Extraction of Protein Interaction Information from Unstructured Text Using a Link Grammar Parser

Rania A. Abul Seoud
Department of Computer Engineering, Faculty of Engineering, Fayoum University, Fayoum, Egypt
Abou-Bakr M. Youssef, Yasser M. Kadah
Department of Biomedical Engineering, Faculty of Engineering, Cairo University, Giza, Egypt
E-mail: r-abulseoud@k-space.org

***Abstract*-As research continues to generate vast amounts of data, pertaining to protein interactions, there is a critical need to capture these results in structured formats permitting for computational analysis. Automated the extraction of interactions from unstructured text, would improve the content of databases that store this information and set a method for managing the continued growth of new literature being published. Many algorithms have been reported for extracting biochemical interactions from biomedical text. Natural language processing approaches at various complexity levels have been recorded for extracting biochemical interactions from biomedical text. Some algorithms used simple template matching, others exploit sophisticated parsing techniques. In this paper, we present an automated NLP-based information extraction system, to identify protein interactions in biomedical text. Link grammar parsing can handle many syntactic structures and is computationally relatively efficient. Customizing the parser for the biomedical domain is expected to improve its performance further. Our approach is based on first, tagging biological entities with the help of biomedical and linguistic protein names databases. The system extracts complete interactions by analyzing the matching contents of syntactic roles and their linguistically significant combinations.**

Keywords - Bioinformatics; Natural Language Processing; Information Extraction; Protein-Protein Interactions; Link Grammar.

I. Introduction

Genomic research in the last decade has resulted in the production of a large amount of Information about protein function. That generated data is highly connected; hence, should be such data is made easily available. In addition, scientists in the field are aided by many online databases covering different aspects of protein function, such as protein–protein interaction DIP[1] and BIND[2], CSNDB[3] and SPAD[4]. 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. There is an urgent need for an automatic system capable of accurate extracting protein function information from literature. Many approaches have been proposed for information extraction (IE) from scientific publications, ranging from simple statistical methods to advanced natural language processing (NLP) systems. The first step done towards Information extraction was to recognize the names of proteins, genes, drugs and other molecules [1]. The next step was to recognize interaction events between such entities [2]. Basic information extraction approaches rely on the matching of pre-specified templates (patterns) or rules. A number of groups reported application of pattern-matching-based systems for protein-function information extraction [2], [3], [4]. The shortcoming of such systems is their inability to process correctly anything other than short, straightforward statements, which are quite rare in information-saturated MEDLIN[5] and PubMED[6] abstracts.

In the last few years, natural language processing (NLP) has become a rapidly-expanding field within bioinformatics, as the literature keeps growing exponentially [5] beyond the ability of human researchers to keep track of, at least without computer assistance. 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.

More advanced systems utilizing shallow parsing techniques have been described to extract protein interactions [6]. Shallow parsers perform partial decomposition of a sentence structure. Unlike word-based pattern matchers, shallow parsers [7] 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. In some cases, shallow-parsers are used in combination with various heuristic and statistical methods [8]. 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. However, full parsers are significantly slower and require more memory. A problem of parsing ambiguity can be reduced by employment of domain-specific *context-sensitive grammars*. This approach has been implemented in a system called MedLee[7]. Another system is called GENIES [9] which utilizes a grammar based NLP engine for information extraction. *Context-free parsing systems*, on the other hand, are general enough to be applicable to any domain, but completely generic systems seem to be impractical and inefficient. The Pathway Assist system uses an NLP system, MedScan[8], for the bio-medical domain that tags the entities in text and produces a semantic tree. Slot filler type rules are engineered based on the semantic tree representation to extract relationships from text. Recently, it has been extended as GeneWays[9], which also provides a Web interface that allows users to search and submit papers of

[1] http://dip.doe-mbi.ucla.edu/
[2] http://www.bind.ca/
[3] http://geo.nihs.go.jp/csndb/
[4] http://www.grt.kyushu-u.ac.jp/eny-doc/
[5] http://medline.cos.com/
[6] http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=PubMed
[7] http://lucid.cpmc.columbia.edu/medlee/
[8] http://www.ariadnegenomics.com/products/medscan/
[9] http://geneways.genomecenter.columbia.edu/

