Classifying Semantic Relations in Bioscience Texts

Barbara Rosario
SIMS
UC Berkeley
Berkeley, CA 94720
rosario@sims.berkeley.edu

Marti A. Hearst
SIMS
UC Berkeley
Berkeley, CA 94720
hearst@sims.berkeley.edu

Abstract

A crucial step toward the goal of automatic extraction of propositional information from natural language text is the identification of semantic relations between constituents in sentences. We examine the problem of distinguishing among seven relation types that can occur between the entities "treatment" and "disease" in bioscience text, and the problem of identifying such entities. We compare five generative graphical models and a neural network, using lexical, syntactic, and semantic features, finding that the latter help achieve high classification accuracy.

1 Introduction

The biosciences literature is rich, complex and continually growing. The National Library of Medicine's MEDLINE database[1] contains bibliographic citations and abstracts from more than 4,600 biomedical journals, and an estimated half a million new articles are added every year. Much of the important, late-breaking bioscience information is found only in textual form, and so methods are needed to automatically extract semantic entities and the relations between them from this text. For example, in the following sentences, *hepatitis* and its variants, which are DISEASES, are found in different semantic relationships with various TREATMENTs:

(1) *Effect of interferon on hepatitis B*

(2) *A two-dose combined hepatitis A and B vaccine would facilitate immunization programs*

(3) *These results suggest that con A-induced hepatitis was ameliorated by pretreatment with TJ-135.*

In (1) there is an unspecified **effect** of the treatment *interferon* on *hepatitis B*. In (2) the *vaccine* **prevents** *hepatitis A and B* while in (3) *hepatitis* is **cured** by the treatment *TJ-135*.

We refer to this problem as *Relation Classification*. A related task is *Role Extraction* (also called, in the literature, "information extraction" or "named entity recognition"), defined as: given a sentence such as "*The fluoroquinolones for urinary tract infections: a review*", extract all and only the strings of text that correspond to the roles TREATMENT (*fluoroquinolones*) and DISEASE (*urinary tract infections*). To make inferences about the facts in the text we need a system that accomplishes both these tasks: the extraction of the semantic roles and the recognition of the relationship that holds between them.

In this paper we compare five generative graphical models and a discriminative model (a multilayer neural network) on these tasks. Recognizing subtle differences among relations is a difficult task; nevertheless the results achieved by our models are quite promising: when the roles are not given, the neural network achieves 79.6% accuracy and the best graphical model achieves 74.9%. When the roles are given, the neural net reaches 96.9% accuracy while the best graphical model gets 91.6% accuracy. Part of the reason for the

[1] http://www.nlm.nih.gov/pubs/factsheets/medline.html

Relationship	Definition and Example
Cure 810 (648, 162)	TREAT cures DIS *Intravenous immune globulin for recurrent spontaneous abortion*
Only DIS 616 (492, 124)	TREAT not mentioned *Social ties and susceptibility to the common cold*
Only TREAT 166 (132, 34)	DIS not mentioned *Flucticasone propionate is safe in recommended doses*
Prevent 63 (50, 13)	TREAT prevents the DIS *Statins for prevention of stroke*
Vague 36 (28, 8)	Very unclear relationship *Phenylbutazone and leukemia*
Side Effect 29 (24, 5)	DIS is a result of a TREAT *Malignant mesodermal mixed tumor of the uterus following irradiation*
NO Cure 4 (3, 1)	TREAT does not cure DIS *Evidence for double resistance to permethrin and malathion in head lice*
Total relevant: 1724 (1377, 347)	
Irrelevant 1771 (1416, 355)	TREAT and DIS not present *Patients were followed up for 6 months*
Total: 3495 (2793, 702)	

Table 1: Candidate semantic relationships between treatments and diseases. In parentheses are shown the numbers of sentences used for training and testing, respectively.

success of the algorithms is the use of a large domain-specific lexical hierarchy for generalization across classes of nouns.

In the remainder of this paper we discuss related work, describe the annotated dataset, describe the models, present and discuss the results of running the models on the relation classification and entity extraction tasks and analyze the relative importance of the features used.

