have developed and evaluated PIELG, for analysis of biomedical literature. From the results of the PIELG evaluation process, we can conclude that its performance is satisfactory for the real-time PubMed processing. The results also shows that syntactic role-based approach compounded with linguistically sound interpretation rules applied on the full sentence's parse can achieve better performance than existing systems which are based on manually engineered patterns. Those systems are both costly to develop and are not as scalable as the automated mechanisms presented in this thesis.

In conclusion, PIELG is high precision information extraction system capable of extracting various types of protein function information. Semantic Parsers for English language will be more useful and meaningful for extraction task compared to Syntactic parsers. But constructing semantic parser is a difficult task and this parser will be more domains dependent. General literature mining at the semantic level is not technically feasible. For simpler problems in a restricted domain, such as the protein–protein interaction, our theory works just fine. It is not possible to compare results due to differing datasets and the limited information available about their methods.

Utilization of protein names dictionary provides an ability to change the scope of extracted information, making entire system more flexible, and along with high performance, favorably differentiates it from the other systems. The high precision of the PIELG stems from its full-sentence parsing approach and presently comes at the price of a lower recall rate. However, the volume of data can be increased several times by implementing a reasonable set of improvements to the system, extending the protein names dictionary towards the description of experimental data.

We estimate that the current PIELG**'s** coverage rate could be enhanced by increasing the lexicon size of the Link Grammar Parser, improving its quality, and by slightly improving its grammar. In addition, even with its coverage PIELG is still immediately applicable for an information extraction task. However, several issues make extracting such interactions and relationships difficult since**:**

1. The task involves free text - hence there are many ways of stating the same fact.
2. The genre of text is not grammatically simple.
3. The text includes a lot of technical terminology unfamiliar to existing natural language processing systems.
4. Information may need to be combined across several sentences.
5. There are many sentences from which nothing should be extracted.

Since writers are apt to keep their own writing styles throughout the whole paper, more training corpus is needed to learn the various extraction rules to cope with the various style differences across the several papers.

So we recommend to the publisher to advice authors to submit their papers in simple and clear sentences to be easily used in information extraction systems. The authors should avoid using compound or complex sentences especially those sentences that ensure the interactions between two proteins. For Example:- the sentence *"We also observed that the expression of DSPP and DMP-1 was induced by TGF-beta3 in primary cultured dental pulp cells, however, not in calvaria osteoblasts, whereas OCN, osteopontin and osteonectin expression was increased after treatment with TGF-beta3 in both dental pulp cells and calvaria osteoblasts."* It is better to divide this sentence into two simple sentences.

8.2 Future Work

PIELG system can be further enhanced with the following features:

8.2.1 Anaphora Resolution:

There are some additional linguistic features notable about required patterns like anaphora, i.e. substitution of pronouns for actual words. To extract interactions from the all sentences, the anaphor are to be resolved first. Sentences including anaphora have property different classes. There is also a feature of optional prepositional phrases, which include one of entities. The entity is somehow the semantic subject of an interaction verb, but it is hard to find out by parsing. Anaphora handling is still future work. A co-reference resolution module is a necessary part of any information extraction system. In our system we plan to extract anaphors and pronominal anaphors as they are very common in abstracts. For example:- the sentences " *Dentin sialoprotein (DSP) and phosphophoryn (PP) are the two noncollagenous proteins classically linked to dentin but more recently found in bone, kidney, and salivary glands. These two proteins are derived from a single copy DSP-PP gene*. "Anaphora is used to resolve the pronoun "These".

8.2.2 Handling negations

Negations in the sentences (such as “not interact”, “fails to induce”, “does not inhibit”) is also an area we have to look into. Negations in most of the abstracts talk about an experiment done to some action but that has resulted in failure. Extracting negations might be of real use to biologist to share their failures so that similar experiments can be avoided or some lessons learnt from these failures (“shared failures”).

8.2.3 Handling more forms

PIELG system can be further enhanced with enhancement to its rules to handle the following forms:

- **Gerund in prepositional phrases.**
    - The *ENTITY1* plays key roles by activating *ENTITY2*.