1-4244-1366-4/07/$25.00 ©2007 IEEE

interest for analysis. The BioRAT[10] system uses manually engineered templates that combine lexical and semantic information to identify protein interactions. *Grammar engineering approaches*, on the other hand use manually generated specialized grammar rules that perform a deep parse of the sentences. *Machine learning approaches* have also been used to learn extraction rules from user tagged training data [10]. These approaches represent the rules learned in various formats such as decision trees or grammar rules. 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. The IntEx (A Syntactic Role Driven Protein-Protein Interaction Extractor for Bio-Medical Text) the system, [11] has used a dependency based English grammar parser, the LG (Sleator and Temperley 1993), to identify syntactic roles for information extraction. Dependency parsers analyze the sentence as a set of pair wise word-to-word dependencies, each dependency having a type that specifies its grammatical function (e.g. Subject and object) **[12]**

This paper investigates the link grammar parsing for extracting protein - protein interactions. The information extraction system efficiently processes sentences from PubMED abstracts using a dependency based English grammar parser to produce to set of simple sentences with various syntactically links.. The Link Grammar [13] used to identify interactions between proteins. This approach relies on various linkage paths between named entities such as protein names. We focus our research on extracting interactions on the sentence level and base our method on link grammar further extending the idea of Sayed T. et al., 2005 [11].

II. Methodology

A. System Overview

The proposed information extraction system can be split into the following steps Fig. 1,

1) *Information Retrieval (IR):* The user provides an initial search specification, which he/she thinks that it is best represents and characterizes the required protein. Then the information retrieval module starts retrieving all PubMed's abstracts satisfying user's specification.

2) *Sentence segmentation and tokenization:* Splitting the retrieved abstracts into sentences included titles of each paper. Title of 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 done by using a simple heuristic to identify sentence boundaries, assuming any period followed by a space and an uppercase letter is a sentence boundary.

3) *Named entity recognition and conversion:* Each retrieved abstract is scanned to identify sentences that mention interaction of wanted proteins and also marking each protein name in each sentence.

