Robust Relational Parsing over Biomedical Literature: Extracting Inhibit Relations[a]

J. Pustejovsky, J. Castaño, J. Zhang
Department of Computer Science, Brandeis University
415 South St., Waltham, MA 02454, U.S.A.

M. Kotecki, B. Cochran
Department of Physiology, Tufts University, 136 Harrison Ave., Boston, MA, U.S.A.

We describe the design of a robust parser for identifying and extracting biomolecular relations from the biomedical literature. Separate automata over distinct syntactic domains were developed for extraction of nominal-based relational information versus verbal-based relations. This allowed us to optimize the grammars separately for each module, regardless of any specific relation resulting in significantly better performance. A unique feature of this system is the use of text-based anaphora resolution to enhance the results of argument binding in relational extraction. We demonstrate the performance of our system on *inhibition*-relations, and present our initial results measured against an annotated text used as a gold standard for evaluation purposes. The results represent a significant improvement over previously published results on extracting such relations from Medline: Precision was 90 %, Recall 57 %, and Partial Recall 22%. These results demonstrate the effectiveness of a corpus-based linguistic approach to information extraction over Medline.

1 Introduction

A vast amount of new biological information is made available in electronic form on a regular basis. Medline contains over 10 million abstracts, and approximately 40,000 new abstracts are added each month. Although there are growing numbers of sequence databases and other hand-constructed databases, most new information is unstructured text in Medline and full-text journals. This information, which is coming to be referred to as the "biobibliome", is a repository of biomedical knowledge that is larger and faster growing than the human genome sequence itself (Stapley and Benoit[22]). In this age of genomics and proteomics, the ability to process this natural language based information computationally is becoming increasingly important. It is now not uncommon for biologists to study protein complexes and pathways composed of dozens of dynamically interacting proteins. With the recent advent of high sensitivity methods to rapidly identify components of multiprotein complexes (Link

[a] This work was supported by NIH grant R01-LM06649 to Prof. Pustejovsky at Brandeis and Prof. Cochran at Tufts. Direct all correspondences to `jamesp@cs.brandeis.edu`.

et al.[10]), the extent of this complexity is likely to grow exponentially in the next few years. For this reason, the automatic extraction of information from Medline articles and abstracts will play an increasingly critical role in aiding in research and speeding up the discovery process.

To begin addressing this problem computationally, we have begun developing advanced natural language tools for the automated extraction of structured information from biomedical texts as part of a project we call MEDSTRACT (www.medstract.org). Previously we have reported a strategy for the automatic extraction and compilation of biomedical acronyms call *Acromed*. Here we utilize this and other NLP techniques to extract reported relationships between biological entities using the inhibit relation as an example.

The use of computational linguistic techniques for automatically extracting information from biological texts (in particular from Medline) has received increasing attention lately (e.g., Tagaki et al.[23], Sekimizu et al.[21], Hishiki et al.[6], Andrade et al.[1], Blasche et al.[2], Craven et al.[4], Rindfleisch et al. [20], Pustejovsky et al.[17]). Much of the work reported on thus far has focused on specific protein-protein interactions, and in particular, on predicates implicated in binding activities (cf. Sekimizu et al.[21], Blasche et al.[2], and Rindfleisch et al. [20]). Craven et al.[4] use a relational learning algorithm to induce pattern-matching rules on shallow parsed trees for protein-location relations. Although the precision is quite high (92%), their recall is quite low: (21%). The data set they examine is the YPD corpus (Yeast Protein Database). Rindfleisch et al.[20] use shallow parsing combined with UMLS semantic types to extract binding relations from Medline. Their results gave precision of 73% and recall of 51%. Proux et al.[14] also use shallow parsing and domain knowledge (gene type identification) to extract gene-gene interactions from the Flybase corpus. This work is the first we know to pose the problem of retrieving partial x relation information (only one argument of the relation.) Their results were: Precision 81%, Recall 44% and Partial recall (they call it *weak interaction*) 26%. Finally, Sekimizu et al.[21] apply shallow parsing using a general purpose parser (EngCC) to retrieve assertions corresponding to the most frequent set of verbs from Medline abstracts. Their average estimated precision was 73%, for identifying the right subject and object in the relation. No recall is given because no gold standard was created. Partial projected precision for some relations considered in other works mentioned here are: (*interact*: 83.3%, *bind*: 72%, *inhibit*: 83.3%). The results from these experiments are summarized in Table 1 below.

