RC23060 (W0401-067) January 13, 2004
Computer Science

IBM Research Report

Discovery of Protein-Protein Interactions Using a Combination of Proximity and Linguistic Information

James W. Cooper
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 704
Yorktown Heights, NY 10598

Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

Discovery of Protein-Protein Interactions Using a Combination of Proximity and Linguistic Information

James W. Cooper
IBM Thomas J Watson Research Center, PO Box 704
Yorktown Heights, NY 10598
914-784-7285 (voice) 914-784-6912(fax)
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 our work in discovering protein interactions using a standard collection of yeast protein abstracts. After adjusting our recognition algorithms to include complexes and resolve apparent false positives, we obtained a precision of 0.92 and a recall of 0.84.

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.

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 analyzed noun phrase co-occurrence statistics by choosing seed words and finding words near them to choose additioal 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[18], 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. This is a broad version of using mutual co-occurrence.

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, [27] and Fu

and Mostafa [28] 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] using mutual co-occurrence, and by Ng and Wong [23] where they employed simple pattern matching and some visualization.

The above-cited papers give only a few points for comparison. Blaschke's paper [8] does not cite any precision or recall numbers, Leroy[27] 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 [31], 73/51 [32], 81/44 [33] and 73/none [34], and themselves reported 90/59 for a limited number of inhibit relations.

1.1 Goals

Our goal is to use a combination of Natural Language Processing (NLP) and statistical techniques to improve on the state of the art. We divide the problems to be solved into the following groups.

1. Recognizing protein names and collapsing all their synonyms into single canonical forms.
2. Recognizing sentence and paragraph structure.
3. Computing the proximity of protein names.
4. Parsing protein-containing sentences into their component parts.
5. Discovering significant biological action verbs and their nominalizations.
6. Recognizing document constructions that are used to relate proteins indirectly.
7. Recognizing noun-phrase lists of pronouns, and using linguistic cues to determine which actually interact.

In this paper, we use the first 5 of these techniques to compute protein-protein interactions found in Medline documents. We consider points 6 and 7 further work.

1.2 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[29], 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.

1.3 The JTafTalent 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 previous version (Talent 5.1) has been described in detail by Neff [16]. The current version is called Taf/Talent [37] and operates in the Unstructured Information Management (UIMA) environment. [36]

We have constructed the JTafTalent library that utilizes the IBM UIMA system to enable us to call functions in Taf/Talent from Java. In addition, we have written a 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 and 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 [10]. We compute relations between terms in the collection in two ways. First, for each abbreviation whose long form is detected by TafTalent[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 a sentence or 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

$$m = \log\left(\frac{totalterms \bullet paircount}{freq1 \bullet freq2}\right) \qquad (1)$$

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.

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 (*saccharomyces cerevisiae*) protein interactions prepared by the Munich Information Center for Protein Sequences (MIPS)[29]. 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.

We then ran the JTafTalent 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 abstract. 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 the document number, 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/564 and precision is matches/retrieved.

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	564	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.

One possible explanation for the false positives is the occurrence of noun phrase lists (NPLIST) such as

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

In the above, proximity calculations would suggest that Spc98p and Spc97p themselves interact, in addition to the clearly stated binding to Spc110p. This may not always be true.

4 DETECTING RELATIONS IN INDIVIDUAL DOCUMENTS

In an effort to improve the accuracy of protein-protein interaction detection, we undertook a detailed study of 65 of the abstracts to determine what algorithms and approaches would be most effective. In this study, we printed out each abstract with a list of the interactions reported by the MIPS table, including all of the synonyms for each protein. Then we read each abstract carefully, marking all the interactions we discovered. In this process we made some interesting discoveries.

1. Some interactions were not reported in the abstracts, but only in the full papers. In fact some review articles contained no protein names at all in the abstracts.
2. Some interactions were described that were not tabulated by MIPS. For example, the abstract might mention prior work.
3. Protein complexes were frequently mentioned: for example the dimer "Ddc2-Mec1" or the trimer "Hap2p-Hap3p-Hap5p." Such complexes are in fact protein interactions and should also be detected and reported.
4. Proteins were frequently referred to by two synonyms separated by a slash, such as "GIM1/YKE2."
5. In all but one case, the interactions were described in the same sentence, and thus resolving co-reference issues would add only marginally to the quality of the interaction detection. Thus, the fact that two proteins occurred distantly in the same abstract, was not a good indicator that an interaction was being reported. Nearly all he reported interactions could be found in a single sentence.
6. Only a few verbs were used to describe protein-protein interactions. In a sample of 65 abstracts, we found *act, activate, associate, bind, complex, co-precipitate, depend, inhibit, interact, mediate, phosphorylate* and *stabilize.*

Accordingly we wrote two additional annotators and an extractor to operate on these abstracts. One annotator recognized protein complexes: dimers and trimers, and the other recognized protein synonyms in the "slash notation" we illustrated in point 4 above. When the annotator found these synonyms, it only annotated one of the two mentions, to avoid skewing the mention statistics. We treated all protein complexes as reports of interactions and annotated as such.