2 Related work

While there is much work on role extraction, very little work has been done for relationship recognition. Moreover, many papers that claim to be doing relationship recognition in reality address the task of role extraction: (usually two) entities are extracted and the relationship is *implied* by the co-occurrence of these entities or by the presence of some linguistic expression. These linguistic patterns could in principle distinguish between different relations, but instead are usually used to identify examples of *one* relation. In the related work for statistical models there has been, to the best of our knowledge, no attempt to distinguish between *different* relations that can occur between the *same* semantic entities.

In Agichtein and Gravano (2000) the goal is to extract pairs such as (*Microsoft, Redmond*), where *Redmond* is the *location of* the organization *Microsoft*. Their technique generates and evaluates lexical patterns that are indicative of the relation. Only the relation *location of* is tackled and the entities are assumed given.

In Zelenko et al. (2002), the task is to extract the relationships *person-affiliation* and *organization-location*. The classification (done with Support Vector Machine and Voted Perceptron algorithms) is between positive and negative sentences, where the positive sentences contain the two entities.

In the bioscience NLP literature there are also efforts to extract entities and relations. In Ray and Craven (2001), Hidden Markov Models are applied to MEDLINE text to extract the entities PROTEINS and LOCATIONS in the relationship *subcellular-location* and the entities GENE and DISORDER in the relationship *disorder-association*. The authors acknowledge that the task of extracting relations is different from the task of extracting entities. Nevertheless, they consider positive examples to be all the sentences that simply contain the entities, rather than analyzing which relations hold between these entities. In Craven (1999), the problem tackled is relationship extraction from MEDLINE for the relation *subcellular-location*. The authors treat it as a text classification problem and propose and compare two classifiers: a Naive Bayes classifier and a relational learning algorithm. This is a two-way classification, and again there is no mention of whether the co-occurrence of the entities actually represents the target relation. Pustejovsky et al. (2002) use a rule-based system to extract entities in the *inhibit*-relation. Their experiments use sentences that contain verbal and nominal forms of the stem *inhibit*. Thus the actual task performed is the extraction of entities that are connected by some form of the stem *in-*

hibit, which by requiring occurrence of this word explicitly, is not the same as finding all sentences that talk about inhibiting actions. Similarly, Rindflesch et al. (1999) identify noun phrases surrounding forms of the stem *bind* which signify entities that can enter into molecular binding relationships. In Srinivasan and Rindflesch (2002) MeSH term co-occurrences within MEDLINE articles are used to attempt to infer relationships between different concepts, including diseases and drugs.

In the bioscience domain the work on relation classification is primary done through hand-built rules. Feldman et al. (2002) use hand-built rules that make use of syntactic and lexical features and semantic constraints to find relations between genes, proteins, drugs and diseases. The GENIES system (Friedman et al., 2001) uses a hand-built semantic grammar along with hand-derived syntactic and semantic constraints, and recognizes a wide range of relationships between biological molecules.

3 Data and Features

For our experiments, the text was obtained from MEDLINE 2001[2]. An annotator with biology expertise considered the titles and abstracts separately and labeled the sentences (both roles and relations) based solely on the content of the individual sentences. Seven possible types of relationships between TREATMENT and DISEASE were identified. Table 1 shows, for each relation, its definition, one example sentence and the number of sentences found containing it.

We used a large domain-specific lexical hierarchy (MeSH, Medical Subject Headings[3]) to map words into semantic categories. There are about 19,000 unique terms in MeSH and 15 main sub-hierarchies, each corresponding to a major branch of medical ontology; e.g., tree A corresponds to Anatomy, tree C to Disease, and so on. As an example, the word *migraine* maps to the term C10.228, that is, C (a disease), C10 (Nervous System Diseases), C10.228 (Central Nervous System Diseases). When there are multiple MeSH terms for one word, we simply choose the first one. These semantic features are shown to be very useful for our tasks (see Section 4.3). Rosario et al. (2002) demonstrate the usefulness of MeSH for the classification of the semantic relationships between nouns in noun compounds.