Within information extraction (IE) tasks, entity extraction is typically viewed as a procedure distinct from relation extraction. For example, in enterprise IE systems, products, dates, and company names are easily distinguished from ventures, buy-outs, and product release relations. For most ordinary us-

Lab	Relation	Type Constraints	Data Set	Precision	Recall	P. Recall
Crav.[4]	Location	Protein	YPD	92%	21%	–
Rind.[20]	binding	UMLS	MEDLINE	73%	51%	–
Proux[14]	interact	gene	Flybase	81%	44%	26%
Seik.[21]	several	–	MEDLINE	73%	?	–

Table 1: Previous Relation Extraction Results

age of language, however, and for Medline in particular, the syntax of the sublanguage maps imperfectly to basic semantic distinctions, such as entities and relations. That is, not all entity-looking phrases are entity types; specifically, relations may be expressed as nominalizations (*phosphorylation of GAP by the PDGF receptor*) as well as verbal predications (*X inhibits/phosphorylates Y*). Things become even more complicated for IE when true entities embed relational information by virtue of their semantics, such as the relational entities *the Ron receptor* and *Tissue inhibitors of metalloproteinases*. The difficulty in this example is that such entities are proteins and also incorporate relational semantic information; "x inhibits metalloproteinase". Such considerations demand more sophisticated linguistic processing than is typically employed for IE tasks in enterprise deployments, and certainly richer than the statistical techniques that have received attention in the bioinformatics community recently (cf. Jenssen et al.[8], Marcotte et al.[12]).

2 Design and Methodology

In this paper, we address the problems mentioned above by exploiting a combination of lexical semantic techniques and corpus analytics (Pustejovsky et al[15],[16]). In the section below, we briefly describe this methodology as employed in the Medline domain for targeted information extraction tasks.

Semantic Automata: We begin by constructing simple *semantic automata* from the UMLS database for the relations we are interested in targeting (cf. Humphreys et al.[7]). For example, for *inhibit*-relations and *regulate*-relations, there are four basic selectional patterns (or frames), corresponding to the two options available for each of the two arguments to the relation. These frames are summarized in the table below.

The family of syntactic forms for a lexical item and the mappings to semantic values are part of the typing information encoded within a word, as seeded by UMLS and stored in the lexicon. It should be noted that because the syntactic typing of *inhibit* and *regulate* is transitive, additional semantic automata corresponding to the syntactic passive forms are automatically

ARG-TYPES	Obj = Bio-entity	Obj = Process
Subj = Bio-entity	*(entity,entity)*	*(entity,process)*
Subj = Process	*(process,entity)*	*(process,process)*

Table 2: Selection Patterns

generated. Furthermore, nominal and verbal predicative forms have distinct syntactic distributions and different semantic bindings; hence, they map to different semantic automata, as we see in Section 3.3 below.

Corpus Analytics: We then apply corpus analytics over a subset of Medline corresponding to the target relations, e.g., *inhibit*. Corpus analytics involves several steps (cf. Pustejovsky and Hanks[18]):

i. Create concordances over the predicates (verbal or nominal) associated with the semantic automata;

ii. Automatically cluster complementation patterns of the relation over the concordances, to propose grammar patterns;

iii. Semi-automatically verify and amend grammar rules to ensure correctness and completeness of the patterns for the automata;

For this experiment, we focused on only expressions corresponding to the *inhibit*-relation. The methodology used and the subsequent grammar developed, however, applies to any binary relation with similar semantic typing restrictions. By limiting our case study to this relation set, we were able to create a gold standard corpus with which to evaluate our algorithm. Examples of the concordances used to derive grammar patterns are shown below.