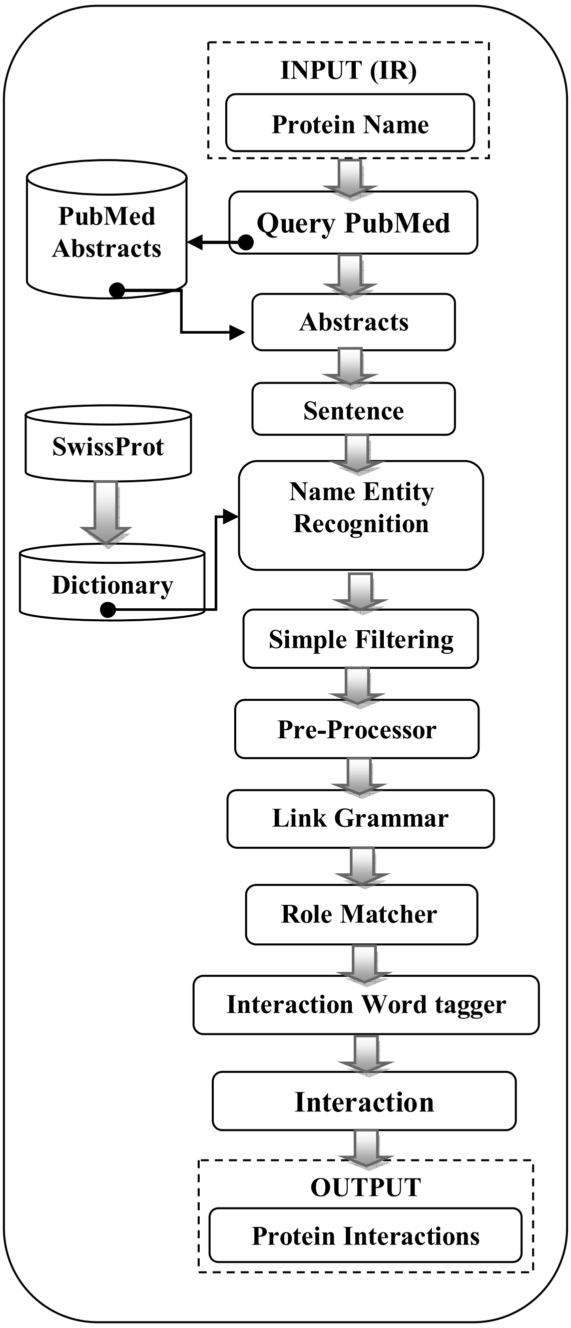


Figure 1. System Architecture.

Each sentence is considered an "evidence" for an interaction. Our recognition method includes both dictionary-based and name-guessing technique. We distill protein names from Swiss-Prot database and build a dictionary which carries about 10000 entries. After recognition, we convert each biochemical name into a personal name. This is necessary because link parser does not have an unbounded dictionary which may hold the vocabulary of all chemical substances. Common personal names are already known to the link grammar parser and doing this can prevent it from guessing the biochemical names. 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.

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

4) *Simple filtering:* Those sentences were again searched for the protein pairs. Sentences without more than one protein name or without any interaction-related verbs are ignored. This reduces the processing time for the abstracts by filtering out sentences that do not contain interactions.

5) *Preprocessor:* The preprocessor removes some constructs that cause the Link Grammar Parser to produce an incorrect output such as parentheses in the sentences. We should Force 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 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", etc., and removes some information that is unrelated to biochemical interactions, such as a window of time: "(1994-2007)", 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.

6) *Link Grammar and Link Grammar Parser:* The proposed information extraction system uses the Link Grammar Parser (LGP) by [14] as the Natural Language Processor to produce a group of a set Link- Grammar representation representing each sentence.
Link grammar is first introduced by Sleator and Temperly to simplify English grammar [13]. The basic idea of link grammar is to connect pairs of words in a sentence with various links. Each word is viewed as a block with connectors coming out. There are various types of connectors, and connectors may point to the right or to the left. A link consists of a left-pointing connector connected with a right-pointing connector of the same type on another word. A valid sentence is one in which all the words are connected in some way (a complete linkage). Rather than examine the basic context of a word within a sentence, the link grammar is based on words within a text form "links" with one another. 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. The Link Grammar is based on a characteristic that if one draws arcs between related words in a sentence [15], the sentence is ungrammatical if arcs cross one another and grammatical if they do not. In Link Grammar, a linkage is a single successful parse of a sentence: a set of links in which none of the connecting arcs crosses. A sample parse of the sentence, "The dog chased a cat." is shown in Fig. 2, [11]. In this example the link between 'dog' and 'chased' is 'S' ('S' links Subject-noun to verbs), the link between 'chased' and 'cat' is 'O' ('O' links verbs to be direct or indirect Objects) and the link between 'the' and 'dog' is 'D' ('D' links determiners to nouns).

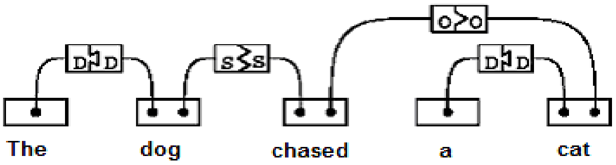
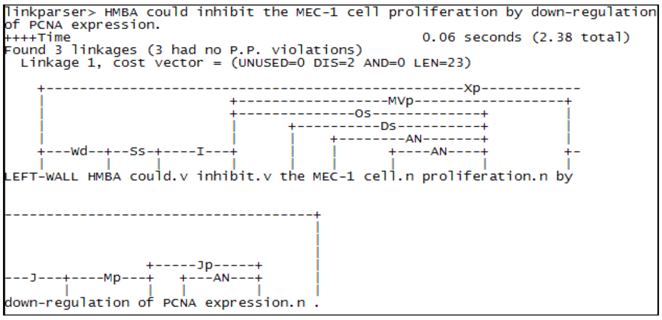


Figure 2. Link Grammar Representation of a sentence "*The dog chased a cat.*" [11].