The results reported in this paper were obtained with the following features: the word itself, its part of speech from the Brill tagger (Brill, 1995), the phrase constituent the word belongs to, obtained by flattening the output of a parser (Collins, 1996), and the word's MeSH ID (if available). In addition, we identified the sub-hierarchies of MeSH that tend to correspond to treatments and diseases, and convert these into a tri-valued attribute indicating one of: disease, treatment or neither. Finally, we included orthographic features such as 'is the word a number', 'only part of the word is a number', 'first letter is capitalized', 'all letters are capitalized'. In Section 4.3 we analyze the impact of these features.

4 Models and Results

This section describes the models and their performance on both entity extraction and relation classification. Generative models learn the prior probability of the class and the probability of the features given the class; they are the natural choice in cases with hidden variables (partially observed or missing data). Since labeled data is expensive to collect, these models may be useful when no labels are available. However, in this paper we test the generative models on fully observed data and show that, although not as accurate as the discriminative model, their performance is promising enough to encourage their use for the case of partially observed data.

Discriminative models learn the probability of the class given the features. When we have fully observed data and we just need to learn the mapping from features to classes (classification), a discriminative approach may be more appropriate, as shown in Ng and Jordan (2002), but has other shortcomings as discussed below.

For the evaluation of the role extraction task, we calculate the usual metrics of precision, recall and F-measure. Precision is a measure of how many of

[2] We used the first 100 titles and the first 40 abstracts from each of the 59 files medline01n*.xml in Medline 2001; the labeled data is available at biotext.berkeley.edu

[3] http://www.nlm.nih.gov/mesh/meshhome.html

the roles extracted by the system are correct and recall is the measure of how many of the true roles were extracted by the system. The F-measure is a weighted combination of precision and recall[4]. Our role evaluation is very strict: every token is assessed and we do not assign partial credit for constituents for which only some of the words are correctly labeled. We report results for two cases: (i) considering only the relevant sentences and (ii) including also irrelevant sentences. For the relation classification task, we report results in terms of classification accuracy, choosing one out of seven choices for (i) and one out of eight choices for (ii). (Most papers report the results for only the relevant sentences, while some papers assign credit to their algorithms if their system extracts only one instance of a given relation from the collection. By contrast, in our experiments we expect the system to extract *all* instances of every relation type.) For both tasks, 75% of the data were used for training and the rest for testing.

4.1 Generative Models

In Figure 1 we show two static and three dynamic models. The nodes labeled "Role" represent the entities (in this case the choices are DISEASE, TREATMENT and NULL) and the node labeled "Relation" represents the relationship present in the sentence. We assume here that there is a single relation for each sentence between the entities[5].

The children of the role nodes are the words and their features, thus there are as many role states as there are words in the sentence; for the static models, this is depicted by the box (or "plate") which is the standard graphical model notation for replication. For each state, the features f_i are those mentioned in Section 3.

The simpler static models S1 and S2 do not assume an ordering in the role sequence. The dynamic models were inspired by prior work on HMM-like graphical models for role extraction (Bikel et al., 1999; Freitag and McCallum, 2000; Ray and Craven, 2001). These models consist of a