1. `A peptide representing the carboxyl-terminal tail of the met receptor` inhibits `kinase activity`.
2. Whereas `phosphorylation of the IRK by ATP` is inhibited by `the nonhydrolyzable competitor adenylyl-imidodiphosphate`.
3. `The Met tail peptide` inhibits `the closely related Ron receptor` but does not affect ...

A set of 2,000 abstracts was first selected from the Medline database, identified from the concordance constructed around verbal forms and nominal forms of the stem *inhibit*. This set of abstracts was used as a development corpus to optimize each of the components of our system. ¿From this development corpus, a subset of 500 abstracts was used as a core for manual mark-up by domain-expert biologists. In the development-test cycle, a complete month

of Medline abstracts was used for robustness test.[b] We then preprocessed and tagged 500 abstracts of the development corpus , with the following results: There were 497 verbal instances of *inhibit* (*inhibited*, *inhibit* and *inhibits*); There were 342 instances of nominal forms (187 correspond to *inhibition*, and 155 instances to *inhibitor/s*. All the other forms were either gerundives instances (13 of *inhibiting*) or instances of compound forms, e.g., *bisphosphonate-inhibited* (7 instances). Given this distribution and some independent linguistic considerations to be mentioned later, we focused primarily on the development of the grammar of verbal predication.

3 A Description of the system

3.1 General Architecture

As mentioned above, the task of recognizing and extracting *inhibit*-relations between biological entities and processes is part of a much larger research effort underway at Brandeis and Tufts, called MEDSTRACT. The goal of MEDSTRACT is to provide tools and resources to biomedical researchers for better search, retrieval, and navigation of new facts and products within the biological literatures. An illustration of the relevant portion of the architecture is shown below in Figure 1.

3.2 Preprocessing

After identifying the corresponding fields of the Medline documents, titles and abstracts are tokenized. Tokens are then tagged, using a Brill-like rule-based decision procedure. A lexicon with single or multiple tags for each word is used. If the word in question has multiple tags in the lexicon, then it is tested to match a set of disambiguation rules. If it matches any, the corresponding tag is assigned. Otherwise the most probable tag is assigned. The source lexicons used were the lexicon produced by Brill's tagger and the corresponding lexicon for the UMLS Thesaurus (Humphreys et al[7]), with its corresponding syntactic information. The tagged elements were then stemmed with a version of the Porter stemmer. The information corresponding to the *string*, its syntactic tag, and the corresponding stem is stored in a *preterminal* object.

[b] We are currently developing gold standards for other *inhibit*-relations, and testing the robustness of the algorithm on these sets; e.g., *block*, *regulate*, *stimulate*, etc.

