An Evaluation of Unnamed Relations Computation for Discovery of Protein-Protein Interactions

James W. Cooper
IBM Thomas J Watson Research Center
PO Box 704
Yorktown Heights, NY 10598
914-784-7285
jwcnmr@watson.ibm.com

ABSTRACT

Many researchers have attempted to find relations in the Biomedical domain using strategies for recognizing protein and gene names, for example. By contrast, our strategy is to combine statistical and lexical techniques to find major noun and verb phrases of all types and compute relations by recurring proximity. We then can apply biomedical term recognition as a filter against the relations we discover. We report here on preliminary work in discovering protein interactions using a standard collection of yeast protein abstracts. Our initial findings are that statistical term relations yield high recall but poor precision when compared to actual tabulated protein interactions.

We also examined these relations using our graphical display of the computed relations. In this case it also helps us discover additional relations indirectly and indicates a fruitful avenue for further inquiry.

Categories and Subject Descriptors

H.3.1 – Content analysis and indexing. *Abstracting methods, Linguistic processing*. H.3.3 -- Information search and retrieval. *Information filtering*.

General Terms

Algorithms, Design, Experimentation, Human Factors, Measurement , Theory, Verification.

Keywords

Text mining, Databases, XML, Java, Protein interactions, Term relations, Lexical navigation, Unnamed relations, Yeast proteins.

1 INTRODUCTION

We have previously described detecting term relations [1] and the layout algorithms for the representation of a lexical network [2],[3], [4]. In this paper, we discuss the algorithms we have used to combine statistical and lexical approaches to finding protein interactions in Medline abstracts.

A number of workers have detected relations between terms. For example, Roark and Charniak[5] have analyze noun phrase co-occurrence statistics by choosing seed words and finding words near them to choose need seed words. This is essentially similar to the Dual Iterative Pattern Relation Expansion (DIPRE) bootstrapping technique originally described by Brin[6]. Agichtein and Gravano [7] generated relations in a manner similar to DIPRE, but used a tagger to add more grammatical intelligence to the process.

In the Biomedical domain, Blaschke *et. al*.[8] identified protein-protein interactions using a small dictionary of common verbs, and Pustejovsky, Castano and Zhang [9] described methods for detecting the *inhibit* relation in a small number of abstracts. Stephens *et. al* .[12] detected a limited set of gene relations from Medline abstracts using small hand-built dictionaries of genes, and relation verbs. They illustrated some of these relations with graphical diagram, but did not describe how it was generated.

Enright and Ouzounis [13] described BioLayout, a graphical system for displaying similarities between proteins, Spencer and Bennett[14] described ProtInAct, an interactive system for displaying interactions between a number of proteins, using the yFiles graph drawing package[17], and Zhang *et.al* [19] described an interactive 3D visualization system for protein interaction mapping. Jenssen *et. al.*[20] constructed a network of genes co-cited in the same abstract, but without any semantic relationship.

Recently, Leroy reported on Genescene, which uses simple parsing to detect relations between genes, and utilized relevant verbs and negation to further characterize the type of relation, [28] and Fu and Mostafa [29] reported on the detection of protein relations using hand-crafted rules and Latent Semantic Indexing.

Ideally we would like to detect relations and construct a relations network that allows knowledge discovery such as that originally found manually by Swanson [21], where he found the relationship between “Raynaud’s disease” and “fish oil.” Some work along this line has also been carried out by Grell[22], and by Ng and Wong [23] where they employed simple pattern matching and some visualization.

Our goal is to use a combination of Natural Language Processing (NLP) and statistical techniques to improve on the state of the art. The above-cited papers give only a few points for comparison. Blaschke’s paper [8] does not cite any precision or recall numbers, Leroy[28] reports a precision of 95% for parser (file)-based relations and 60% for corpus-based (statistical) relations, but no recall figures. Pustejovsky [9] reviewed a number of the other current relation discovery papers, reporting precision recall of 92/21 [33], 73/51 [34], 81/44 [35] and 73/none [36], and themselves reported 90/59 for a limited number of inhibit relations.