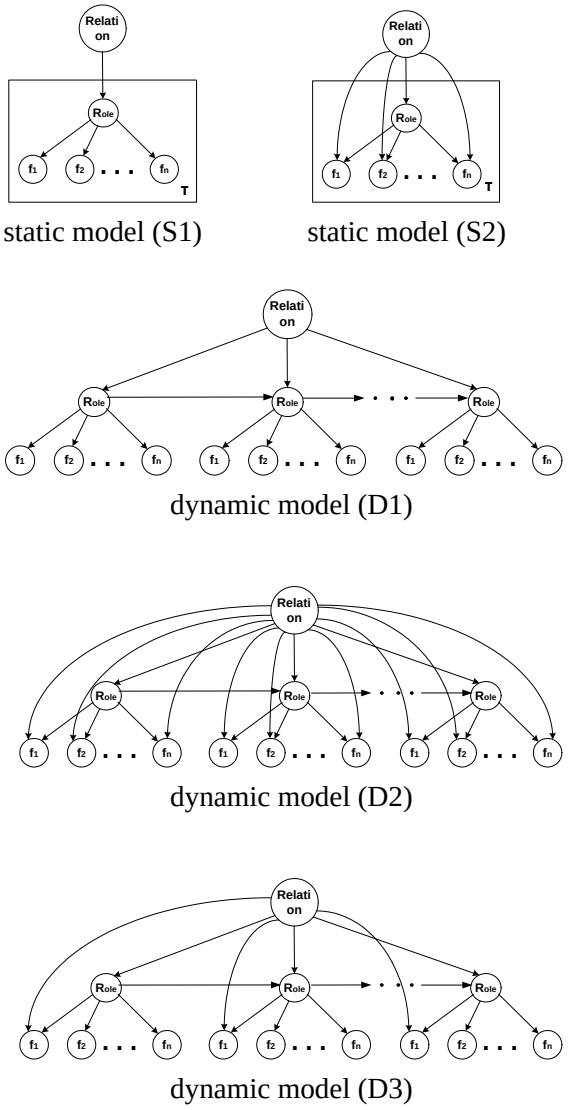

Figure 1: Models for role and relation extraction.

Markov sequence of states (usually corresponding to semantic roles) where each state generates one or multiple observations. Model D1 in Figure 1 is typical of these models, but we have augmented it with the Relation node.

The task is to recover the sequence of Role states, given the observed features. These models assume that there is an ordering in the semantic roles that can be captured with the Markov assumption and that the role generates the observations (the words, for example). All our models make the additional assumption that there is a relation that generates the role sequence; thus, these

[4] In this paper, precision and recall are given equal weight, that is, F-measure = $(2 * PRE * REC)/(PRE + REC)$.

[5] We found 75 sentences which contain more than one relationship, often with multiple entities or the same entities taking part in several interconnected relationships; we did not include these in the study.

Sentences	Static		Dynamic		
	S1	**S2**	**D1**	**D2**	**D3**
	No Smoothing				
Only rel.	0.67	0.68	0.71	0.52	0.55
Rel. + irrel.	0.61	0.62	0.66	0.35	0.37
	Absolute discounting				
Only rel.	0.67	0.68	0.72	**0.73**	0.73
Rel. + irrel.	0.60	0.62	0.67	**0.71**	0.69

Table 2: F-measures for the models of Figure 1 for *role* extraction.

models have the appealing property that they can *simultaneously* perform role extraction and relationship recognition, given the sequence of observations. In S1 and D1 the observations are independent from the relation (given the roles). In S2 and D2, the observations are dependent on both the relation and the role (or in other words, the relation generates not only the sequence of roles but also the observations). D2 encodes the fact that even when the roles are given, the observations depend on the relation. For example, sentences containing the word *prevent* are more likely to represent a "prevent" kind of relationship. Finally, in D3 only one observation per state is dependent on both the relation and the role, the motivation being that some observations (such as the words) depend on the relation while others might not (like for example, the parts of speech). In the experiments reported here, the observations which have edges from both the role and the relation nodes are the words. (We ran an experiment in which this observation node was the MeSH term, obtaining similar results.)

Model D1 defines the following joint probability distribution over relations, roles, words and word features, assuming the leftmost Role node is $Role_0$, and T is the number of words in the sentence:

$$\begin{aligned} & P(Rel, Role_0, .., Role_T, f_{10}, .., f_{n0}, ..., f_{1T}, .., f_{nT}) \\ = \quad & P(Rel)P(Role_0 \mid Rel) \prod_{j=1}^{n} P(f_{j0} \mid Role_0) \qquad (1) \\ & \prod_{t=1}^{T} P(Role_t \mid Role_{t-1}, Rel) \prod_{j=1}^{n} P(f_{jt} \mid Role_t) \end{aligned}$$

Model D1 is similar to the model in Thompson et al. (2003) for the extraction of roles, using a different domain. Structurally, the differences are (i) Thompson et al. (2003) has only one observation node per role and (ii) it has an additional node "on top", with an edge to the relation node, to represent a predicator "trigger word" which is always observed; the predicator words are taken from a fixed list and one must be present in order for a sentence to be analyzed.

