Bio-Medical Entity Extraction using Support Vector Machines

Koichi Takeuchi and **Nigel Collier**
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku
Tokyo 101-8430, Japan
{koichi,collier}@nii.ac.jp

Abstract

Support Vector Machines have achieved state of the art performance in several classification tasks. In this article we apply them to the identification and semantic annotation of scientific and technical terminology in the domain of molecular biology. This illustrates the extensibility of the traditional named entity task to special domains with extensive terminologies such as those in medicine and related disciplines. We illustrate SVM's capabilities using a sample of 100 journal abstracts texts taken from the {*human, blood cell, transcription factor*} domain of MEDLINE. Approximately 3400 terms are annotated and the model performs at about 74% F-score on cross-validation tests. A detailed analysis based on empirical evidence shows the contribution of various feature sets to performance.

1 Introduction

With the rapid growth in the number of published papers in the scientific fields such as medicine there has been growing interest in the application of Information Extraction (IE), (Thomas et al., 1999) (Craven and Kumlien, 1999), to help solve some of the problems that are associated with information overload. IE can benefit the medical sciences by enabling the automatic extraction of facts related to *prototypical* events such as those contained in patient records or research articles regarding molecular processes and their affect on human health. These facts can then be used to populate databases, aid in searching or document summarization and a variety of tasks which require the computer to have an intelligent understanding of the contents inside a document.

Our aim here is to show a state of the art method for identifying and classifying technical terminology. This task is an extension of the *named entity* task defined by the DARPA-sponsored Message Understanding Conferences (MUCs) (MUC, 1995) and is aimed at acquiring the shallow semantic building blocks that contribute to a high level understanding of the text. Although our study here looks at shallow semantics that can be captured using IE our basic goal is to join this with deep semantic representations so that computers can obtain a full understanding of the facts in a text using logical inference and reasoning. The scenario is that human experts will create taxonomies and axioms (*ontologies*) and by providing a small set of annotated examples, machine learning can take over the role of instance capturing though information extraction technology.

Recent studies into the use of supervised learning-based models for the named entity task have shown that models based on hidden Markov models (HMMs) (Bikel et al., 1997), and decision trees (Sekine et al., 1998), and maximum entropy (Borthwick et al., 1998) are much more generalisable and adaptable to new classes of words than systems based on hand-built patterns (including wrappers) and domain specific heuristic rules such as (Herzig and Johns, 1997).

The method we use is based on support vector machines (SVMs)(Vapnik, 1995), a state of the art model that has achieved new levels of performance in many classification tasks. In previous work we have shown SVMs to be superior to several other commonly used machine learning methods for named entity in previous experiments such as HMMs and C4.5 (*citations omitted*). This paper explores the underlying SVM model and shows through detailed empirical analysis the key features and parameter settings.

To show the application of SVMs to term extraction in unstructured texts related to the medical sciences we are using a collection of abstracts from PubMed's MEDLINE (MEDLINE, 1999). The MEDLINE database is an online collection of abstracts for published journal articles in biology and medicine and contains more than nine million articles. The collection we use in our tests is a controlled subset of MEDLINE obtained using three search keywords in the domain of molecular biology. From the retrieved abstracts 100 were randomly chosen for annotation by a human expert according to classes in a small top-level ontology.

In the remainder of this paper in Section (2) we outline the background to the task and the data set we are using; in Section (3) we described the basic advantages of SVMs and the formal model we are using as well as implementation specific issues such as the choice of feature set and report experimental results. In Section (4) we provide extensive results and a discussion of four sets of experiments we conducted that show the best feature sets and parameter settings in our sample domain.

2 Background

The names that we are trying to extract fall into a number of categories that are outside the definitions used for the traditional named-entity task used in MUC. For this reason we consider the task of term identification and classification to be an *extended named entity* task (NE+) in which the goal is to find types as well as individuals and where the term classes belong to an explicitly defined *ontology*. The use of an ontology allows us to associate human-readable terms in the domain with a set of computer-readable classes, relations, properties and axioms (Gruber, 1993).