1.1 Current Work

In this discussion, we describe how we computed term relations for a set of 523 Medline documents referred to in a table provided by the Munich Information Center for Protein Sequences[30], as discussed further below, and correlated them with protein-protein interactions known to be described in those documents. The system is in no way limited to such small collections, but this collection merely provides a convenient and interesting set of publicly available example documents. We discuss how we used a simple protein dictionary to filter the relations we discover. Finally we illustrate how the relations can be exported and represented in an interactive graphical lexical navigation system for further information discover.

1.2 The JTalent Library

Our system is constructed using our Talent (Text Analysis and Language Engineering Tools) text mining system that recognizes names [24] and multiword technical terms [25] and performs a shallow parse of the document. We use a relational database to store the terms it discovers. The most recent version (Talent 5.1) has been described in detail by Neff [16].

We have constructed the JTalent library and a set of JNI functions that enable us to call functions in Talent from Java. In addition, we have written the KSS library of functions[17] for managing tables in databases such as IBM's DB2 from Java as well. Thus, all of the work we describe here was performed entirely in Java.

We start with this collection of Medline abstracts and run the Talent processor on this collection. This gives us

- A database load file of all the salient terms per document, and their relative token positions in the document.
- A load file of the Medline document metadata: dates, titles, authors and ID numbers.

We can then use a few simple database queries to construct a Terms database table of all the unique terms in the document collection, and compute their frequencies, and the number of documents in which they appear once and more than once. Then we can compute the Information Quotient (IQ) [10] or salience of each term based on these frequencies.

2 COMPUTING RELATIONS

We describe here the Java library code that carries out the computation of relations. The computation is similar to and derived from that described by Byrd and Ravin [11].

We can compute relations between terms in the collection in two ways. First, for each abbreviation whose long form is detected by Talent[15], we compute a "same-as" relation, such as NO for "nitric oxide," and store it in a table as a *named relation.* We can also compute relations between terms based on their proximity. If two terms occur near each other on several occasions within the collection of documents they have a stronger relation than those that co-occur but once. We refer to these as *unnamed relations*, but we regard them as relations for which we have not yet been able to discover a name.

Since we store the document number, and token position for each term in the database, it is a simple matter to find terms that co-occur within any specified distance. Further, we can tune these relations to select only those where one or both of the terms have a salience above a specific value.

We compute the weights of these relations using the mutual information formula

$$m = \log\left(\frac{totalterms \bullet paircount}{freq1 \bullet freq2}\right)$$

where *totalterms* is the total number of unique terms in the collection, *paircount* is the number of documents in which both terms occur, and *freq1* and *freq2* are the frequencies of the two terms in the collection. After computing all the mutual information values *m* for the term pairs, we scale them to lie between 0 and 100.

We can then generate a database load file for the terms and their weights of their unnamed relations. We construct the Relations database table to contain both of the related terms, the strength of the relation and the relation name or "none" for unnamed relations. Named relations are assigned a weight of "100" automatically.

2.1 Use of Dictionaries and Ontologies

In addition to co-occurrence and salience measures, it is particularly useful to relate discovered terms to those in a known dictionary or ontology of terms in a particular domain. Our group has developed code that matches terms with those in the MeSH[27] ontology and assigns MeSH IDs and taxonomy tree locations to each recognized term.

This dictionary matching need not be limited to a single source, however, and it is not unreasonable to search several such dictionaries for term matches. Then, we can further filter the relations we discover by whether one or both of them belong to a particular ontology. In this work we generated a dictionary of protein names and their variants.

3 ANALYSIS OF PROTEIN DOCUMENTS

In this work, we started with a set of documents that are known to contain reports of protein-protein interactions, and evaluate whether relations based on proximity and mutual information can be used to detect these reported interactions. We used the table of yeast (s*accharomyces cerevisiae*) protein interactions prepared by the Munich Information Center for Protein Sequences (MIPS)[30]. This table gives 2604 pairs of protein names and links to the Medline abstract of the document where the relations are reported. It also provides a link to additional information on each protein, including synonyms.