The joint probability distributions for D2 and D3 are similar to Equation (1) where we substitute the term $\prod_{j=1}^{n} P(f_{jt}|Role_t)$ with $\prod_{j=1}^{n} P(f_{jt}|Role_t, Rel)$ for D2 and $P(f_{1t}|Role_t, Rel) \prod_{j=2}^{n} P(f_{jt}|Role_t)$ for D3. The parameters $P(f_{jt}|Role_t)$ and $P(f_{j0}|Role_0)$ of Equation (1) are constrained to be equal.

The parameters were estimated using maximum likelihood on the training set; we also implemented a simple absolute discounting smoothing method (Zhai and Lafferty, 2001) that improves the results for both tasks.

Table 2 shows the results (F-measures) for the problem of finding the most likely sequence of roles given the features observed. In this case, the relation is hidden and we marginalize over it[6]. We experimented with different values for the smoothing factor ranging from a minimum of 0.0000005 to a maximum of 10; the results shown fix the smoothing factor at its minimum value. We found that for the dynamic models, for a wide range of smoothing factors, we achieved almost identical results; nevertheless, in future work, we plan to implement cross-validation to find the optimal smoothing factor. By contrast, the static models were more sensitive to the value of the smoothing factor.

Using maximum likelihood with no smoothing, model D1 performs better than D2 and D3. This was expected, since the parameters for models D2 and D3 are more sparse than D1. However, when smoothing is applied, the three dynamic models achieve similar results. Although the additional edges in models D2 and D3 did not help much for the task of role extraction, they did help for relation classification, discussed next. Model D2

[6]To perform inference for the dynamic model, we used the junction tree algorithm. We used Kevin Murphy's BNT package, found at http://www.ai.mit.edu/ murphyk/Bayes/bnintro.html.

achieves the best F-measures: 0.73 for "only relevant" and 0.71 for "rel. + irrel.".

It is difficult to compare results with the related work since the data, the semantic roles and the evaluation are different; in Ray and Craven (2001) however, the role extraction task is quite similar to ours and the text is also from MEDLINE. They report approximately an F-measure of 32% for the extraction of the entities PROTEINS and LOCATIONS, and an F-measure of 50% for GENE and DISORDER.

The second target task is to find the most likely relation, i.e., to classify a sentence into one of the possible relations. Two types of experiments were conducted. In the first, the true roles are hidden and we classify the relations given only the observable features, marginalizing over the hidden roles. In the second, the roles are given and only the relations need to be inferred. Table 3 reports the results for both conditions, both with absolute discounting smoothing and without.

Again model D1 outperforms the other dynamic models when no smoothing is applied; with smoothing and when the true roles are hidden, D2 achieves the best classification accuracies. When the roles are given D1 is the best model; D1 does well in the cases when both roles are not present. By contrast, D2 does better than D1 when the presence of specific words strongly determines the outcome (e.g., the presence "prevention" or "prevent" helps identify the Prevent relation).

The percentage improvements of D2 and D3 versus D1 are, respectively, 10% and 6.5% for relation classification and 1.4% for role extraction (in the "only relevant", "only features" case). This suggests that there is a dependency between the observations and the relation that is captured by the additional edges in D2 and D3, but that this dependency is more helpful in relation classification than in role extraction.

For relation classification the static models perform worse than for role extraction; the decreases in performance from D1 to S1 and from D2 to S2 are, respectively (in the "only relevant", "only features" case), 7.4% and 7.3% for role extraction and 27.1% and 44% for relation classification. This suggests the importance of modeling the sequence of roles for relation classification.

To provide an idea of where the errors occur, Table 4 shows the confusion matrix for model D2 for the most realistic and difficult case of "rel + irrel.", "only features". This indicates that the algorithm performs poorly primarily for the cases for which there is little training data, with the exception of the ONLY DISEASE case, which is often mistaken for CURE.

4.2 Neural Network

To compare the results of the generative models of the previous section with a discriminative method, we use a neural network, using the Matlab package to train a feed-forward network with conjugate gradient descent.

The features are the same as those used for the models in Section 4.1, but are represented with indicator variables. That is, for each feature we calculated the number of possible values v and then represented an observation of the feature as a sequence of v binary values in which one value is set to 1 and the remaining $v - 1$ values are set to 0.