The particular difficulties with identifying and classifying terms in scientific and technical domains are the size of the vocabulary (Lindberg et al., 1993), an open growing vocabulary (Lovis et al., 1995), irregular naming conventions as well as extensive cross-over in vocabulary between named entity classes. The irregular naming arises in part because of the number of researchers and practitioners from different fields who are working on the same knowledge discovery area as well as the large number of entities that need to be named. Despite the best efforts of major journals to standardize the terminology, there is also a significant problem with synonymy so that often an entity has more than one name that is widely used. In molecular biology for example class cross-over of terms may arise because many DNA and RNA are named after the protein with which they transcribe. This *semantic ambiguity* which is dependent on often complex contextual conditions is one of the main reasons why we need learnable models and why it is difficult to re-use existing term lists and vocabularies such as MeSH(NLM, 1997), UMLS (Lindberg et al., 1993) or those found in databases such as SwissProt (Bairoch and Apweiler, 1997). An additional obstacle to re-use is that the classification scheme used within an existing thesaurus or database may not be the same as the one in the users' ontology which may change from time to time as the consensus view of the structure of knowledge is refined.

Our work has focussed on identifying names belonging to the classes shown in Table 1 which are all taken from the domain of molecular biology . Example sentences from a marked up abstract are given in Figure 1. The ontology (Tateishi et al., 2000) that underlies this classification scheme describes a simple top-level model which is almost flat except for the *source* class which shows places where genetic activity occurs and has a number of sub-types. Further discussion of our use of deep semantic structures in the ontology is given elsewhere[1] and we will now focus our attention on the machine learning model used to capture low level semantics.

The training set we used in our experiments called Bio1 consists of 100 MEDLINE abstracts, marked up in XML by a doctoral-qualified domain expert

[1]Now being submitted for publication

Class	#	Description
PROTEIN	2125	proteins, protein groups, families, complexes and substructures.
DNA	358	DNAs, DNA groups, regions and genes
RNA	30	RNAs, RNA groups, regions and genes
SOURCE.cl	93	cell line
SOURCE.ct	417	cell type
SOURCE.mo	21	mono-organism
SOURCE.mu	64	multiorganism
SOURCE.vi	90	virus
SOURCE.sl	77	sublocation
SOURCE.ti	37	tissue

Table 1: Markup classes used in Bio1 with the number of word tokens for each class.

TI - Differential interactions of <NAME cl="PROTEIN" >Rel </NAME >- <NAME cl="PROTEIN" >NF-kappa B </NAME > complexes with <NAME cl="PROTEIN" >I kappa B alpha </NAME > determine pools of constitutive and inducible <NAME cl="PROTEIN" >NF-kappa B </NAME > activity.
AB - The <NAME cl="PROTEIN" >Rel </NAME >- <NAME cl="PROTEIN" >NF-kappa B </NAME > family of transcription factors plays a crucial role in the regulation of genes involved in inflammatory and immune responses. We demonstrate that in vivo, in contrast to the other members of the family, <NAME cl="PROTEIN" >RelB </NAME >associates efficiently only with <NAME cl="PROTEIN" >NF-kappa B1 </NAME > (<NAME cl="PROTEIN" >p105-p50 </NAME >) and <NAME cl="PROTEIN" >NF-kappa B2 </NAME > (<NAME cl="PROTEIN" >p100-p52 </NAME >), but not with <NAME cl="PROTEIN" >cRel </NAME > or <NAME cl="PROTEIN" >p65 </NAME >. The <NAME cl="PROTEIN" >RelB </NAME >- <NAME cl="PROTEIN" >p52 </NAME >heterodimers display a much lower affinity for <NAME cl="PROTEIN" >I kappa B alpha </NAME > than <NAME cl="PROTEIN" >RelB </NAME >- <NAME cl="PROTEIN" >p50 </NAME > heterodimers or <NAME cl="PROTEIN" >p65 </NAME > complexes.