We parsed this web page, creating a table of all the interactions that were reported, and fetched all the abstracts from Medline using a simple Java program. Then, using a pre-annotation program, we marked the protein names in each document and labeled then as a non-existent part of speech so that they would not get combined into larger noun phrases.

We then ran the JTalent system and computed the terms that were nearby each other that were also protein names. Initially, this was not particularly successful because each protein has a number of possible representations that needed to be matched to a common canonical form. For example, the protein SRV2 can also be represented as Srv2p, SRV2p, CAP and (CAP). Synonyms for most of these proteins are available on pages linked from the original page on the MIPS web site. We expanded the dictionary to contain all these synonyms and reran the analysis, storing all terms and their document positions in a TermDoc database table.

Again, we found that the number of relations we could identify was much smaller than the 2604 that the initial MIPS table claimed. However, this table merely indicated that the relations could be found in the *complete article* and not necessarily in the abstract. In order to determine a baseline number of protein names that we could possibly detect in pairs, we constructed a database query to return all of the protein name pairs found in any document. Then we compared this list with all of the pairs extracted from the MIPS site. This query returned 564 different pairs that are also in the MIPS table. Thus, we base our recall numbers on 564 rather than on 2604.

3.1 Computing Relations by Proximity

Once we have all the terms from these abstracts stored in a database table that includes their, paragraph number, sentence number, and offset, we can design queries to ask which proteins occur near each other in the same or adjacent sentences. The results of this computation for spacings of 0, 1, 2, and 3 sentences are shown in Table 1. In this table, precision is the number of matches divided by the total number found. Recall is the fraction of the detected relations which are also listed in the MIPS table and which can be found in the abstracts. Thus recall is matches/569.

Spacing	Matches	No match	Total	Precision	Recall
0	388	432	820	.473	.682
1	494	626	1120	.441	.868
2	531	706	1237	.429	.933
3	548	794	1342	.408	.963
All	569	2360	2929		

Table 1 - Protein interactions found in 0, 1, 2, or 3 sentence spacings.

3.2 Unnamed Relations by Rank

Compared to previous work, the above precision is not that encouraging, although the recall is acceptable. Thus, in the following experiment we evaluated the protein relations within a single sentence based on the computed (mutual information) rank. The ranks are scaled to lie between 0 and 100, with the higher ranks those relations which co-occur more frequently and in more documents than those with lower rank. Intuitively, it would seem that those of higher rank would more likely be correct. However, as shown in Table 2, this does not seem to be the case.

While it first appeared that those relations of lower rank might actually be more accurate, this also did not appear to be the case, as evidenced by the last 3 lines of the table.

Ranks	Match	Nomatch	Precision	recall
0-100	371	1263	.294	.653
51-100	286	1031	.217	.503
56-100	232	882	.208	.408
61-100	154	637	.242	.272
0-75	336	1132	.229	.591
40-75	319	1100	.225	.561
45-75	298	1030	.224	.524

Table 2 - Protein interaction matches by mutual information rank.

We believe that the principle reason for the low precision of these results is that sentences along the lines of

A, b, and c interact with d.

occur throughout the collection, and our proximity algorithm detects relations

a-d, b-d and c-d [1]

as well as

a-b and a-c [2]

where only type [2] relations are correct.

For example we find the following sentence:

> Genetic and biochemical data indicate that Spc98p and Spc97p of the Tub4 complex bind to the N-terminal domain of the SPB component Spc110p.

Correct relations are Spc98p/Spc110p and Spc97p/Spc110p but not Spc97/Spc98. Infurther work we expect to develop methods of eliminated proximity relations between members of a noun phrase list (NPLIST) in most cases.

3.3 Relations using sentence structure

In order to detect only relations of type [2], we went back to the database table that stores noun phrases and verb groups from the JTalent shallow parser and asked for relations between proteins that were separated by a verb. The results are shown in Table 3.