- **Relative clauses.**
    - *ENTITY1* **that binds specificity with** *ENTITY2*

- **Anaphora**
    - *ENTITY1* **and its binding specificity with** *ENTITY2.*

- **Coordination**
    - This study demonstrates that *ENTITY1* **recognizes** *ENTITY2a* and *ENTITY2b*.

- **Process Negative Sentences by constructing the following patterns of regular expression:**
    - *ENTITY1* * not (interact | associate | bind | complex) * *ENTITY2.*
    - *ENTITY1* does not interact with *ENTITY2* or *ENTITY2a*
    - *ENTITY1* * PATTERN. * but not *ENTITY2*
    - *ENTITY1* interacts with another *ENTITY2* family member *ENTITY2a*, but not with *ENTITY*2b.

- **Other forms.**
    - *ENTITY1* is an Optional Prepositional Phrase

- **Unlike** human *ENTITY1*, the viral cytokine uses hydrophobic amino acids to **contact** *ENTITY2*.

We also need to Identify relationships among interactions extracted from a collection of sentences (such as one interaction stimulating or inhibiting another) to construct a “Large Protein Interaction Pathways” from abstracts and full text articles.

REFERENCES

1. Rania A. Abul Seoud, Nahed H. Solouma, Abou-Bakr M. Youssef, Yasser M. Kadah, “General Domain-Oriented Engine Based on A Link Grammar Parser Used To Extract Protein Interactions,” *Proc. 3rd Cairo International Biomedical Engineering Conference*, Cairo, Dec. 2006.
2. Rania Abulseoud, Abou-Bakr Youssef and Yasser M. Kadah, “Extraction of protein interaction information from unstructured text using a link grammar parser,” Proc. of the 2007 International Conference on Computer Engineering & Systems (ICCES'07), Cairo, Egypt, November 2007.
3. “MEDLINE - National Library of Medicine (NLM),” *National Institutes of Health (NIH)*, http://www.nlm.nih.gov. 1993.
4. “PubMed Centeral,”*National Center for Biotechnology Information (NCBI),* http://www.ncbi.nlm.nih.gov/sites/entrez/. 1988.
5. Isaaq H.J., Veenstra T.D., Conrads T.P.," The SELDI-TOF MS approach to proteomics: protein profiling and biomarker identification", biochem. Biophyus. Ress. Commun, 292:587-96, 2002.
6. Fung ET, Enderwick C," Protein chip clinical proteomics: computational challenges and solutions", Biotechniguies, 34:40-50, 2002.
7. Dayhoff M.O.," Atlas of protein sequence and structure", national biochemical Research Foundation, vol. 5, Georgetown University, Washington, D. C. 1972.
8. Mount D. W.," Sequence and Genome Analysis", Bioinformatics, Gold Spring Harbor Laboratory press, 1-1-8, 2001.
9. Marei M.K., Nouh S.R., Fata M.M., Faramawy A. M. ,"Fabrication of polymer root from scaffold to be utilized for alveolar bone regulation", tissue engineering. 9:713-731. 2003.
10. Cate, A.R. Ten. *Oral Histology: development, structure, and function.* 5th ed. Page 150. ISBN 0-8151-2952-1.1998.

11. Goldszmidt, M., and Sahami, M., "A probabilistic approach to full-text document clustering," Technical Report ITAD-433-MS-98-044, SRI International, 1998.

12. Ian H. Witten, Alistair Moffat and Timothy C. Bell, "Textual Images," Managing Gigabytes: Compressing and Indexing Documents and Images, Chapter 7, New York:Van Nostrand Reinhold, pp. 254-293, 1994.

13. Allen, J.F. "Natural Language Understanding, Benjamin Cummings," 1987, Second Edition, 1994.

14. Francis, W.N., and Kucera, H., "Brown Corpus Manual," *www.hit.uib.no/icame/brown/bcm.html*, 1979.

15. Merialdo, B., "Tagging English text with a probabilistic model," *Computational Linguistics* 22(2), 155–172, 1994.

16. Marcus, M. "The Penn Treebank Project," *www.cis.upenn.edu/~treebank*, 1992.

17. Brill, E. "A simple rule-based part of speech tagger," *Proc. 3rd Ann. Conf. on Applied Natural Language Processing, ACL*. 1992.