Davy Temperley, Daniel Sleator and John Lafferty *11* (2005) implemented a parser for link grammar12. It has a dictionary of about 60000 English words. The LGP based on a syntactic dependency grammar of the English language producing links between the words in a sentence that correspond to the syntactic structure of the sentence via subject, object, determiner etc. The parser can recognize a wide range of English syntactic phenomena: noun-verb agreement, questions, imperatives, complex and irregular verbs, many types of nouns, past- or present-participles in noun phrases, commas, a variety of adjective types, prepositions, adverbs, relative clauses, possessives, coordinating conjunctions, and others [11].

All previous mentioned steps of our system are purely implemented with Perl. For the Link Grammar Parser, we use the Lingua::LinkParser Perl module1.0813 at Carnegie Mellon University, which is available in CPAN14. The Lingua::LinkParser provides 107 primary types of links (indicated by the uppercase letters); with many, additional subtypes further detailing the relationship of words (showed by the lowercase characters). The parser also uses a dictionary that contains the linking requirements of each word and the possible part of speech assignments for the entries. The LG parsers' dictionary can also be easily enhanced to produce better parses for biomedical text [16]. We also put a WordNet module (LinkGrammar-WN15) to the dictionary for a larger size of vocabulary. WordNet16 is an online lexical reference system that in recent years has become a popular tool for AI researchers. We also used the extended Link Grammar Parser17 where they extended the lexicon by the lexicon from UMLS'18 (Specialist lexicon enabled to general-purpose language processing tools). The extension of Link Grammar's dictionary [12] effects on its performance. The extension introduces more than 125,000 new words into the LG dictionary, more than tripling its size. This extension can significantly improve efficiency, parsing performance and significantly reduced ambiguity. The extended parser manipulates biomedical text well. In this extended parser they augmented the typically non-technical vocabularies of the ordinary LGP with a large medical lexicon. A sample parse output of the LG parser in the Bioinformatics domains for the sentences "HMBA could inhibit the MEC-1 cell proliferation by down-regulation of PCNA expression." Shown in Fig. 3,

[11] http://www.link.cs.cmu.edu/link/index.html
[12] http://www.link.cs.cmu.edu/link/submit-sentence-4.html
[13] http://search.cpan.org/~dbrian/Lingua- LinkParser1.08/
[14] http://search.cpan.org/
[15] http://www.eturner.net/linkgrammar-wn/
[16] http://wordnet.princeton.edu/
[17] http://groups.csail.mit.edu/medg/projects/text/lexicon.html
[18] http://umlsinfo.nlm.nih.gov/



```
linkparser> HMBA could inhibit the MEC-1 cell proliferation by down-regulation
of PCNA expression.
++++Time                                          0.06 seconds (2.38 total)
Found 3 linkages (3 had no P.P. violations)
  Linkage 1, cost vector = (UNUSED=0 DIS=2 AND=0 LEN=23)

    +---------------------------------------------------Xp------------
    |                      +-------------------MVp-----------------+
    |                      +--------------Os-------------+         |
    |                      |      +-----------Ds---------+         |
    |                      |      |    +---------AN------+         |
    +---Wd--+--Ss-+----I---+      |    |      +----AN----+         +-
    |       |     |        |      |    |      |          |         |
LEFT-WALL HMBA could.v inhibit.v the MEC-1 cell.n proliferation.n by


-------------------------------------+
                                     |
                                     |
                                     |
                 +-----Jp-----+      |
---J---+----Mp---+   +---AN---+      |
       |         |   |        |      |
down-regulation of PCNA expression.n .
```

Figure 3. Example: Link Grammar parses of biomedical sentence.