We also wrote an annotation extractor to find the verbs listed above or their noun-equivalents in each sentence, if that sentence contained two or more different protein annotations. These produced reports like [complex AFG3 YTA12] and [activate STE20 CDC42].

4.1 The Detection Algorithm

We describe the algorithm as follows. For each abstract

1. Write a line containing the document id, title, author and date to a document table load file.
2. Find each sentence boundary and annotate each sentence with results of shallow parse.
3. Find each reference to a protein name and annotate with base form of name.
4. Find proteins referred to in slash notation as *prot1*/*prot2*. If they are synonyms, annotate only the first one, otherwise annotate both.
5. Find protein pairs and triples separated by hyphens. If they are different even as variant forms, annotate both proteins as a protein complex.
6. Find the sentences that contain two or more proteins and one of the action verbs or their nominalizations, and annotate those.
7. Extract each protein position (paragraph, sentence, offset) into a database load table.
8. Compute the proteins-protein interactions based on mutual co-occurrence.
9. Compute protein-verb-protein interactions based on the annotations (#6) and store them.

When the documents are done, compute the term frequency, IQ and strength of relations found using mutual information equation 1.

4.2 Evaluation of Revised Annotations

We selected 26 documents randomly from the collection and tabulated

1. The number of relations reported in the MIPS database
2. The number of relations we found by unnamed relation proximity
3. Relations missed by point 2.
4. Relations that could not be found by careful reading of the abstract.
5. Additional relations detected by discovering complexes
6. Relations detected by verbs and proteins.

We found that nearly all of the relations detected by our unnamed relations algorithm actually existed in the document, whether reported by MIPS or not, and that of those our algorithm missed, nearly all were not discussed in the abstract at all.

In these 26 documents, MIPS had reported 129 relations. We found that 17 of these were not in the abstracts. We also found an additional 52 interactions by mutual co-occurrence of which only 6 were incorrect. By reporting complexes as protein interactions as well, we found an additional 37 interactions. Overall, the results showed a precision of 0.92 and a recall of 0.84. This is summarized in Table 3.

Table 3 - Statistics on Protein Interactions in 26 Documents

Total number of MIPs interactions	129
Total number not in the abstract	27
Additional interaction found by proximity	52
Incorrect interactions found by proximity	(6)
Interactions found by annotating complexes	37
Precision	0.92
Recall	0.84
F-number	0.88

While we had anticipated using the protein interaction verbs to filter the excess relations we discovered, we actually found very few cases where spurious relations were discovered.

4.3 Using Classification to Discover Protein-Protein Interactions

While the above results are quite encouraging, we also undertook a study as to whether linear classification methods could improve our accuracy. For this experiment, we extracted 124 sentences or pairs of sentences describing protein interactions from the 523 abstracts and used them as a training set against a linear classifier. This classifier operates on "bag of words" statistics from the sentences. We also extracted 100 sentences or sentence pairs that did *not* contain any statement about protein interaction and used these for training the non-interaction category. The results were disappointing. After training, all of the documents we submitted to the classifier were misclassified 100% of the time with a certainty of 0.135.

In a second experiment, we replaced every instance of any protein with the string "[protein]" and recomputed the training data. All of the trial documents were again misclassified, but with a certainty of only 0.083.

We believe that classification can indeed be a powerful tool for detecting sentences that describe protein interactions, but rather using than a bag of words classifier, we would need to use one where we can in some way specify both a set of features and the order in which they occur. For example, we might supply training data where the subjects, verbs and predicate objects are specifically tagged. This might be a powerful way of detecting co-reference across sentences where it occurs. For example, consider the following:

> We describe the identification of GIM1/YKE2, GIM2/PAC10, GIM3, GIM4 and GIM5*[genes]* in a screen for mutants that are synthetically lethal with tub4-1, encoding a mutated yeast gamma-tubulin. The cytoplasmic *Gim proteins* encoded by these GIM genes are present in common complexes as judged by co-immunoprecipitation and gel filtration experiments. (…) We show that the *Gim proteins* are important for Tub4p function and bind to overproduced Tub4p.

In cases like the above, the phrase "Gim proteins" refers back to a list of 5 different proteins which are present in complexes, later that they bind to Tub4p. Since this is a very specialized domain-specific type of co-reference it is unlikely to be detected by the usual text-mining approaches.

5 USING A GRAPHICAL VIEWER FOR RELATION DISCOVERY

In another experiment, we exported the relations we discovered into an XML file and used a graphical relations viewer [30] 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.

5.1 Discovery of Secondary Relations

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 a darker blue color. Figure 2 illustrates one such navigation, illustrating relations around TIP20. By inspection we see the relations TIP20-UFE1p and TIP20-SEC20p. But if we look in the original MIPS data, we find that there are also interactions between TIP20-SEC22, SEC20p-SEC22 and SEC20p-UFE1 as well.