18. Maltese, G., and Mancini, F. ,"A technique to automatically assign parts-of-speech to words taking into account word-ending information through a probabilistic model," *Proc. of Eurospeech-91*, 753–756. 1991.

19. Hindle, D., "Acquiring disambiguation rules from text,"*Proc. 27th Annual Meeting of the Association for Computational Linguistics*. 1989.

20. Appelt, D.E., and Israel, D.J., "Introduction to information extraction technology," A tutorial prepared for the International Joint Conference on Artificial Intelligence (IJCAI-99). *www.ai.sri.com/~appelt/ie-tutorial/IJCAI99.pdf*. 1999.

21. Hobbs, J.,"Resolving pronoun references, in *Readings in Natural Language Processing*," 339–352. Morgan-Kaufmann, Los Altos, CA. 1986.

22. Grishman, R. "The NYU system for MUC-6 or where's the syntax," *Proc. 6th Message Understanding Conference (MUC-6)*, Columbia, Maryland, 1995.

23. Hacioglu, Kadri, Sameer Pradhan, Wayne Ward, James Martin, and Daniel Jurafsky. ,"Semantic role labeling by tagging syntactic chunks," In *Proceedings of the 8th Conference on CoNLL-2004, Shared Task. Semantic Role Labeling*, 2004.
24. Atkins, Sue, Michael Rundell and Hiroaki Sato, "The Contribution of Framenet to Practical Lexicography," International Journal of Lexicography, Volume 16.3: 333-357, 2003.
25. Shatkay Hagit, "Mining the biomedical literature: State of the art, challenges and evaluation," Tutorial, ISMB’05, 2005.
26. “LocusLink - Database of genes,” *National Center for Biotechnology Information (NCBI),* http:// www.ncbi.nlm.nih.gov/sites/ entrez?db=gene. 1988.
27. “Universal Protein Resource (UniProt),” *European Bioinformatics Institute (EBI), the Swiss Institute of Bioinformatics (SIB) and the Protein Information Resource (PIR),* http://beta.uniprot.or g. 2002.
28. Blaschke, C., *et al.* ,"Automatic extraction of biological information from scientific text: Protein–protein interactions," *Proc. AAAI Conf. on Intelligent Systems for Molecular Biology (ISMB)*, 60–67, 1999.
29. Fukuda, K., *et al.,* "Toward information extraction: Identifying protein names from biological papers. *Proc. Pacific Symposium on Biocomputing (PSB)*, 705–716, 1998.
30. S. Sekine, R. Grishman, and H. Shinnou, "A decision tree method for finding and classifying names in Japanese texts," In Proceedings the Sixth Workshop on Very Large Corpora, 1998.
31. A. Borthwick, J. Sterling, E. Agichtein, and R. Grishman, "Exploiting diverse knowledge sources via maximum entropy in named entity recognition," In Proceedings of the Sixth Workshop on Very Large Corpora, Montreal, 1998.

32. A. Mikheev and C. Grover, "LTG: Description of the NE recognition system as used for MUC-7," In Proceedings of the Seventh Message Understanding Conference, 1998.

33. D. M. Bikel, S. Miller, R. Schwartz, and R. Weischedel. Nymble, "a high performance learning name-finder," In Proceedings of the Fifth Conference on Applied Natural Language Processing, 1997.

34. M. Asahara and Y. Matsumoto, "Japanese Named Entity Extraction with Redundant Morphological Analysis," In Proceedings of Human Language Technology Conference(HLT-NAACL), 2003

35. Meenakshi Narayanaswamy, K. E. Ravikumar, and K. Vijay-Shanker, "A biological named entity recognizer," In Pacific Symposium on Biocomputing, 2003.

36. Lorraine Tanabe and W. John Wilbur, "Tagging gene and protein names in biomedical text," Bioinformatics, 18(8):1124–1132, 2002.

37. Zhou GuoDong and Su Jian, "Exploring deep knowledge resources in biomedical name recognition," In Proceedings of 2004 Joint Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA'2004 shared task), pages 99–102, 2004.