7) *Role Types and Role Type Matcher:*

Given the syntactic constituents for each sentence we identify the roles based on the data they contain. For example in a sentence ‘‘HMBA could inhibit the MEC-1 cell proliferation by down-regulation of PCNA expression." subject “HMBA” contains one protein name, Object “the MEC-1 cell proliferation” contains one protein name, and modifying phrase “by down-regulation of PCNA expression” contains one interaction word and one protein name. For each syntactic role of the sentence, the role type matcher identifies the type of each role as either ‘‘elementary", ‘‘partial" or ‘‘complete" based on its matching content, (see Table, I). A Syntactic role labeling, done using syntactic parsers like Link Grammar Parser. Semantic role labels are assigned to the constituents of each parse using SVM classifiers.

8) *Interaction Word Tagger*

The words that convey a biologically significant action between two gene/protein names are labeled as “interaction words”. For example in a sentence ‘‘HMBA could inhibit the MEC-1 cell proliferation by down-regulation of PCNA expression.", the main verb “inhibit”, describes the action performed by “HMBA” on “MEC-1”, is an example of interaction word. Some other example of interaction words are “bind”, “down-regulation”, “phosphrylation”, etc. We use a category/keyword dictionary for identifying terms describing interactions. The category/keyword dictionary was adapted from Friedman et al. [9] with additional categories and keywords found to be prevalent in our corpus. The system at this level won't deal with the preposition phrases. The proposed system didn't deal with some of the interactions which differ only in the directionality (e.g., regulated by and inhibited by, etc.).

TABLE I
ROLE TYPE MATCHER

Role Type	Description
Elementary	If the role contains a Protein name or an interaction word.
Partial	If the role has a Protein name and an interaction word.
Complete	If the role has at least two Protein names and an interaction word.

We use dictionary look-up method to identify the interaction words in the sentences. Most of the times these words are in different morphological forms like for the word “regulate” , it can be in some of these morphological forms “regulates”, “regulated”, or even as a noun “regulation”. Porter Stemmer [Por97] was used for stemming from such words. Interaction word tagger first tokenizes the words, and then stems from them before doing the dictionary lookup. The words stored in dictionary are stemmed too.

9) *Interaction Extractor (IE)*

The main component of this module is a set of rules, which can be applied to first identify all the main verbs, i.e., the verbs that truly represent the action in the verb phrase, in the text and then predict the subject for each of these. At the core of our event information extraction scheme is the set of rules to predict the subject and object of a verb as well as modifiers of all verbs and nouns those rules are proposed by [17]. This subject/object 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 and objects.

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 1 i.e., infinitives (e.g. - to, will), modal verbs (e.g. - must, should) and sometimes forms of “be” (like in “he was playing”) 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 specific conditions. After all the main verbs have been identified, the subject (if it exists) for each of them is predicted based on hierarchical fashion with the next rule being applied only if the subject is not found with all the rules before it [17]. The module also helps to find out the object of the verb, when present, as well as the modifiers of all verbs and nouns. Each occurrence of the key verb (interaction word), as a main verb is considered to be one occurrence of the required event. This set of rules predicts the subject and object of a key verb (interaction word) as well as modifiers of all verbs and nouns. 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.

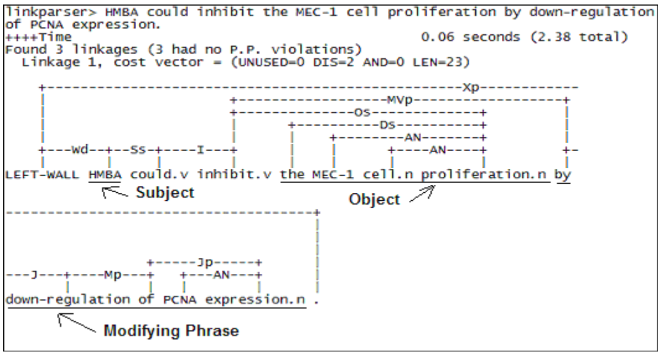
The aim here is to do deep analysis of the sentence to extract multiple and nested interactions from the sentence. The algorithm (Algorithm 1) is based on generic templates constructed using English Grammar syntax, looks into all parts of the sentence. The IE algorithm (Algorithm 1) progresses bottom up, starting with each syntactic role subject, verb or modifying phrases and expanding them uses the lattice until all “Complete” singleton or composite role types are obtained.