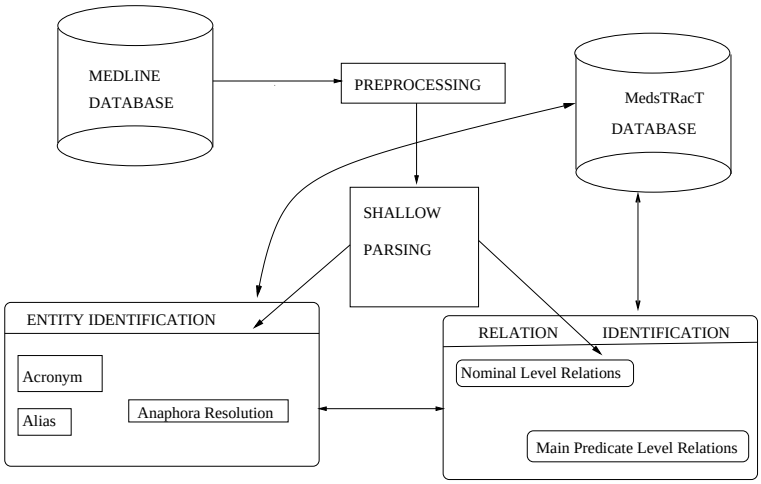

Figure 1: System Architecture

3.3 Type Identification

Our system uses one of two resources for dynamic semantic typing of the input: (a) the UMLS Thesaurus can be exploited to assign types to nouns or noun phrases according to the UMLS type ontology; (b) the GO ontology is also available as a type resource for specific genomic data. For the present experiment, however, neither resource was used, since we were focused primarily on evaluating the construction and deployment of syntactic patterns from semantic automata. Integration of semantic tags into the parsing procedure is under development. Furthermore, we wanted to test the robustness of syntactic techniques independently of typing information. The UMLS types were however used in the anaphora resolution task, as one of the parameters in ranking the possible antecedents list.

3.4 Shallow Parsing Module

The construction of shallow parse trees involves a cascade of five separate automata, each focusing on a distinct family of grammatical constructions. This is very much in the spirit of Hindle[5], McDonald[11] and Pustejovsky et al.[16]. These can be distinguished as follows:

> **Level I**: Noun chunking, groups Proper Nouns and common nouns. It also groups some double prepositions, and compound relational terms.

Level II: Creates noun phrase chunks without prepositional phrases (including adjectives and determiners). It also creates relational terms chunks (verbal chunks, including some adjectival and adverbial terms).

Level III: Creates chunks for coordinated nouns or noun chunks and coordinated verbal chunks or verbs.

Level IV: Creates chunks of noun phrases with *of*-prepositional phrase.

Level V: Identification of subordinate clauses chunks.

3.5 Relation Identification Module

As mentioned briefly above, the concordances derived for *inhibit*-relations distinguished the verbal forms from the nominal forms. Because of their distinct argument binding and complementation behaviors, we decided to develop separate automata for each form, and then merge the results in a subsequent database population phase. In fact, however, there is reason to believe that keeping the results extracted from the two modules separate is actually desirable for database purposes as well; this is due in large part to the degree of relevance associated with 'given' versus 'new' information as presented in documents (cf. Pustejovsky et al,[19]).
The relation identification module was built independent of the specifics of how the verb *inhibit* and associated nominals behave in Medline. Rather, this module was defined and designed to work on the output of the shallow parsing module to identify argument and relational chunks, independently of any specific lexical item. The extraction of a particular relation (e.g. *inhibit* or *regulate*), is accomplished by specifying stems that denote the required relation.

Sentence-level parsing identifies the following constructions:

SENTENCE-LEVEL RELATION IDENTIFICATION

1. Main predicate relational chunk in the sentence.
2. Subject nominal chunk (Nominal chunks at 4th level above)
3. Object nominal chunks.
4. Subordinate clauses (identifying also antecedents of relative clauses, and main predicates of object clauses).
5. Sentential coordination.

It has also the capability of identifying:

1. Preverbal adjuncts.
2. Post Object target adjuncts (ambiguous between adjuncts and nominal modifiers, PP attatchment ambiguity)

In the nominal domain, head nouns may typically carry relational semanctis; for example the noun *inhibitor* can refer to both the relation as well as the biological entity itself, the parsing decisions involved for these forms are distinct from the verbal form. The constructions and relations identified by the nominal-level module are given below:

NOMINAL-LEVEL RELATION IDENTIFICATION

1. Nominal chunks of Level IV.
2. prepositional relational chunks.

Note that relations inside Level IV are decomposed first, i.e., *of*-prepositional relations. Our next step will be to add reduced relative clauses and gerundive relations to this parser module.