	Match	Nomatch	Precision	Recall
n-v-n	151	458	.248	.265
n-n (with verb anywhere)	382	1317	.225	.671

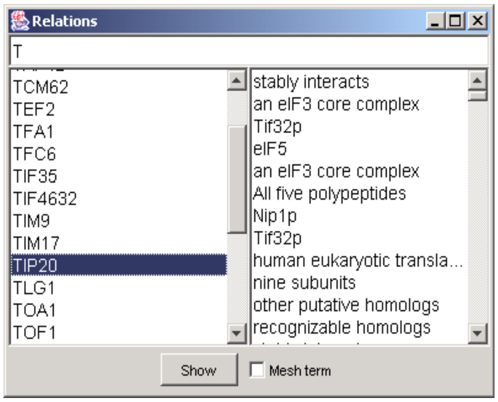
Table 3- Detecting protein interactions in the presence of a verb.

Even though, by inspection, the preponderance of the verbs discovered between two proteins were of biologically interest, such as "describe," "show", "initiates," "are required," and so forth, the precision and recall were extremely poor compared to those in Table 1. We did note, however, that we detected 12 of the 14 verbs used originally by Blaschke. [8] And, considering the small number of documents, it is likely that the other 2 ("acetylate," "is conjugated to") were not mentioned.

4 USING A GRAPHICAL RELATIONS VIEWER

At this point, we determined that inspection of the detected data graphically might give us some clues as to why our precision and recall was so low. Thus we exported the relations into an XML file and used the graphical relations viewer [31] we have described earlier to study these relations.

In this case, we restricted one side of the relations to those which were proteins, but allowed the other term to be from the general collection vocabulary. The system reads an XML file exported from the computed unnamed relations database tables and allows you to explore these relations visually. We illustrate the list box entry point into the system in Figure 1. You enter a part of a term, and the system displays all the terms containing that substring.



Relations

T

TCM62
TEF2
TFA1
TFC6
TIF35
TIF4632
TIM9
TIM17
TIP20
TLG1
TOA1
TOF1

stably interacts
an eIF3 core complex
Tif32p
eIF5
an eIF3 core complex
All five polypeptides
Nip1p
Tif32p
human eukaryotic transla...
nine subunits
other putative homologs
recognizable homologs

Figure 1 – A display of terms related to "TIP20"

When you select one of the terms, it triggers a lookup into the table of Relations stored in a Trie and displays all the relations in the right hand list box. You can select any term and see a graphical display of its relations by clicking on the "Show" button.

The graphical display starts by displaying the single term you selected, and then expands each node when you double-click on it. Nodes which have been expanded turn to a darker blue color.

4.1 Discovery of Secondary Relations

Figure 2 illustrates one such navigation, illustrating relations around TIP20. By inspection we see the relations

TIP20-UFE1

TIP20-SEC20

But if we look in the original MIPS data, we find that there are also interactions between

TIP20-SEC22

SEC20-SEC22

SEC20-UFE1

Each of these can be observed here as a "secondary" relation, one step away from an relation we actually detected. Thus we need to design algorithms to find these relations even though they are one step apart. It is the analysis of these indirect relations which we believe is likely to be the most fruitful way to further improve the accuracy of this combined statistical and linguistic method. We regard this sort of "lexical distance" relation similar to the kind of "semantic distance" discussed in relation to Wordnet. [37]

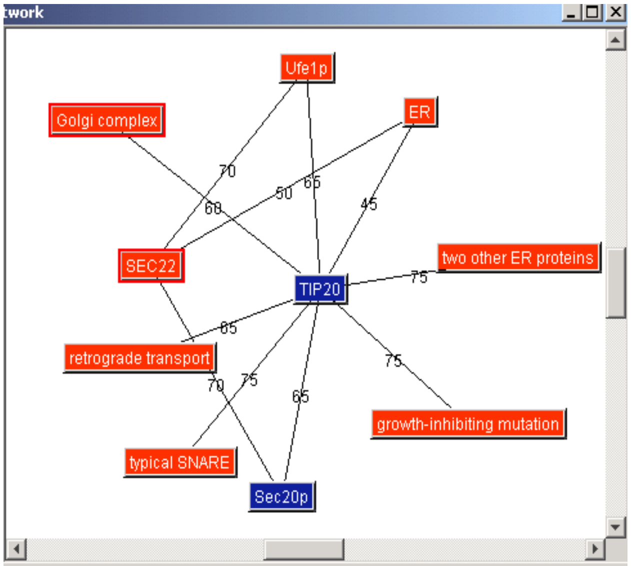


Figure 2 – A Lexical Navigation screen, showing a network of relations around "TIP20."