38. Pyysalo S, Ginter F, Heimonen J, Björne J, Boberg J, Järvinen J, Salakoski T, "BioInfer: A corpus for information extraction in the biomedical domain," BMC Bioinformatics 2007, 8:50.

39. Jin-Dong Kim, Tomoko Ohta, Yuka Tateisi, and Jun'ichi Tsujii. ,"GENIA corpus—a semantically annotated corpus for bio-text mining," *Bioinformatics*, 19:i180–182, 2003.

40. Pyysalo S, Ginter F, Laippala V, Haverinen K, Heimonen J, Salakoski T, "On the unification of syntactic annotations under the Stanford dependency scheme: A case study on BioInfer and GENIA," In Proceedings of the ACL BioNLP'07 Workshop. Prague, Czech Republic. 2007.

41. Swales, John M. Genre Analysis, "English in academic and research settings," The Cambridge Applied Linguistics Series. Cambridge University Press, 1997.

42. Stapley, B.J., and Benoit, G. Bibliometrics, "Information retrieval and visualization from co-occurrences of genenames in medline abstracts," *Proc. Pacific Symposium on Biocomputing (PSB),* 526–537, 2000.

43. Friedman, C., *et al.*, " Genies: A natural-language processing system for the extraction of molecular pathways from journal articles, " *Proc. Int. Conf. on Intelligent Systems for Molecular Biology (ISMB),* S74–S82. 2001

44. Stephens, M., *et al.*, "Detecting gene relations from medline abstracts," *Proc. Pacific Symposium on Biocomputing (PSB),* 483–496, 2001.

45. Sebastiani, F., "Machine learning in automated text categorization. *ACM Computing Surveys* 34(1), 1–47, 2002.

46. Blaschke, C., and Valencia, A., "The frame-based module of the SUISEKI information extraction system. *IEEE Intelligent Systems, Special Issue on Intelligent Systems in Biology*, 17(2), 14–20, 2002.

47. Swanson, D.R., "Somatomedin C and arginine: Implicit connections between mutually isolated literatures. *Perspectivesin Biology and Medicine* 33(2), 157–186, 1990.

48. Sekimizu, T., Park, H.S. and Tsujii,J. , "Identifying the Interaction between Genes and Gene Products Based on Frequently Seen Verbs in Medline Abstracts. Genome Informatics Workshop, 62–71, 1998.

49. Ng, See-Kiong and Wong, Marie, "Toward routine automatic pathway discovery from on-line scientific text abstracts. Genome Informatics, 10:104{112, December 1999.

50. Ono,T., Hishigaki,H., Tanigami,A. and Takagi,T, " Automated extraction of information on protein–protein interactions from the biological literature. Bioinformatics, 17(2), 155–161, 2001.

51. Yu Hao, Xiaoyan Zhu, Minlie Huang and Ming Li, "Discovering patterns to extract protein–protein interactions from the literature: Part II ", *Vol. 21*

*no. 15 2005, pages 3294–3300, doi:10.1093/bioinformatics/bti493,*May 12, 2005.

52. Koike, A., Kobayashi,Y. and Takagi,T. ,"Kinase pathway database: an integrated protein-kinase and NLP-based protein-interaction resource," *Genome Res.*, 13, 1231–1243, 2003.
53. Thomas J., Milward D., Ouzounis C., Pulman S., and Carroll M. ,"Automatic extraction of protein interactions from scientific abstracts," In Proceedings of the pacific Symposium on biocomputing, pages. 2000.
54. Park J.C., Kim H.S., and Kim J.J. ,"Bidirectional incremental parsing for automatic pathway identification with combinatory categorial grammar," In Proceedings of Pacific Symposium on Biocomputing, pages 396-407, Hawaii,. 2001.
55. Park, J.C., "Using combinatory categorial grammar to extract biomedical information", Intelligent Systems, IEEE Volume 16, Issue 6, Page(s): 62 – 67 Digital Object Identifier 10.1109/5254.972092, Nov-Dec 2001.
56. J.M. Temkin and M.R. Gilder, "Extraction of Protein Interaction Information from Unstructured Text Using a Context-Free Grammar," 2003GRC070, April 2003.
57. Yakushiji A, Tateisi Y, Miyao Y, Tsujii J**,** "Event extraction from biomedical papers using a full parser," *Pac Symp Biocomput*, 408-419, 2001.
58. Nanda, K., "Combining lexical, syntactic and semantic features with maximum entropy models for extracting relations," Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-2004), (2004)
59. Huang, M., et al, "Discovering patterns to extract protein-protein interactions from full texts," Bioinformatics, 20, 3604-3612, 2004.
60. Craven, M., and Kumlien, J., "Constructing Biological Knowledge Bases by Extracting Information from Text Sources, Proceeding of the 7th