3.6 Anaphora Resolution Module

Identifying the arguments of the relations may not be enough for identifying the actual entities involved in the relation. Quite often anaphors (e.g., *it, they*) and sortal anaphoric noun phrases (e.g. *the protein*, *both enzymes*) are the actual arguments to a relation, but unfortunately are not specific enough to establish a unique reference to an entity or process. Although the use of anaphoric terms seems to be relatively infrequent in Medline abstracts, the use of sortal anaphors is quite prevalent. This module focuses on the resolution of biologically relevant sortal terms (i.e., proteins, genes, and bio-processes), as well as pronominal anaphors, including third person pronouns and reflexive pronouns. The initial data source for this resolution algorithm is the pre-processed Medline text (shallow parsed), where each noun phrase (NP) has been identified and annotated with a syntactic tag and semantic tag(s). The anaphora resolution algorithm examines the text sequentially and represents each sentence as a "frame environment". Every NP within a sentence is a potential referent and is made into an entity with a unique ID and syntactic/semantic tags, and added to the sentence environment in which it occurs. If an NP is identified as an anaphor, then the resolution algorithm will attempt to resolve it by traversing through the sentence environments from the most recent (which contains the anaphor), back to the first sentence of the abstract, and selecting the NP among the sentences that has the highest compatibility with the anaphor as the antecedent (cf. Kennedy and Boguraev[9]). The choice of antecedent is determined by matching syntactic and semantic features of the candidate NP with that of the anaphor, which includes person/number agreement, semantic type, as well as physical string comparisons. In the case that more than one NP is found to be equally compatible, preference is given to the one that is most adjacent to the anaphor in the text. If an anaphor requires

multiple antecedents (e.g., the anaphor *both enzymes*) then the resolution algorithm will continue in the sentence environment where the first antecedent is found, and then select the subsequent antecedent which is most compatible with both the anaphor and the first antecedent.

Each resolved anaphor retains its assigned antecedent(s) in memory so as to enable cascading anaphoric links of coreference between an anaphor and a previous discourse referent which could be another anaphor. In addition, special filters are used to exclude the resolution of expletives as well as restricting antecedents of reflexive pronouns to occur in the same sentence as the anaphor.

4 The Evaluation Test

4.1 The mark-up

A new data set of abstracts was collected using a different search, using the strings *protein* and *inhibit*. We ensured that there was no overlap between the training set and the evaluation set. This data set consisted of 56 abstracts, which was manually annotated in XML format, as described at `www.medstract.org`. Those instances which had an argument which referred to an antecedent were annotated as were those corresponding strings for the relations. If the instance in question was particularly difficult to annotate, the comments of the annotator were included. The corresponding entities were annotated with the appropriate semantic type; however, the type information was not used or processed in this experiment. Below is an example of parsed output showing types and bindings of entities in a relation, together with an anaphoric binding

```
<Entity id="83" Type="small molecule"> Cyanide</Entity>,
<Entity id="84" Type="small molecule">azide</Entity>,
<Entity id="85" Type="small molecule">p-hydroxymercuribenzoate</Entity>,
<Entity id="86" Type="small molecule">iodoacetamide</Entity>, and
<Entity id="87" Type="small molecule">oxygen </Entity>
<InhibitRelation id="88" Inhibitor="83, 84, 85, 86, 87"
Inhibitee="82">inhibit </InhibitRelation>
<Entity id="82" Antecedent="81">the enzyme</Entity>
```

The antecedent of the string "the enzyme" corresponds to a previous occurrence of:

```
<Entity id="81" Type="Protein">Formate dehydrogenase</Entity>
```

If a particular instance of a relation did not have an argument which could be interpreted from the document, then the argument value was annotated as *unspecified*.

4.2 Results

There were 95 instances of the *inhibit*-relation annotated in the 56 articles. Our system identified 84 of these instances: 56 were correct instances: (57% Recall) There were 21 instances in which one argument was identified correctly, but the second was not identified, and there was no False Positive argument: (22% Partial Recall). There were 8 False Positive (incorrect) answers: (Precision 90%). We understand it is important to consider the partial information retrieved. If the two arguments are absolutely necessary for any retrieval purpose, those instances which have only one argument specified can be easily filtered out.