The input layer of the NN is the concatenation of this representation for all features. The network has one hidden layer, with a hyperbolic tangent function. The output layer uses a logistic sigmoid function. The number of units of the output layer is fixed to be the number of relations (seven or eight) for the relation classification task and the number of roles (three) for the role extraction task. The network was trained for several choices of numbers of hidden units; we chose the best-performing networks based on training set error. We then tested these networks on held-out testing data.

The results for the neural network are reported in Table 3 in the column labeled NN. These results are quite strong, achieving 79.6% accuracy in the relation classification task when the entities are hidden and 96.9% when the entities are given, outperforming the graphical models. Two possible reasons for this are: as already mentioned, the discriminative approach may be the most appropriate for fully labeled data; or the graphical models we proposed may not be the right ones, i.e., the independence assumptions they make may misrepresent underlying dependencies.

It must be pointed out that the neural network

Sentences	Input	B	Static		Dynamic			NN
			S1	S2	D1	D2	D3	
			No Smoothing					
Only rel.	only feat.	46.7	51.9	50.4	65.4	58.2	61.4	**79.8**
	roles given		51.3	52.9	66.6	43.8	49.3	**92.5**
Rel. + irrel.	only feat.	50.6	51.2	50.2	68.9	58.7	61.4	**79.6**
	roles given		55.7	54.4	82.3	55.2	58.8	**96.6**
			Absolute discounting					
Only rel.	only feat.	46.7	51.9	50.4	66.0	72.6	70.3	
	roles given		51.9	53.6	83.0	76.6	76.6	
Rel. + irrel.	only feat.	50.6	51.1	50.2	68.9	74.9	74.6	
	roles given		56.1	54.8	91.6	82.0	82.3	

Table 3: Accuracies of *relationship* classification for the models in Figure 1 and for the neural network (NN). For absolute discounting, the smoothing factor was fixed at the minimum value. B is the baseline of always choosing the most frequent relation. The best results are indicated in boldface.

is much slower than the graphical models, and requires a great deal of memory; we were not able to run the neural network package on our machines for the role extraction task, when the feature vectors are very large. The graphical models can perform both tasks simultaneously; the percentage decrease in relation classification of model D2 with respect to the NN is of 8.9% for "only relevant" and 5.8% for "relevant + irrelevant".

4.3 Features

In order to analyze the relative importance of the different features, we performed both tasks using the dynamic model D1 of Figure 1, leaving out single features and sets of features (grouping all of the features related to the MeSH hierarchy, meaning both the classification of words into MeSH IDs and the domain knowledge as defined in Section 3). The results reported here were found with maximum likelihood (no smoothing) and are for the "relevant only" case; results for "relevant + irrelevant" were similar.

For the role extraction task, the most important feature was the word: not using it, the GM achieved only 0.65 F-measure (a decrease of 9.7% from 0.72 F-measure using all the features). Leaving out the features related to MeSH the F-measure obtained was 0.69% (a 4.1% decrease) and the next most important feature was the part-of-speech (0.70 F-measure not using this feature). For all the other features, the F-measure ranged between 0.71 and 0.73.

For the task of relation classification, the MeSH-based features seem to be the most important. Leaving out the word again lead to the biggest decrease in the classification accuracy for a single feature but not so dramatically as in the role extraction task (62.2% accuracy, for a decrease of 4% from the original value), but leaving out all the MeSH features caused the accuracy to decrease the most (a decrease of 13.2% for 56.2% accuracy). For both tasks, the impact of the domain knowledge alone was negligible.

As described in Section 3, words can be mapped to different levels of the MeSH hierarchy. Currently, we use the "second" level, so that, for example, *surgery* is mapped to G02.403 (when the whole MeSH ID is G02.403.810.762). This is somewhat arbitrary (and mainly chosen with the sparsity issue in mind), but in light of the importance of the MeSH features it may be worthwhile investigating the issue of finding the optimal level of description. (This can be seen as another form of smoothing.)