International Conference on the Intelligent System for molecular Biology (ISMB-99),: 77-86, 1999.

61. J. Xiao, J. Su, G. Zhou, and C. Tan, “Protein-protein interaction extraction: a supervised learning approach,” in *Proceedings of the 1st Symposium on Semantic Mining in Biomedicine (SMBM '05)*, pp. 51–59, Hinxton, Cambridgeshire, UK, April 2005.
62. Eunju Kim, Yu Song, Gary Geunbae Lee, Byoung-Kee Yi., "Learning for interaction extraction and verification from biological full articles," Proceedings of the ACM SIGIR 2004 workshop on search and discovery in bioinformatics, July 2004.
63. Wong,L., "PIES, a protein interaction extraction system," *Pac.Symp. Biocomput.*, 520–531 ,2001.
64. Daraselia N., Yuryev A., Egorov S., Novichkova S., Nikitin A., Mazo I. , "Extracting human protein interactions from MEDLINE using a full-sentence parser," Bioinformatics, Vol. 20, Number 5 , pp. 604-611(8) , 22 March 2004.
65. Novichkova, S., S. Egorov, et al., "MedScan, a natural language processing engine for MEDLINE abstracts," Bio-informatics 19(13): 1699-1706, 2003.
66. Donaldson I, Martin J, De Bruijn B, Wolting C, Lay V, Tuekam B, Zhang S, Baskin B, Bader GD, Michalickova K, Pawson T, Hogue CW, "PreBIND and Textomy - mining the biomedical literature for protein-protein interactions using a support vector machine," BMC Bioinformatics, Mar 27,2003.
67. David P. A. Corney, Bernard F. Buxton,William B. Langdon and David T. Jones, "BioRAT: extracting biological information from full-length papers," Received December 19, 2003; revised on June 4, 2004; accepted on June 25, 2004 Advance Access publication July 1, 2004.
68. Cunningham, H., Maynard, D., Bontcheva, K. & Tablan, V., "GATE: A framework and graphical development environment for robust NLP tools

and applications," in ‘Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL’02)’, Philadelphia, USA, 2002.

69. Leroy, G. Hsinchun Chen Martinez, J.D. Eggers, S. Falsey, R.R. Kislin, K.L. Zan Huang Jiexun Li Jie Xu McDonald, D.M. Gavin Ng ,"Genescene: biomedical text and data mining," The University of Arizona, Proceedings. 2003 Joint Conference on, on page(s): 116- 118, 27-31 May 2003.

70. A. Clegg, and A. Shepherd, “Benchmarking Natural-Language Parsers for biological Applications using dependency Graphs,” *J. BMC Bioinformatics,* vol.8- pp. 24, Jan 2007.

71. J. Ding, D. Berleant, J. Xu, and A.W. Fulmer, “Extracting Biochemical Interactions from MEDLINE Using a Link Grammar Parser,” *Proc. 15th IEEE Inter. Conf. Tools with Artificial Intelligence (ICTAI’03),* pp. 467-471, 2003.

72. Y.C. Lin, C.L. Peng, C.Y. Kao, H.F. Juan,H. C. Huang, "ProtExt: A system for protein-protein interactionextraction from PubMed abstracts" , *Proc. 12th Inter. Conf. Intelligent Systems for Molecular Biology (ISMB) and Conf. Computational Biology (ECCB),* 2005.