Example of Interaction Extractor Algorithm for the following sentence: "HMBA could inhibit the MEC-1 cell proliferation by down-regulation of PCNA expression." is as follow: - The sentences from the biomedical abstracts are parsed using the Link Grammar (LG) Parser. The LG parser gives the output in the form of links between words "Fig. 4," The Algorithm uses the links given by the Link Grammar parser to obtain this syntactic constituents: *Subject (S): "HMBA"*, *Object (O): "the MEC-1 cell proliferation"* and Modifying Phrase (MP): *"by down-regulation of PCNA expression"* and then find the role of each (Table 1).

1. The system identifies the roles based on the information they contain. We will take a sentence as an example. For this sentence the subject "HMBA" contains one protein name, Object "the MEC-1 cell proliferation" contains one protein name, and modifying phrase "by down-regulation of PCNA expression" contains one interaction word and one protein name. For each syntactic role of the sentence, the role type matcher identifies the type of each role as either ‘‘elementary", ‘‘partial" based on its matching content. Here the subject is *Elementary*, object is *Elementary* and modifying phrase is *Partial*. Identify the main verb of the sentence and extract interaction from the combination of Subject and Object roles, when main verb is not an interaction word and when it is an interaction word. We have taken various possible cases in which interaction can occur in a sentence.
2. The main verb ‘‘inhibit" is identified and we try to extract interaction between subject and object. As main verb is an interaction word, we obtain the interaction: (‘‘HMBA", ‘‘inhibit", ‘‘the MEC-1 cell proliferation").
3. Similarly extract interaction from the combination of Subject and modifying roles. We go even further and extract interaction between subject and modifying phrase. Thus we obtain interaction: (‘‘HMBA", ‘‘down-regulation", ‘‘PCNA expression").

III. Preliminary Experimental Results

First, we evaluate our system by selecting pairs of proteins which are known to be interacting with each other from the protein-protein interaction databases. We choose five queries currently considered to have applications in dental medicine "Dentin sialophosphoprotein (DSPP)", "Dentin matrix protein 1 (DMP-1)", "Dentin glycoprotein (DGP)", "Dentin sialoprotein (DSP)", and "Dentin phosphoprotein (DPP)". We look up their interaction properties using an existing protein-protein interaction database like DIP, BIND etc. We send those five queries to PubMed retrieving 1000 abstracts. After manually reviewing all these abstracts, 89 (82%) among them are correct and the recovery rate was found to be 17%. This low recovery rate is primarily due to the low coverage (56%) of the Link Parser module and the imperfection for protein name recognition. The parsing took 18 ms per sentence on a 600MHz Pentium III processor with 128MB of RAM. This means that system is faster than similar systems, and preliminary evaluation indicates that performance can be further increased by a factor of 3–5 using better implementations of programming components such as more efficient memory management.



```
linkparser> HMBA could inhibit the MEC-1 cell proliferation by down-regulation
of PCNA expression.
++++Time                                          0.06 seconds (2.38 total)
Found 3 linkages (3 had no P.P. violations)
  Linkage 1, cost vector = (UNUSED=0 DIS=2 AND=0 LEN=23)

    +---------------------------------------------------Xp------------
    |                      +------------------MVp------------------+
    |                      +--------------Os-------------+         |
    |                      |      +----------Ds----------+         |
    |                      |      |    +--------AN-------+         |
    +---Wd--+--Ss-+----I---+      |    |      +----AN----+         +-
    |       |     |        |      |    |      |          |         |
LEFT-WALL HMBA could.v inhibit.v the MEC-1 cell.n proliferation.n by

-------------------------------------+
                                     |
                                     |
                                     |
                 +-----Jp-----+      |
---J---+----Mp---+   +---AN---+      |
       |         |   |        |      |
down-regulation of PCNA expression.n .
```

Figure 4. The linkage (parse) given by the link grammar parser.