Figure 1: Example MEDLINE sentence marked up in XML for molecular biology named-entities.

for the name classes given in Table 1. The number of named entities that were marked up by class are also given in Table 1 and the total number of words in the corpus is 29940. The abstracts were chosen from a sub-domain of molecular biology that we formulated by searching under the terms *human*, *blood cell*, *transcription factor* in the PubMed database. An example can be seen in Figure 1

3 Method

3.1 Basic model

The named entity task can be formulated as a type of classification task. In the supervised machine learning approach which we adopt here we aim to estimate a classification function f,

$$f : \chi^N \rightarrow \{\pm 1\} \tag{1}$$

so that error on unseen examples is minimized, using training examples that are N dimensional vectors x_i with class labels y_i. The sample set S with m examples is

$$S = (x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m) \in \chi^N \times \{\pm 1\} \tag{2}$$

The classification function returns either $+1$ if the test data is a member of the class, or -1 if it is not.

SVMs use linear models to discriminate between two classes. This raises the question of how can they be used to capture non-linear classification functions? The answer to this is by the use of a non-linear mapping function called a kernel,

$$\Phi : \chi^N \rightarrow \Gamma \tag{3}$$

which maps the input space χ^N into a feature space Γ. The kernel function k requires the evaluation of a dot product

$$k(x_i, x_j) = (\Phi(x_i) \cdot \Phi(x_j)) \tag{4}$$

Clearly the complexity of data being classified determines which particular kernel should be used and of course more complex kernels require longer training times.

By substituting $\Phi(x_i)$ for each training example in S we derive the final form of the optimal decision function f,

$$f(x) = sgn(\sum_{i}^{m} y_i \alpha_i k(x, x_i) + b) \quad (5)$$

where $b \in R$ is the bias and the Lagrange parameters α_i ($\alpha_i \geq 0$) are estimated using quadratic optimization to maximize the following function

$$w(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j}^{m} \alpha_i \alpha_j y_i y_j k(x_i, x_j) \quad (6)$$

under the constraints that

$$\sum_{i=1}^{m} \alpha_i y_i = 0 \quad (7)$$

and

$$0 \leq \alpha_i \leq C \quad (8)$$

for $i = 1, \ldots, m$. C is a constant that controls the ratio between the complexity of the function and the number of misclassified training examples.

The number of parameters to be estimated in α therefore never exceeds the number of examples. The influence of α_i basically means that training examples with $\alpha_i > 0$ define the decision function (the support vectors) and those examples with $\alpha_i = 0$ have no influence, making the final model very compact and testing (but not training) very fast. The point x is classified as positive (or negative) if $f(x) > 0$ (or $f(x) < 0$).

The kernel function we explored in our experiments was the polynomial function $k(x_i, x_j) = (x_i \cdot x_j + 1)^d$ for $d = 2$ which was found to be the best by (Takeuchi and Collier, 2002). Once input vectors have been mapped to the feature space the linear discrimination function which is found is the one which gives the maximum the geometric margin between the two classes in the feature space.

Besides efficiency of representation, SVMs are known to maximize their generalizability, making them an ideal model for the NE+ task. Generalizability in SVMs is based on statistical learning theory and the observation that it is useful sometimes to misclassify some of the training data so that the margin between other training points is maximized. This is particularly useful for real world data sets that often contain inseparable data points.

We implemented our method using the Tiny SVM package from NAIST[2] which is an implementation of Vladimir Vapnik's SVM combined with an optimization algorithm (Joachims, 1999). The multiclass model is built up from combining binary classifiers and then applying majority voting.

3.2 Generalising with features

In order for the model to be successful it must recognize regularities in the training data that relate preclassified examples of terms with unseen terms that will be encountered in testing.

Following on from previous studies in named entity we chose a set of linguistically motivated word-level features that include surface word forms, part of speech tags using the Brill tagger (Brill, 1992) and orthographic features. Additionally we used head-noun features that were obtained from preanalysis of the training data set using the FDG shallow parser from Conexor (Tapanainen and Järvinen, 1997). A significant proportion of the terms in our corpus undergo a local syntactic transformations such as coordination which introduces ambiguity that needs to be resolved by shallow parsing. For example *the c- and v-rel (proto) oncogenes* and *NF-kappaB and I kappa B protein families*. In these cases the head noun features *oncogene* and *family* would be added to each word in the constituent phrase. Head information is also needed when deciding the semantic category of a long term such as *tumor necrosis factor-alpha* which should be a PROTEIN, whereas *tumor necrosis factor (TNF) gene* and *tumor necrosis factor promoter region* should both be types of DNA.