73. S.T. Ahmed, D. Chidambaram, H. Davulcu, and C. Baral, “IntEx: A Syntactic Role Driven Protein-Protein Interaction Extractor for Bio-Medical Text,” *Proc. ACL-ISMB workshop linking biological literature, ontologies and databases: Mining biological semantics,* pp. 54-61, 2005.

74. Z. Yang, H. Lin, and B. Wu, "BioPPIExtractor: A Protein–Protein Interaction Extraction System for PubMed Abstracts," *J. Expert Systems with Applications,* Article in press, doi: 10.1016 /j.eswa.2007.12.014. 23 Dec. 2007.

75. Pustejovsky, J., *et al*, "Robust relational parsing over biomedical literature: Extracting inhibit relations," *Proc. Pacific Symposium on Biocomputing (PSB),* 362–373, 2002.

76. Humphreys, K., *et al.*, "Two applications of information extraction to biological science journal articles: Enzyme interactions and protein structures," *Proc. Pacific Symposium on Biocomputing (PSB),* 502–513. 2000.

77. Rindflesch, T.C., *et* al., "Edgar: Extraction of drugs, genes and relations from the biomedical literature," *Proc. Pacific Symposium on Biocomputing (PSB),* 514–525. 2000.

78. Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella., "Kernel methods for information extraction," *Journal of Machine Learning Research,* 3:1083–1106, 2003.

79. Aron Culotta and Jeffrey Sorensen., "Dependency tree kernels for relation extraction,"In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004)*, Barcelona, Spain, 2004.

80. Dennis Grinberg, John Lafferty, and Daniel Sleator. A robust parsing algorithm for LINK grammars. Technical Report CMU-CS-TR-95-125, Pittsburgh, PA, 1995.

81. D. Temperley, D. Sleator, and J. Lafferty, "Link Grammar," *Carnegie Mellon University,* http://www.link.cs.cmu.Edu/link. 1998.

82. D. Sleator, and D. Temperley, “Parsing English with a Link Grammar,” *Third International Workshop on Parsing Technologies,* pp. 277-292, 1993.

83. D. Temperley, D. Sleator, and J. Lafferty, "Abiword- word processor for everyone," *Carnegie Mellon University,* http://www.abisource.com. 1998.

84. S. Pyysalo, T. Salakoski, S. Aubin and A. Nazarenko, “Lexical Adaptation of Link Grammar to the Biomedical Sublanguage: a Comparative Evaluation of Three Approaches,” *J. BMC Bioinformatics,* vol. *7,* pp. 60-67, Novembe*r* 2006.

85. E. Turner, "The LinkGrammar-WN," http://www.eturner.net/ linkgrammar-wn.2007

86. "WordNet-a lexical database for the English language," *Princeton University*, http://wordnet.princeton.edu. 2006

87. P. Szolovits, "Adding a Medical Lexicon to an English Parser," *Proc. AMIA 2003 Annual Symposium.* pp. 639-643 ,2003

88. "UMLS'-Unified Medical Language System," *U.S. National Library of Medicine*, http://umlsinfo.nlm.nih.gov.1999.

89. S. Pyysalo, F. Ginter, T. Pahikkala, J. Boberg, J. J¨arvinen, and T. Salakoski, “Evaluation of Two Dependency Parsers on Biomedical Corpus Targeted at Protein—Protein interactions,” *J. Inter. Medical Informatics*, Vol. 75, Issue 6, pp. 430-442, June 2005.

90. D. Brian “Lingua::LinkParser- Perl module implementing the Link Grammar Parser,” *Carnegie Mellon University*, http://search.cpan.org/~dbrian/Lingua- LinkParser 1.08. 2004.

91. "CPAN - Comprehensive Perl Archive Network," http:// www.cpan.org. 1995

92. D. Temperley, D. Sleator, and J. Lafferty, "The parser Application Program Interface (API)," *Carnegie Mellon University*, http://www.abisource.com/projects/link-grammar/api/index.html. 1998.

93. V. Harsha, Madhyastha, N. Balakrishnan, K.R. Ramakrishnan “Event Information Extraction Using Link Grammar,” *Inter. Workshop Research Issues in Data Eng.: Multi-lingual Information Management (RIDE'03),* pp. 16- 22, 2003.