5 CONCLUSIONS

We have used a combination NLP and statistical means to attempt to predict protein-protein relations on a collection of documents with known relations. The precision is highest (0.47) when the proteins are detected within a single sentence, and the recall highest (0.93) when the window is expanded to 3 sentences. Computation of the strength of these relations based on how often they occur using mutual information methods does not correlate particularly well with the actual relations in the documents. Detection of proteins involved in noun phrase-verb-noun phrase structures does not increase precision or recall. Overall, though, we have made more progress than Ingels and Chen [32].

A promising line for further inquiry appears to be secondary relations, detected through an intermediate protein term. In addition, we will be investigating whether NPLIST structures described in section 3.2 contribute to the low precision.

6 ACKNOWLEGEMENTS

We thank Bhavani Iyer for writing the XML extractor from our database representation, Eric Brown for the use of his DictMatcher code for detecting dictionary terms, and Bob Mack for numerous helpful discussions. The original force-based layout algorithm used in Lexical navigation was written by Daniel Tunkelang while a summer student in our lab.

7 REFERENCES

[1] Cooper, J. W. and Byrd, R J, "Lexical Navigation: Visually Prompted Query Refinement," ACM Digital Libraries Conference, Philadelphia, 1997.

[2] Cooper, James W. and Byrd, Roy J., OBIWAN – "A Visual Interface for Prompted Query Refinement," Proceedings of HICSS-31, Kona, Hawaii, 1998.

[3] Cooper, J. W. "The Technology of Lexical Navigation," JCDL-2001.

[4] Tunkelang, D. D., Byrd, R. J., and Cooper, J. W., "Lexical Navigation: Using Incremental Graph Drawing for Query Refinement," Graph Drawing 97.

[5] Roark, B. and Charniak, C., "Noun phrase co-occurrence statistics for semi-automatic lexicon construction." Proceedings of the 36th Annual Meeting of Association for Computational Linguistics, 1998.

[6] Brin, S "Extracting Patterns and Relations for the World Wide Web," Proceedings of the 6th Annual WebDB Workshop, EBDT98, 1998.

[7] E. Agichtein and L Gravano, "Snowball: extracting Relations from Large Plain-text collections." Proceedings of the 19th IEEE Conference on Data Engineering, 2003.

[8] Blaschke, C., Andrade, M.A., Ouzounis, C. and Valencia, A., "Automatic extraction of biological information from scientific text: protein-protein interactions," *BioInformatics* 4(7), 1998.

[9] Pustejovsky, J, Castano, J. and Zhang, J. "Robust Relational Parsing over Biomedical Literature: Extracting Inhibit Relations," Proceedings of the Pacific Symposium on Biocomputing (PSB) 2002.

[10] Prager, John M., Linguini: Recognition of Language in Digital Documents*, in Proceedings of the 32nd Hawaii International Conference on System Sciences,* Wailea, HI, January, 1999.

[11] Byrd, R.J. and Ravin, Y. Identifying and Extracting Relations in Text. *Proceedings of NLDB 99*, Klagenfurt, Austria.

[12] Stephens, M., Palkal, M., Mukhopadhyay, R and Mostafa, J., "Detecting Gene Relations from Medline Abstracts," Proceedings of the Pacific Symposium on Biocomputing, 2001, Honolulu, HI.

[13] Enright, A.J. and Ouzounis, C. A., "BioLayout –An automatic graph layout algorithm for similarity visualization," *Bioinformatics* 17(9), 853-854 (2001).

[14] Spencer, H. and Bennett, S.P., "Visualizing Protein-Protein Interactions on a Genomic Scale," IEEE Conference on Information Visualization, Boston, 2002.