Table 2 shows the orthographic features that we used. We hypothesize that such features will help the model to find similarities between known words that were found in the training set and unknown words (of zero frequency in the training set) and so overcome the unknown word problem.

In the experiments we report below we use feature vectors consisting of differing amounts of 'context' by varying the window around the focus word which is to be classified into one of the semantic classes. The full window of context considered in these experiments is ± 3 about the focus word.

[2] Tiny SVM is available from http:// http://cl.aist-nara.ac.jp/ taku-ku/software/ TinySVM/

Feature	Example	Feature	Example
DigitNumber	15	CloseSquare	]
SingleCap	M	Colon	:
GreekLetter	alpha	SemiColon	;
CapsAndDigits	I2	Percent	%
TwoCaps	RalGDS	OpenParen	(
LettersAndDigits	p52	CloseParen	)
InitCap	Interleukin	Comma	,
LowCaps	kappaB	FullStop	.
Lowercase	kinases	Determiner	the
Hyphon	-	Conjunction	and
Backslash	/	Other	* + #
OpenSquare	[		

Table 2: Orthographic features with examples

4 Experiment and Discussion

Results are given as F-scores (van Rijsbergen, 1979) using the CoNLL evaluation script and are defined as $F = (2PR)/(P+R)$. where P denotes Precision and R Recall. P is the ratio of the number of correctly found NE chunks to the number of found NE chunks, and R is the ratio of the number of correctly found NE chunks to the number of true NE chunks. All results are calculated using 10-fold cross validation.

4.1 Experiment 1: Effect of Training Set Size

The effect of context window size is shown along the top column of Tables 3 and 4. It can be seen that without exception more training data results in higher overall F-scores except at 10 per cent. where the result seems to be biased by the small sample, perhaps because one abstract is partly included in the training and testing sets. As we would expect larger training sets reduce the effects of data sparseness and allow more accurate models to be induced.

The rate of increase in improvement however is not uniform according to the feature sets that are used. For surface word features and head noun features the improvement in performance is consistently increasing whereas the improvement for using orthographic and part of speech features is quite erratic. This may be an effect of the small sample of training data that we used and we could not find any consistent explanation why this occurred.

As we observed before, the best overall result comes from using *Or hd*, i.e. surface words, orthographic and head features. However the total score hides the fact that three classes, i.e. SOURCE.mo, SOURCE.mu and SOURCE.ti actually perform worse when using anything but surface word forms (shown in Table 5). One possible explanation for this is that all of these classes have very small numbers of samples and the effect of adding features may be to blur the distinction between these and other more numerous classes in the model. However it is interesting to note that this does not happen with the RNA class which is also very small.

4.2 Experiment 2: Effect of Feature Sets

The effects of feature sets is of major importance in modelling named entity. In general we would like to identify only the necessary features that are required and to remove those that do not contribute to an increase in performance. This also saves time in training and testing.

The results from Tables 3 and 4 at 100 per cent. training data are summarized in Table 5 and clearly illustrate the value of surface word level features combined with orthographic and head noun features. Orthographic features allow us to capture many generalities that are not obvious at the surface word level such as *IkappaB alpha* and *IkappaB beta* both being PROTEINs and *IL-10* and *IL-2* both being PROTEINs.

The orthographic-head noun feature combination (*Or hd*) gives the best combined-class performance of 74.23 at 100 per cent. training data on a -2+2 window. Overall orthographic features combined with surface word features gave an improvement of between 4.9 and 22.0 per cent. at 100 per cent. data depending on window size over surface words alone. This was the biggest contribution by any feature except the surface words. Head information for example allowed us to correctly capture the fact that in the phrase *NF-kappaB consensus site* the whole of it is a DNA, whereas using orthographic information alone the SVM could only say that *NF-kappaB* was a PROTEIN and ignoring *consensus site*. We see a similar case in the phrase *primary NK cells* which is correctly classified as SOURCE.ct using head noun and orthographic features but only *NK cells* are found using orthographic features. This mistake is a natural consequence of a limited contextual view which the head noun feature helped to rectify.