94. L. Hirschman, J.C. Park, J. Tsujii, L. Wong, and C.H. Wu5, "Accomplishments and Challenges in Literature Data Mining for Biology." *J. Bioinformatics*, vol. 18, pp. 1553-1561, June 2002.

منـهجيّات إستخلاص تفاعلات البروتينـات من الملخصات الطبيه بإستخدام محلل قواعد الربط

إعداد

رانيا احمد عبد العظيم عبد الرحمن أبو السعود

رسالة مقدمه إلي كليه الهندسة، جامعه القاهرة

كجزء من متطلبات الحصول علي درجه الدكتوراه

في

الهندسه الحيوية الطبية والمنظومات

كلية الهندسه، جامعه القاهرة

الجيزه- جمهورية مصر العربيه

يوليو ٢٠٠٨

منهجيّات إستخلاص تفاعلات البروتينات من الملخصات الطبيه بإستخدام محلل قواعد الربط

إعداد

رانيا احمد عبد العظيم عبد الرحمن أبو السعود

رسالة مقدمه إلي كليه الهندسة، جامعه القاهرة
كجزء من متطلبات الحصول علي درجه الدكتوراه

في
الهندسه الحيوية الطبية والمنظومات

تحت إشراف

أ.د. أبو بكر محمد يوسف
قسم الهندسة الطبيه والنظم
كليه الهندسة

أ.م.د. ياسر مصطفي قدح
قسم الهندسه الطبيه والنظم
كليه الهندسة

جامعه القاهرة

أ.م.د. ناهد حسن سلومه

معهد الليزر

جامعه القاهرة

كلية الهندسه، جامعه القاهرة

الجيزه- جمهورية مصر العربيه

يوليو ٢٠٠٨

منـهجيّات إستخلاص تفاعلات البروتينـات من الملخصات الطبيه بإستخدام محلل قواعد الربط

إعداد

رانيا احمد عبد العظيم عبد الرحمن أبو السعود

رسالة مقدمه إلي كليه الهندسة، جامعه القاهرة
كجزء من متطلبات الحصول علي درجه الدكتوراه

في
الهندسة الحيوية الطبية والمنظومات

يعتمد من لجنه الممتحنين:

الأستاذ الدكتور: **أبو بكر محمد يوسف** — المشرف الرئيسي

الأستاذ المساعد الدكتور: **ياسر مصطفي قدح** — مشرف

الأستاذة الدكتورة: **سامية مشالي** — ممتحن

الأستاذ الدكتور: **محمد عماد موسي رسمي** — ممتحن

كلية الهندسه، جامعه القاهرة
الجيزه- جمهورية مصر العربيه
يوليو ٢٠٠٨

ملخص الرسالة

شهد العقد الماضي نموا لم يسبق له مثيل في كل من الطب الحيوي في إنتاج كمية من البيانات والكتابات المنشورة مناقشته.التقدم في الأساليب البيولوجية الحاسوبية وتغير ملحوظ في حجم بحوث الطب الإحيائي. الجينوم الكامل يمكن الآن يتسلسل في غضون أشهر وحتى أسابيع ، الأساليب الحسابية الإسراع في تحديد عشرات الآلاف من الجينات على نطاق واسع وأساليب تجريبية. البيانات المتولدة عن هذه التجارب هي مرتبطة بدرجه عالية ؛ من نتائج تحليل تسلسل والجزئي المصفوفات تعتمد على المعلومات الفنية ونقل الإشارة المذكورة في الممرات لاستعراض الأقران لمنشورات الادله. على الرغم من العلماء في هذا المجال ، يتم مساعدتهم بالعديد من قواعد البيانات على الانترنت من التفاعلات الكيميائية الحيوية ، لكن في الوقت الحاضر اغلبيه هذه هي المنسقة : العمل بشكل مكثف من قبل خبراء المجال. ولذلك فقد السعي لاستخراج المعلومات من النص بنشاط في محاولة لانتزاع المعرفة من المواد المنشورة والإسراع في عملية التدوين في قواعد البيانات إلى حد كبير. استخراج أداة من شأنه ليس فقط توفيرا للوقت والجهد، ولكنها أيضا تمهد الطريق لاكتشاف معلومات غير معروفة حتى الآن ضمنا في النص. هذه الفرضية تعرض النظام الذي يحمل اسم PIELG لاستخراج التفاعلات البروتين في الخلاصات الطبية الحيوية. ونهجنا يقوم على تقسيم الأولى الخلاصات إلى الجمل البسيطة. بعد ذلك ، التعليم البيولوجي الكيانات مع مساعدة الطبية واللغوية PIELG. وأخيرا، والاستخراج وذلك من خلال تحليل التفاعلات الكامل المطابقة محتويات نحوي أدوار ومجموعات كبيرة لغويا. النظام يتناول الأحكام ومعقدة ومتداخلة متعددة مقتطفات التفاعلات المحددة في حكم صادر بحقه. ونحن في نطاق التجارب يقتصر على المستخلصات التي تصف حقوق وظيفة البروتين. فان مجموعة من النظام يتم اختيار لتقييم المقترح من البروتين التفاعل المصادقة طريقه. هذا هو مجموعة مختارة لتكون البروتينات في الوقت الراهن عن أن يكون في الأسنان تشكيل الأدوار والمشاركة في عملية تكوين الأسنان. نفذنا عمليات التقييم التجريبية للنظام PIELG.