Our system has extracted successfully that Dentin sialophosphoprotein (DSPP) interacts with Dentin matrix protein 1 (DMP-1). Also, it has extracted that Dentin sialophosphoprotein (DSPP) interacts with Dentin phosphoryn. Also, Dentin sialophosphoprotein interacts with the Probable dentin sialophosphoprotein precursors. Dentin sialoprotein interacts with Dentin matrix protein 1 (DMP-1), Dentin phosphoryn protein, and Probable dentin sialophosphoprotein precursor. Dentin sialoprotein interacts with bone sialoprotein. Also we notice that Dentin sialophosphoprotein Contains of two proteins Dentin phosphoprotein (DPP), and Dentin sialoprotein (DSP). The second queries are arbitrary pairs of proteins.

Then, we evaluate our system by determining pairs of unknown proteins. We don'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 the protein-protein interaction databases. Then we start to compare their interaction properties from databases with the obtained interactions to see if there is an interaction or not. Currently the evaluation Algorithm is running, which test the power of the linkage representation generated by the Link Grammar parser to extract protein function information with high precision. The results of this phase will be described elsewhere.

IV. Discussion

In this paper we don't introduce a new parsing technique rather than we investigate the power of the Link Grammar Parser to be utilized in an Information extraction system to extract information about protein-protein interaction. Link grammar parsing can handle many syntactic structures and is computationally relatively efficient. The highly technical terminology and the complex grammatical constructs that are present in the biomedical abstracts make the extraction task difficult. That's why our IE system is based on a deep parse tree structure presented by the Link Grammar. The LG parser's ability to detect multiple verbs and their constituent linkage in a complex sentence makes it better suited for the proposed approach. The quality of parsing has a well-established affect on the performance of IE systems. Given the power of link grammar parser, this method is considered much simpler framework than context-free grammar proposed by [18]. Currently it is not necessarily the case that more powerful grammars lead to better biochemical interaction extraction. Until recently, most Information Extraction (IE) systems for mining semantic relationships from texts of technical sublanguages avoided full parsing [16]. 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 LGP 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. Below is a list of further developments that would enhance the importance of link grammar parsing in the biomedical domain.

1. Extend its dictionary to include technical terms.
2. Extend 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. Develop other algorithms, such as template matching, to further process link paths extracted from the parser's output.
4. Most problems cannot be removed by extending the dictionary and must instead be addressed by modifications of the grammar of the parser.

The scope of the proposed system was limited to sentences describing human protein function. We have also limited our protein name dictionary to the SwissProt entries. DIP contains protein interactions from both abstracts and full text. Since our extraction system was tested only on the abstracts and titles, the system missed out on some interactions that were only present in the full text of the abstract.

V. Conclusion

This paper, presents an information extraction system based on NLP for the purpose of analysis biomedical literature. The link grammar parser is a robust system, which handles almost all aspects of English grammar. Although it is a dictionary-based system, it can handle sentences admirably well even if they have two words or more, which are not in the dictionary and predict the pan-of-speech for these words with a fair degree of accuracy. Also, we have shown that a syntactic role-based approach compounded with linguistically sound interpretation rules applied on the full sentence's parse can achieve better performance than the existed systems which are based on manually engineered patterns which are both costly to develop and are not as scalable as the automated mechanisms presented in this paper. It is concluded that its performance is satisfactory for the real-time PubMed processing.

References

[1] Fukuda K., T. Tsunoda, et al., "Toward information extraction: Identifying protein names from biological papers," PSB, pp. 705-716, 1998.

[2] Blaschkke, C., M. A. Andrade, et al., "Automatic extraction of biological information from scientific text: Protein-protein interactions," *Proceedings of International Symposium on Molecular Biology*, pp. 60-67, 1999.