Part of speech (*POS*) when combined with surface word features gave an improvement of between 7.9 and 11.7 per cent. at 100 per cent. data. The influence of POS though does not appear to be sustained when combined with other features and we found that it actually degraded performance slightly in many cases. This may possibly be due to either overlapping knowledge or more likely subtle inconsistencies between POS features and say, orthographic features. This could have occurred during training when the POS tagger was trained on an out of domain (news) text collection. It is possible that if the POS tagger was trained on in-domain texts it would make a greater and more consistent contribution. An example where orthographic features allowed correct classification but adding POS features resulted in failure is *p50* in the phrase *consisting of 50 (p50) - and 65 (p65) -kDa proteins*. Also in the phrase *c-Jun transactivation domain* where only *c-Jun* should be tagged as a protein, by using orthographic features and POS the model tags the whole phrase as a PROTEIN. This is probably because POS tagging gives a NN feature value (common noun) to each word. This is very general and does not allow the model to discriminate between them.

The fourth feature we investigated is related to syntactic rather than lexical knowledge. We felt though that there should exist a strong semantic relation between a word in a term and the head noun of that term. The results in Table 5 show that while the overall contribution of the *Head* feature is quite small, it is consistent for almost all classes.

5 Conclusion

The method we have shown for identifying and classifying technical terms has the advantage of being portable, not requiring large domain dependent dictionaries and no hand-made patterns were used. Additionally, since all the word level features are found automatically there is no need for intervention to create domain specific features. Indeed the only thing that is required is a quite small corpus of text containing entities tagged by a domain expert. For future work we are now looking at how to balance the scores from SVM for each word-class over the whole of a sentence using dynamic programming. Theoretically the existing SVM model cannot consider evidence from outside the context window, in particular evidence related to named entity class scores in the history and later in the sentence.

References

A. Bairoch and R. Apweiler. 1997. The SWISS-PROT protein sequence data bank and its new supplement TrEMBL. *Nucleic Acids Research*, 25:31–36.

D. Bikel, S. Miller, R. Schwartz, and R. Wesichedel. 1997. Nymble: a high-performance learning name-finder. In *Proceedings of the Fifth Conference on Applied Natural Language Processing (ANLP'97), Washington D.C., USA.*, pages 194–201, 31 March – 3 April.

A. Borthwick, J. Sterling, E. Agichtein, and R. Grishman. 1998. Exploiting diverse knowledge sources via maximum entropy in named entity recognition. In *Proceedings of the Sixth Workshop on Very Large Corpora (WVLC'98), Montreal, Canada*, pages 152–160.

E. Brill. 1992. A simple rule-based part of speech tagger. In *Third Conference on Applied Natural Language Processing – Association for Computational Linguistics, Trento, Italy*, pages 152–155, 31st March – 3rd April.

M. Craven and J. Kumlien. 1999. Constructing biological knowledge bases by extracting information from text sources. In *Proceedings of the 7th International Conference on Intelligent Systemps for Molecular Biology (ISMB-99)*, pages 77–86, Heidelburg, Germany, August 6–10.

T. R. Gruber. 1993. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 6(2):199–221.

T. Herzig and M. Johns. 1997. Extraction of medical information from textual sources: a statistical variant of the boundary word method. In *Proceedings of the American Medical Informatics Association (AMIA) 1997 Annual Fall Symposium, Nashville, USA*, 25–29 October.

T. Joachims. 1999. Making large-scale SVM learning practical. In B. Scholkopf, C. Burges, and A. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*. MIT Press.

Donald A.B. Lindberg, L. Humphreys, Betsy, and T. McCray, Alexa. 1993. The unified medical language system. *Methods of Information in Medicine*, 32:281–291.