التفاعلات المستخرجة من النظام تفحص يدويا للدقة والتذكير. حساسية هذا النظام هو الذي توليه يذكر التدبير، محسوبة على أنها النسبة بين التفاعلات المستخرجة بطريقة صحيحة والتفاعلات في هذا النص. الدقة هي مقياس لصحة النظام من خلال قياس عدد مرات النتائج المستخرجة بشكل صحيح بالمقارنة مع العدد الكلي للنتائج. ونحن تجريبي وتبين النتائج أن النظام المعروضة هنا يحقق أداء أفضل من دون الحاجة إلى إنشاء نمط من الدليل (من قبل المستخدم) التي تلزم لهذه النظم الأخرى. إن مجال تجاربنا محدود إلى الملخصات التي توصف البروتين الإنسانية. إن منظومة PIELG مختارة بأن تقيم طريقة تفاعل التي تعتبر الآن لها أدوارا في عملية تكوين الأسنان. أنجزت تقييم أنظمة PIELG تجريبية. التفاعلات المستخلصة بواسطة نظام PIELG مفحوصة بشكل يدوي. النظام أعطى حساسية حيث استخلصت النسبة بين التفاعلات بشكل صحيح و تقدم التفاعلات في النص. الدقة إن لم تكن النتائج استخلص بشكل صحيح المقارنة مع الرقم العام النتائج. تعرض نتائجنا التجريبية ذلك الذي نظام PIELG مقدم هنا أحرز أداء بشكل أفضل بدون الحاجة من خلق النموذج يدوي (بواسطة المستخدم) أي إن مطلوب لأجل هؤلاء الأنظمة الأخرى.

وتحتوي الرسالة علي عشرة فصول وهي:_

الفصل الأول : يعرض مقدمه عن الاستخلاص الآلي للمعلومات ويشرح مشكله البحث ويقدم ملخص لمحتوي الرسالة.

الفصل الثاني : يعرض الحافز لبناء النظام وتطبيقاته المختلفة .

الفصل الثالث:يعرض الخلفية التاريخية لنظم الاستخراج اللي للمعلومات من النصوص و يعرض الأساسيات المطلوبة للنظم.

الفصل الرابع: يعرض ملخص لبعض الطرق المستخدمة في النظم مع عرض بعض النظم المتواجدة الآن في السوق.

الفصل الخامس: يعرض الهيكل الأساسي لنظام .PIELG

الفصل السادس و الفصل السابع: يعطى هذا الفصل شرحا تفصيليا لكل جزء من أجزاء النظام PIELG.

الفصل الثامن: يعرض النتائج الخاصة ب النظام PIELG مع تقييم لهذه النتائج باستخدام عدة طرق مع عرض أول طريقه للتقويم. و يعرض ثاني طريقه لتقييم النتائج مع شرح وافر لها**.**

الفصل التاسع: الاستنتاجات الكلية من الرسالة.