Each of these can be observed here as a "secondary" relation, one step away from a relation we actually detected. We see how these were derived from the original abstract:

> …In yeast, retrograde transport from the Golgi complex to the ER is mediated by the ER t-SNARE Ufe1p, and also requires two other ER proteins, Sec20p and Tip20p, which bind

each other. Although Sec20p is not a typical SNARE, we show that both it and Tip20p can be co-precipitated with Ufe1p, and that a growth-inhibiting mutation in Ufe1p can be compensated by a mutation in Sec20p. Furthermore, Sec22p, a v-SNARE implicated in forward transport from ER to Golgi, co-precipitates with Ufe1p and Sec20p, and SEC22 acts as an allele-specific multicopy suppressor of a temperature-sensitive ufe1 mutation…

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. [35]

6 CONCLUSIONS

We have used a combination of NLP and statistical means to discover protein-protein relations on a collection of documents with known relations. Specifically, we discovered the sentence boundaries and the sentence parts using NLP techniques, and the mutual co-occurrence using the mutual information computation show in Equation 1.While the existence of a "gold standard" represented by the MIPS yeast protein data was somewhat chimerical, a combination of statistical methods and protein complex detection resulted in the respectable F measure of 0.88. Detection of sentences containing specific verbs involved in interaction and two or more proteins seemed to provide the possibility of a filter against over-generation of relations, but we have so far found few cases where it was needed.

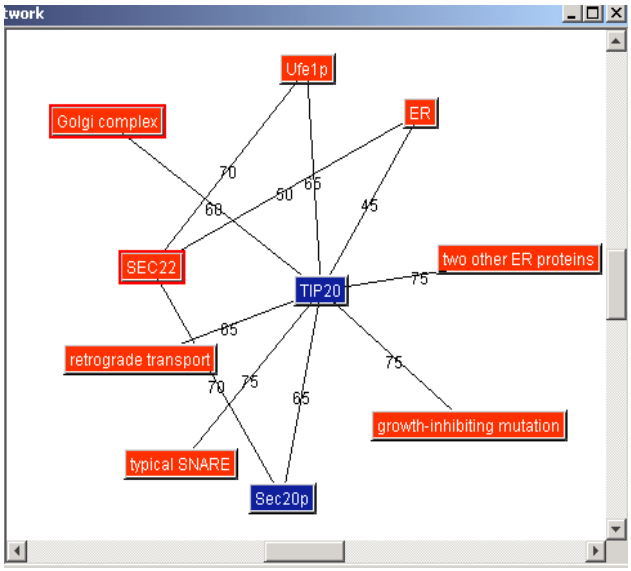

Figure 1 – A Lexical Navigation screen, showing a network of relations around "TIP20."

A promising line for further inquiry appears to be secondary relations, detected through an intermediate protein term. In addition, we plan to investigate the difficult NLP problems of parsing the meaning of NPLIST structures and anaphora resolution in densely written abstract sentences.

7 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.

8 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 n 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, “Information Retrieval in Life Sciences: How to Discover Relations Between Concepts,” unpublished Master’s thesis, University of Heidelberg, December, 2001.
[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] Medical Subject Headings, National Library of Medicine, www.nlm.nih.gov/mesh/meshhome.html.
[27] "Genescene: Biomedical Text and Data Mining," G. Leroy, H. Chen, J. Martinez *et.al.* JCDL 2003, Houston, TX.
[28] Fu, Y., Mostafa, J. and Seki, K. “Protein Association Discovery in Biomedical Literature,” JDCL 2003, Houston, TX.
[29] *Saccharomyces cerevisiae* physical interaction table: *http://mips.gsf.de/proj/yeast/tables/interaction/physical_interact.html,* Munich Information Center for Protein Sequences.
[30] Cooper, J.W. “Visualization of relational text information for biological knowledge discovery,” IVIRA symposium and workshop at JCDL 2003, Houston, TX.
[31] 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).
[32] Rindfleisch, T., Rajan, J., and Hunter, L., “Extracting Molecular Binding Relationships from Biomedical text,” *Proceedings of the ANLP-NAACL 2000*.
[33] Proux, D., Rechenmann, P. and Laurent, J. “A Pragmatic Information Extraction Strategy for gathering Data on Genetic Interactions,” *Proceedings of ISMB-2000*.
[34] 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.
[35] 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.
[36] Ferrucci D., and Lally, A. “The UIMA System Architecture,” *IBM Systems Journal*, March, 2004.
[37] Cooper, J.W. and Neff, Mary S., “Object-Oriented Approaches to developing Document Analysis Systems,” *IBM Systems Journal*, March, 2004.
[38] Mack, Robert L. *et. al.* “BioTeks: Text Analytics for the Life Sciences.” *IBM Systems Journal*, March, 2004.