Feature Set & Window Size	Percentage of data used in experiment									
	10	20	30	40	50	60	70	80	90	100
Wd -10	58.52	47.30	51.44	52.40	52.37	52.30	51.29	53.24	55.57	56.06
Wd -1+1	55.35	48.15	53.91	54.50	56.02	55.30	55.92	58.98	60.28	61.55
Wd -2+2	46.87	40.73	47.92	49.64	53.31	53.20	55.01	56.95	59.40	62.04
Wd -3+2	46.12	38.55	44.19	47.93	49.50	50.50	51.21	54.76	56.66	60.25
Wd -3+3	44.83	35.37	42.67	45.24	46.78	49.10	49.66	54.01	55.59	58.83
Or -10	60.33	55.08	63.49	63.41	64.09	63.04	62.97	62.64	64.59	65.63
Or -1+1	65.35	58.69	66.63	68.18	69.20	68.74	69.55	69.32	71.02	72.13
Or -2+2	60.84	58.90	66.44	67.17	69.88	68.81	69.68	69.62	71.41	72.12
Or -3+2	62.48	59.21	65.64	66.69	67.56	67.25	68.37	68.94	69.92	71.69
Or -3+3	59.61	58.65	64.95	65.68	67.11	66.65	67.85	68.84	69.54	71.78
Head -10	58.51	47.10	51.99	52.74	52.44	52.01	53.09	53.79	55.97	57.01
Head -1+1	57.50	50.00	55.81	57.88	58.03	57.84	58.81	61.08	62.64	63.93
Head -2+2	49.43	45.92	53.40	53.75	57.52	56.94	59.33	61.29	63.36	64.67
Head -3+2	46.51	39.42	49.39	49.75	54.54	54.81	56.95	58.13	59.25	61.96
Head -3+3	45.79	40.81	47.52	48.11	53.58	53.50	55.95	57.02	59.06	61.52
POS -10	61.62	52.89	61.14	62.04	62.62	61.51	61.05	60.78	62.71	62.63
POS -1+1	61.24	57.25	63.83	62.94	65.35	64.82	67.40	66.47	67.43	68.37
POS -2+2	57.52	53.11	59.39	59.98	62.86	62.16	63.72	64.17	64.56	66.92
POS -3+2	56.81	54.55	56.53	56.26	59.60	59.40	61.42	61.86	63.41	64.90
POS -3+3	54.76	53.28	56.79	55.02	57.46	57.66	59.60	59.89	62.39	63.50

Table 3: F-scores on Bio1 showing the effects of training set size, feature sets, and context window sizes. *Wd*: surface word level features; *Or*: Orthographic features; *Head*: Head noun features; *POS*: part of speech features.

Feature Set & Window Size	Percentage of data used in experiment									
	10	20	30	40	50	60	70	80	90	100
Or hd -10	62.16	57.80	64.31	65.70	65.20	63.84	64.90	64.73	66.46	67.31
Or hd -1+1	64.84	60.52	68.42	68.25	68.82	69.34	71.31	71.88	72.60	73.38
Or hd -2+2	61.16	61.10	68.06	67.42	69.32	69.62	70.91	71.31	72.31	**74.23**
Or hd -3+2	61.54	60.06	65.87	66.33	67.43	68.36	70.28	70.15	70.81	72.95
Or hd -3+3	59.68	57.03	64.58	65.76	66.84	67.16	69.07	69.22	70.73	72.12
Or POS -10	61.48	54.04	63.20	63.92	64.11	64.74	63.23	63.62	64.87	66.28
Or POS -1+1	64.57	58.89	66.52	66.77	67.83	67.90	69.32	69.07	70.84	71.70
Or POS -2+2	61.48	58.56	63.37	65.44	67.01	66.74	68.21	68.55	70.09	71.87
Or POS -3+2	61.08	57.14	64.23	63.39	65.53	65.11	67.31	67.78	68.64	71.54
Or POS -3+3	57.92	57.12	62.86	62.36	65.48	64.41	66.10	66.64	68.22	70.46
POS hd -10	64.90	55.39	61.14	61.65	61.91	61.29	61.88	60.51	63.27	63.82
POS hd -1+1	62.25	57.25	63.66	64.81	64.64	65.57	67.78	67.63	68.69	69.68
POS hd -2+2	58.08	53.23	58.91	60.28	62.55	62.06	64.19	64.51	66.18	67.66
POS hd -3+2	57.09	53.20	56.58	57.75	59.34	59.14	62.19	62.93	64.23	65.41
POS hd -3+3	54.69	51.09	55.67	55.46	58.31	58.28	60.88	61.17	62.94	64.31
Or POS hd -10	63.70	56.63	63.29	65.11	64.72	64.14	64.40	64.04	66.01	67.41
Or POS hd -1+1	66.20	59.65	66.49	67.91	68.44	68.14	70.01	70.61	71.80	72.95
Or POS hd -2+2	61.62	58.03	64.76	65.16	66.45	67.26	69.00	69.86	70.83	72.56
Or POS hd -3+2	62.06	57.28	63.74	64.50	66.10	66.25	68.01	69.05	69.44	71.59
Or POS hd -3+3	59.12	56.51	62.43	62.61	65.37	65.09	66.89	67.80	69.36	71.25