[15] Park, Y. and Byrd, R. J., "Hybrid text mining for finding abbreviations and their definitions," Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2001.

[16] Neff, Mary, Byrd, Roy J. and Boguraev, B. "The Talent System: TEXTRACT Architecture and Data Model," NAACL Workshop on Software Engineering and Architecture of Language technology Systems, Edmonton, Canada, 2003.

[17] J. W. Cooper, So, Ed, Cesar, C. and Mack, R., "Construction of an OO Framework for Text Mining," OOPSLA 2001.

[18] "Wiese, R., Eiglesperger, M., and Schaber, P. "yFiles Graph Drawing Package, 2002. www.yWorks.com.

[19] Zhang,Y., Tian, H., Kraemer, E. and Arnold, J. "A visualization System for Protein Interaction Mapping Using Java 3D Technology," submitted to *BioInformatics*. 2003. Nissan.cs.uga.edu/ ~yozhang/ protein3D/.

[20] Jenssen, T-K., Komorowski, A-L, Hovig, E., A literature network of human genes for high throughput analysis of gene expression. *Nature Genetics*. 28, 21-28, May 2001.

[21] Swanson, D.R., "Fish oil, Raynaud's syndrome and undiscovered public knowledge," *Perspectives in Biology and Medicine* 30(10 7-18, (1986)

[22] Grell, Stephan, unpublished Master's thesis, University of Heidelberg.

[23] Ng, S-K. and Wong, M., "Toward routine automatic pathway discovery from on-line scientific text abstracts." *Genome Informatics. 10: 104-112*. 1999.

[24] Ravin, Y. and Wacholder, N. 1996, "Extracting Names from Natural-Language Text," IBM Research Report 20338.

[25] Justeson, J. S. and S. Katz "Technical terminology: some linguistic properties and an algorithm for identification in text." *Natural Language Engineering,* 1, 9-27, 1995.

[26] Goodrich, Michael and McGeoch, Catherine, eds., *Lecture Notes in Computer Science* 1619, Springer-Verlag, 1999.

[27] Medical Subject Headings, National Library of Medicine, www.nlm.nih.gov/mesh/meshhome.html.

[28] "Genescene: Biomedical Text and Data Mining," G. Leroy, H. Chen, J. Martinez *et.al.* JCDL 2003, Houston, TX.

[29] Fu, Y., Mostafa, J. and Seki, K. “Protein Association Discovery in Biomedical Literature,” JDCL 2003, Houston, TX.

[30] *Saccharomyces cerevisiae* physical interaction table: *http://mips.gsf.de/proj/yeast/tables/interaction/physical_interact.html,* Munich Information Center for Protein Sequences.

[31] Cooper, J.W. “Visualization of relational text information for biological knowledge discovery,” IVIRA symposium and workshop at JCDL 2003, Houston, TX.

[32] Ingels, J and Chen, T. “Study Finds Jack,” *The Onion* 39, (21), June 4, 2003.

[33] Craven, M. and Kumlien, J., “Constructing Biological Knowledge ases by Extracting Information from Text Sources,” Proceedings of the 7th International Conference on Intelligent Systems in Molecular Biology (ISMB-99).

[34] Rindfleisch, T., Rajan, J., and Hunter, L., “Extracting Molecular Binding Relationships from Biomedical text,” *Proceedings of the ANLP-NAACL 2000.*

[35] Proux, D., Rechenmann, P. and Laurent, J. “A Pragmatic Information Extraction Strategy for gathering Data on Genetic Interactions,” *Proceedings of ISMB-2000.*

[36] Sekimizu, T., Park, H.S., and Tsujii, J. “Identifying the Interaction Between Genes and Gene Products based on Frequently Seen Verbs in Medline Abstracts,” *Proceedings of Genome Informatics*, 62-71, Tokyo, 1998.

[37] Budanitsky, A. and Hirst, G. “Semantic distance in WordNet: An experimental, application-oriented evaluation of five measures,” *Proceedings of NAACL-200*0, Pittsburgh, Pa, June, 2000.