[3] 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*, vol.9, pp. 62–71, 1998.

[4] See-Kiong, N. and Wong,M., "Toward routine automatic pathway discovery from on-line scientific text abstracts," *Genome Informatics*, **10**, pp. 104–112, 1999.

[5] Andrew Clegg and Adrian Shepherd, "Benchmarking natural-language parsers for biological applications using dependency graphs," *BMC Bioinformatics,* vol.8- pp. 24, Jan 2007.

[6] Thomas J., Milward D., Ouzounis C.A., Pulman S., and Caroll M, "Automatic extraction of protein interactions from scientific abstracts", Pacific Symposium on Biocomputing, pp. 541-552, 2000.

[7] Gondy L., Hsinchun C. and Jesse D., "A shallow parser based on closed-class words to capture relations in biomedical text," *Journal of Biomedical Informatics,* vol.36, pp. 145-158, August 2004.

[8] Claudio G., Alberto L. and Lorenza Romano, "Exploiting Shallow Linguistic Information for Relation Extraction from Biomedical Literature," Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2006), 2006.

[9] C. Friedman, P. Kra, H. Yu, M. Krauthammer, A. Rzhetsky, "GENIES: a natural-language processing system for the extraction of molecular pathways from journal articles," *Bioinformatics*, vol. 17, pp. 74-82(9), June 2001.

[10] Juan Xiao, Jian Su, GuoDong Zhou and ChewLim Tan, "Protein-Protein Interaction Extraction: A Supervised Learning Approach," *Institute for Infocomm Research, Singapore 2School of Computing, National University of Singapore,* 2004.

[11] Sayed T. Ahmed, D. Chidambaram, H. Davulcu, C. Baral, "IntEx: A Syntactic Role Driven Protein-Protein Interaction Extractor for Bio-Medical Text," 2005. in BioLINK SIG: Linking Literature, Information and Knowledge for Biology, a Joint Meeting of The ISMB BioLINK Special Interest Group on Text Data Mining and The ACL Workshop on Linking Biological Literature, Ontologies and Databases: Mining Biological Semantics (Biolink'2005). Detroit, Michigan, pp. 54-61, 2005.

[12] S. Pyysalo, F. Ginter, T. Pahikkala, J. Boberg, J. J¨arvinen, T. Salakoski, "Evaluation of two dependency parsers on biomedical corpus targeted at protein—protein interactions,"*International Journal of Medical Informatics*, Vol. 75, Issue 6, pp. 430-442, June 2005.

[13] Sleator, D. and D. Temperley, "Parsing English with a Link Grammar," Third International Workshop on Parsing Technologies, 1993.

[14] D. Grinberg, J. Lafferty, and D. Sleator, "A robust parsing algorithm for link grammars," in Proceedings of the Fourth International Workshop on Parsing Technologies, Also issued as CMU technical report CMU-CS-95-125

[15] J. Ding, D. Berleant, Jun Xu, Andy W. Fulmer, "Extracting Biochemical Interactions from MEDLINE Using a Link Grammar Parser," Proceedings of the 15th IEEE International Conference on Tools with Artificial Intelligence, pp. 467- 471, 2003.

[16] S. Pyysalo, T. Salakoski, S. Aubin and A. Nazarenko, "Lexical adaptation of link grammar to the biomedical sublanguage: a comparative evaluation of three approaches," Second International Symposium on Semantic Mining in Biomedicine (SMBM) Jena, *BMC Bioinformatics*, vol. 7, pp. 60-67 November 2006.

[17] Harsha V. Madhyastha, N. Balakrishnan, K. R. Ramakrishnan "Event Information Extraction Using Link Grammar," International Workshop on Research Issues in Data Engineering: Multi-lingual Information Management (RIDE'03), pp. 16- 22 , 2003.

[18] J.M. Temkin and M.R. Gilder, "Extraction of Protein Interaction Information from Unstructured Text Using a Context-Free Grammar," *Bioinformatics*, Vol. 19, pp. 2046-2053, April 2003.