Relation	Type Constraints	Data Set	Precision	Recall	Partial Recall
inhibit	No	MEDLINE	90.4%	58.9%	22%

Table 3: Summary of our Results

These results show a marked improvement over previously reported techniques from the literature. It is interesting to analyze the results corresponding to each submodule. In the Sentence-level (main predicate) module, 45 instances were returned of which 36 were correct instances. Only 4 were Partial correct answers and 5 were False Positives: (Precision 88.8%). In the Nominal-level module, 39 instances were returned: 20 of which were correct; 17 were partial correct answers and 2 were False Positives: (Precision 94.8%).

Module	Precision	Recall	Partial Recall	False Negatives
Sentence level	88.8%	37.8% %	4.2%	9.5%
Nominal level	94.8%	21%	17.8%	9.5%

Table 4: Results Per Module

We observed that there was a marked difference in precision between the sentence-level module and the nominal-level module. There was also a difference between the answers with one argument (partial correct answers), a difference which is also reflected in the corpus. Six instances of the markup had *unspecified* arguments, all of which were nominal instances (e.g., *A lupus inhibitor*); five instances of the markup had arguments which had no string representation, and the argument was deduced by the annotator from a preceding instance of the same relation in the context (these instances were marked as anaphoric). 17 instances were reiterations of a previously specified relation, of which 11 were nominal and 6 were verbal. This is summarized in table 5 below.

Relation Type	Anaphoric Instances	Reiterations	Redundancy
Main Pred	0%	6.3%	6.3
Nominal Rel	5.2%	11.5%	17.8%

Table 5: Corpus Redundancy Information

This supports the view that nominal instances of the relation tend to be more redundant, as a total of 23 instances were redundant (24.2%). We also counted how many arguments were anaphoric in nature (e.g. Entity 82 above). Eleven such instances were anaphoric. We applied our Anaphora Resolution module, resulting in the recognition of 10 anaphors from the 11 in the mark-up, with 8 correct and 2 incorrect results.[c]

5 Discussion and Conclusion

In this paper, we presented the results of our initial experiments on identifying and extracting biomolecular relations from the biomedical literature. Our performance represents a significant improvement over previously published results on comparable relation extraction from Medline. We attribute this performance to the integration of lexical semantic techniques, intensive corpus analytics over the corpus and the design of general automata over syntactic chunks. The results of this integration indicated that two separate modules would be most appropriate for relational parsing, allowing us to optimize verbal-based relations separately from the nominal-based cases. We are currently testing it over gold standards for new relational classes and extending the coverage of the grammar to improve the recall.

6 Bibliography

1. Andrade, Miguel A. and Valencia, Alfonso. Automatic annotation for biological sequences by extraction of keywords from MEDLINE abstracts. Development of a prototype system. AAAI, 1997.
2. Blasche, Christian; Andrade, Miguel A.; Ouzounis, Christos and Valencia, Alfonso. Automatic extraction of biological information from scientific text: protein-preotein interactions. AAAI,1999.
3. Buckley, C. Implemetation of the SMART Information Retrieval System. Technical Report 85-686, Cornell University, Computer Science. 1985.
4. Craven, Mark and Kumlien, Johan. Constructing Biological Knowledge Bases by Extracting information from Text Sources. In *In Proceedings of the 7th International Conference on Intelligent Systems for Molecular Biology* 1999.

[c]Due to space limitations, we refer the reader to `www.medstract.org` for an analysis and discussion of the false positive results from the present experiment.