5 Conclusions

We have addressed the problem of distinguishing between several different relations that can hold between two semantic entities, a difficult and important task in natural language understanding. We have presented five graphical models and a neural network for the tasks of semantic relation classification and role extraction from bioscience text. The methods proposed yield quite promising results. We also discussed the strengths and weaknesses of the discriminative and generative

	Prediction								Num. Sent.	Relation
Truth	**Vague**	**OD**	**NC**	**Cure**	**Prev.**	**OT**	**SE**	**Irr.**	(Train, Test)	**accuracy**
Vague	0	3	0	4	0	0	0	1	28, 8	0
Only DIS (OD)	2	69	0	27	1	1	0	24	492, 124	55.6
No Cure (NC)	0	0	0	1	0	0	0	0	3, 1	0
Cure	2	5	0	150	1	1	0	3	648, 162	92.6
Prevent	0	1	0	2	5	0	0	5	50, 13	38.5
Only TREAT (OT)	0	0	0	16	0	6	1	11	132, 34	17.6
Side effect (SE)	0	0	0	3	1	0	0	1	24, 5	20
Irrelevant	1	32	1	16	2	7	0	296	1416, 355	83.4

Table 4: Confusion matrix for the dynamic model D2 for "rel + irrel.", "only features". In column "Num. Sent." the numbers of sentences used for training and testing and in the last column the classification accuracies for each relation. The total accuracy for this case is 74.9%.

approaches and the use of a lexical hierarchy.

Because there is no existing gold-standard for this problem, we have developed the relation definitions of Table 1; this however may not be an exhaustive list. In the future we plan to assess additional relation types. It is unclear at this time if this approach will work on other types of text; the technical nature of bioscience text may lend itself well to this type of analysis.

Acknowledgements We thank Kaichi Sung for her work on the relation labeling and Chris Manning for helpful suggestions. This research was supported by a grant from the ARDA AQUAINT program, NSF DBI-0317510, and a gift from Genentech.

References

E. Agichtein and L. Gravano. 2000. Snowball: Extracting relations from large plain-text collections. *Proceedings of DL '00*.

D. Bikel, R. Schwartz, and R. Weischedel. 1999. An algorithm that learns what's in a name. *Machine Learning*, 34(1-3):211–231.

E. Brill. 1995. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics*, 21(4):543–565.

M. Collins. 1996. A new statistical parser based on bigram lexical dependencies. *Proc. of ACL '96*.

M. Craven. 1999. Learning to extract relations from Medline. *AAAI-99 Workshop on Machine Learning for Information Extraction*.

R. Feldman, Y. Regev, M. Finkelstein-Landau, E. Hurvitz, and B. Kogan. 2002. Mining biomedical literature using information extraction. *Current Drug Discovery*, Oct.

D. Freitag and A. McCallum. 2000. Information extraction with HMM structures learned by stochastic optimization. *AAAI/IAAI*, pages 584–589.

C. Friedman, P. Kra, H. Yu, M. Krauthammer, and A. Rzhetzky. 2001. Genies: a natural-language processing system for the extraction of molecular pathways from journal articles. *Bioinformatics*, 17(1).

A. Ng and M. Jordan. 2002. On discriminative vs. generative classifiers: A comparison of logistic regression and Naive Bayes. *NIPS 14*.

J. Pustejovsky, J. Castano, and J. Zhang. 2002. Robust relational parsing over biomedical literature: Extracting inhibit relations. *PSB 2002*.

S. Ray and M. Craven. 2001. Representing sentence structure in Hidden Markov Models for information extraction. *Proceedings of IJCAI-2001*.

T. Rindflesch, L. Hunter, and L. Aronson. 1999. Mining molecular binding terminology from biomedical text. *Proceedings of the AMIA Symposium*.

B. Rosario, M. Hearst, and C. Fillmore. 2002. The descent of hierarchy, and selection in relational semantics. *Proceedings of ACL-02*.

P. Srinivasan and T. Rindflesch. 2002. Exploring text mining from Medline. *Proceedings of the AMIA Symposium*.

C. Thompson, R. Levy, and C. Manning. 2003. A generative model for semantic role labeling. *Proceedings of EMCL '03*.

D. Zelenko, C. Aone, and A. Richardella. 2002. Kernel methods for relation extraction. *Proceedings of EMNLP 2002*.

C. Zhai and J. Lafferty. 2001. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of SIGIR '01*.