Table 4: F-scores on Bio1 showing the effects of training set size, feature sets, and context window sizes. *Wd*: surface word level features; *Or*: Orthographic features; *Head*: Head noun features; *POS*: part of speech features.

NE+ Class	Feature Set							
	Wd	Or	Head	POS	Or hd	Or POS	POS hd	Or POS hd
DNA	44.53	56.49	50.88	47.33	**62.78**	58.12	47.30	59.19
PROTEIN	65.07	77.50	67.96	72.10	**78.99**	77.03	72.89	77.58
RNA	12.12	42.11	12.90	24.24	**43.24**	37.84	6.67	29.41
SOURCE.cl	52.63	57.14	51.52	54.79	**59.21**	55.90	56.94	59.87
SOURCE.ct	65.83	66.39	66.22	63.70	**69.32**	67.03	65.65	68.94
SOURCE.mo	**32.00**	16.67	9.09	17.39	17.39	16.67	17.39	17.39
SOURCE.mu	**61.02**	58.41	55.24	57.14	51.92	54.55	53.33	51.92
SOURCE.sl	55.22	62.86	62.69	51.20	**68.53**	62.41	54.84	63.38
SOURCE.ti	**23.26**	18.18	0.00	14.63	5.00	14.29	0.00	0.00
SOURCE.vi	76.54	75.16	79.50	73.68	**80.25**	74.84	75.00	73.33

Table 5: Class by class performance using a -2+2 window shown against feature sets. *Wd*: surface word level features; *Or*: Orthographic features; *Head*: Head noun features; *POS*: part of speech features.

C. Lovis, P. Michel, R. Baud, and J. Scherrer. 1995. Word segmentation processing: a way to exponentially extend medical dictionaries. *Medinfo*, 8:28–32.

MEDLINE. 1999. The PubMed database can be found at:. http://www.ncbi.nlm.nih.gov/PubMed/.

DARPA. 1995. *Proceedings of the Sixth Message Understanding Conference(MUC-6)*, Columbia, MD, USA, November. Morgan Kaufmann.

NLM. 1997. Medical subject headings, bethesda, MD. National Library of Medicine.

Satoshi Sekine, Ralph Grishman, and Hiroyuki Shinnou. 1998. A Decision Tree Method for Finding and Classifying Names in Japanese Texts. In *Proceedings of the Sixth Workshop on Very Large Corpora*, Montreal, Canada, August.

K. Takeuchi and N. Collier. 2002. Use of support vector machines in extended named entity recognition. In *Proceedings of the 6th Conference on Natural Language Learning 2002 (CoNLL-2002), Roth, D. and van den Bosch, A. (eds)*, pages 119–125, August 31st.

P. Tapanainen and T. Järvinen. 1997. A non-projective dependency parser. In *Proceedings of the 5th Conference on Applied Natural Language Processing, Washington D.C., Association of Computational Linguistics*, pages 64–71.

Y. Tateishi, T. Ohta, N. Collier, C. Nobata, K. Ibushi, and J. Tsujii. 2000. Building an annotated corpus in the molecular-biology domain. In *COLING'2000 Workshop on Semantic Annotation and Intelligent Content, Luxemburg*, 5th–6th August.

J. Thomas, D. Milward, C. Ouzounis, S. Pulman, and M. Carroll. 1999. Automatic extraction of protein interactions from scientific abstracts. In *Proceedings of the Pacific Symposium on Biocomputing'99 (PSB'99)*, pages 1–12, Hawaii, USA, January 4–9.

C. J. van Rijsbergen. 1979. *Information Retrieval*. Butterworths, London.

V. N. Vapnik. 1995. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York.