5. Hindle, D. "Deterministic Parsing of Syntactic non-fluencies", *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, 1983.
6. Hishiki, T.; Collier, N.; Nobata, C.; Okazaki-Ohta, T.; Ogata, N.; Sekimizu, T.; Steiner, R.; Park, H.S. and Tsujii, J. Developing NLP Tools for Genome Informatics: An Information Extraction Perspective. In *In Proc. of Genome Informatics* pp81-90, Tokyo, Japan, 1998.
7. Humphreys, B. L., Lindberg, D. A. B., Schoolman H. M., and Barnett G. O. "The Unified Medical Language System: An informatics research collaboration", *Journal of the American Medical Informatics Association* 5:1, 1998.
8. Jenssen, T. K., Laegreid, A., Komorowski, J. and Hovig, E. A literature network of human genes for high-throughput analysis of gene expression. Nat Genet 28, 21-8, (2001).
9. Kennedy, C. and B. Boguraev Anaphora for Everyone: Pronominal Anaphora Resolution without a Parser. In: Proceedings of the 16th International Conference on Computational Linguistics (COLING), Vol. I, August 1996, Kopenhagen, 113-118.
10. Link, A. J., Eng, J., Schieltz, D. M., Carmack, E., Mize, G. J., Morris, D. R., Garvik, B. M. and Yates, J. R., 3rd. (1999). Direct analysis of protein complexes using mass spectrometry. Nature Biotechnology 17, 676-82.
11. McDonald, D. D. "Robust Partial Parsing through incremental multi-algorithm processing", in *Text-based Intelligent Systems*, P. Jacobs, ed. 1992.
12. Marcotte, E. M., Xenarios, I. and Eisenberg, D. Mining literature for protein-protein interactions. Bioinformatics 17, 359-63, (2001).
13. Ohta, Yoshihiro; Yamamoto, Yasunori; Okazaki, Tomoko; Uchiyama, Ikuo and Takagi, Toshihisa. Automatic Construction of Knowledge Base from Biological Papers. AAAI, 1997.
14. Proux, D. and Rechenmann, F. and Laurent, J. A Pragmatic Information Extraction Strategy for gathering Data on Genetic Interactions. In *ISMB 2000* 279-285, 2000.
15. Pustejovsky, J., S. Bergler, and P. Anick. (1993) "Semantic Methods for Corpus Analysis," *Computational Linguistics*, 19.2.
16. Pustejovsky, J., B. Boguraev, M. Verhagen, P. Buitelaar, M. Johnston, (1997) Semantic Indexing and Typed Hyperlinking AAAI Symposium on Language and the Web, Stanford, CA.
17. Pustejovsky, J.; Castaño, J.; Cochran, B.; Kotecki, M.; Morrell, M. Automatic Extraction of Acronym-meaning pairs from MEDLINE databases. In *Proceedings of Medinfo*, 2001.
18. Pustejovsky, J. and P. Hanks *Very Large Lexical Databasees: A Tutorial Primer* Association for Computational Linguistics, Toulouse, July, 2001.
19. Pustejovsky, J.; Castaño, J.; Cochran, B.; Kotecki, M. Exploiting Given versus New Information for Information Extraction Tasks *in preparation.*
20. Rindflesch, Thomas C; Rajan, Jayant V. and Hunter, Lawrence. Extracting Molecular Binding Relationships from Biomedical Text. In *Proceedings of the ANLP-NAACL 2000*, pages 188–195 Association for Computational Linguistics, 2000.
21. Sekimizu, T.; Park, H. S. and Tsujii, J. Identifying the Interaction between Genes and Gene Products based on Frequently Seen Verbs in Medline Abstracts. In *Proc. of Genome Informatics*, pp62-71, Tokyo, Japan, 1998.
22. Stapley, B. J. and Benoit, G. (2000). Biobibliometrics: information retrieval and visualization from co- occurrences of gene names in Medline abstracts. Pacific Symposium on Biocomputing 529-40.
23. Fukuda,K; Tsunoda,T.; Tamura, A. and Takagi, T. Toward Information Extraction: Identifying protein names from biological papers. In *PSB*, 707-718, 1998.