Proceedings of the ACL 2003 Workshop on Natural Language Processing in Biomedicine, pp. 49-56.

Effective Adaptation of a Hidden Markov Model-based Named Entity Recognizer for Biomedical Domain

Dan Shen[†‡] **Jie Zhang**[†‡] **Guodong Zhou**[†] **Jian Su**[†] **Chew-Lim Tan**[‡]
[†] Institute for Infocomm Research
21 Heng Mui Keng Terrace
Singapore 119613
[‡] Department of Computer Science
National University of Singapore
3 Science Drive 2, Singapore 117543
{shendan,zhangjie,zhougd,sujian}@i2r.a-star.edu.sg
{tancl}@comp.nus.edu.sg

Abstract

In this paper, we explore how to adapt a general Hidden Markov Model-based named entity recognizer effectively to biomedical domain. We integrate various features, including simple deterministic features, morphological features, POS features and semantic trigger features, to capture various evidences especially for biomedical named entity and evaluate their contributions. We also present a simple algorithm to solve the abbreviation problem and a rule-based method to deal with the cascaded phenomena in biomedical domain. Our experiments on GENIA V3.0 and GENIA V1.1 achieve the 66.1 and 62.5 F-measure respectively, which outperform the previous best published results by 8.1 F-measure when using the same training and testing data.

1 Introduction

As the research in biomedical domain has grown rapidly in recent years, a huge amount of nature language resources have been developed and become a rich knowledge base. The technique of named entity (NE) recognition (NER) is strongly demanded to be applied in biomedical domain. Since in previous work, many NER systems have been applied successfully in newswire domain (Zhou and Su 2002; Bikel et al. 1999; Borthwich et al. 1999), more and more explorations have been done to port existing NER system into biomedical domain (Kazama et al. 2002; Takeuchi et al. 2002; Nobata et al. 1999 and 2000; Collier et al. 2000; Gaizauskas et al. 2000; Fukuda et al. 1998; Proux et al. 1998). However, compared with those in newswire domain, these systems haven't got high performance. It is probably because of the following factors of biomedical NE (Zhang et al. 2003):

1. Some modifiers are often before basic NEs, e.g. *activated B cell lines*, and sometimes biomedical NEs are very long, e.g. *47 kDa sterol regulatory element binding factor*. This kind of factor highlights the difficulty for identifying the boundary of NE.

2. Two or more NEs share one head noun by using conjunction or disjunction construction, e.g. *91 and 84 kDa proteins*. It is hard to identify these NEs respectively.

3. An entity may be found with various spelling forms, e.g. *N-acetylcysteine, N-acetyl-cysteine, NAcetylCysteine, etc.* Since the use of capitalization is casual, the capitalization information may not be so evidential in this domain.

4. NE may be cascaded. One NE may be embedded in another NE, e.g. *<PROTEIN><DNA> kappa 3</DNA> binding factor </PROTEIN>*. More effort must be made to identify this kind of NE.

5. Abbreviations are frequently used in biomedical domain, e.g. *TCEd*, *IFN*, *TPA*, etc. Since abbreviations don't have many evidences for certain NE class, it is difficult to classify them correctly.

These factors above make NER in biomedical domain difficult. Therefore, it is necessary to ex-

plore more evidential features and more effective methods to cope with such difficulties.

In this paper, we will study how to adapt a general Hidden Markov Model (HMM)-based NE recognizer (Zhou and Su 2002) to biomedical domain. We specially explore various evidences for biomedical NE and propose methods to cope with abbreviations and cascaded phenomena. As a result, features (simple deterministic features, morphological features, part-of-speech features and head noun trigger features) and methods (abbreviation recognition algorithm and rule-based cascaded phenomena resolution) are integrated in our system. The experiment shows that system outperforms the best published system by 8.1 F-measure.

In Section 2, we will introduce the HMM-based NE recognizer briefly. In Section 3, we will focus on the features that we have used. The methods and the adaptations of different features will be discussed in detail. In Section 5 and 6, we will present the solutions of abbreviation and cascaded phenomena. Finally, our experiment results will be presented and the contributions of different features will be analyzed in Section 7.

2 HMM-based Named Entity Recognizer

Our system is adapted from a HMM-based NE recognizer, which has been proved very effective in MUC (Zhou and Su 2002).

The purpose of HMM is to find the most likely tag sequence $T_1^n = t_1 t_2 \cdots t_n$ for a given sequence of tokens $G_1^n = g_1 g_2 \cdots g_n$ that maximizes $P(T_1^n \mid G_1^n)$.

In token sequence G_1^n, the token g_i is defined as $g_i =< f_i, w_i >$, where w_i is the word and f_i is the feature set related with the word w_i.

In tag sequence T_1^n, each tag t_i consists of three parts: 1. Boundary category, which denotes the position of the current word in NE. 2. Entity category, which indicates the NE class. 3. Feature set, which will be discussed in Section 3.

When we incorporate a plentiful feature set in HMM, we will encounter data sparseness problem. An alternative back-off modeling approach by means of constraint relaxation is applied in our model (Zhou and Su 2002). It enables the decoding process effectively find a near optimal frequently occurred pattern entry in determining the NE tag probability distribution of current word.

Finally, the Viterbi algorithm (Viterbi 1967) is implemented to find the most likely tag sequence in the state space of the possible tag distribution based on the state transition probabilities. Furthermore, some constraints on the boundary category and entity category between two consecutive tags are applied to filter the invalid NE tags (Zhou and Su 2002).

3 Feature Set

3.1 Simple Deterministic Features (F_{sd})

The purpose of simple deterministic features is to capture the capitalization, digitalization and word formation information. This kind of features have been widely used in both newswire NER system, such as (Zhou and Su 2002), and biomedical NER system, such as (Nobata et al. 1999; Gaizauskas et al. 2000; Collier et al. 2000; Takeuchi and Collier 2002; Kazama et al. 2002). Based on the characteristics of biomedical NEs, we designed simple deterministic features manually. Table 1 shows the simple deterministic features with descending order of priority.

F_{sd} Name	Example
Comma	,
Dot	.
LRB	(
RRB	)
LSB	[
RSB	]
RomanDigit	II
GreekLetter	Beta
StopWord	in, at
ATCGsequence	AACAAAG
OneDigit	5
AllDigits	60
DigitCommaDigit	1,25
DigitDotDigit	0.5
OneCap	T
AllCaps	CSF
CapLowAlpha	All
CapMixAlpha	IgM
LowMixAlpha	kDa
AlphaDigitAlpha	H2A
AlphaDigit	T4
DigitAlphaDigit	6C2
DigitAlpha	19D

Table 1: Simple deterministic features

From Table 1, we can find that:

1. Features such as *comma*, *dot*, *StopWord*, etc. are designed intuitively to provide information to detect the boundary of NE.

2. Features *Parenthesis* is often used to indicate the definition of abbreviation in biomedical documents.

3. Features *GreekLetter* and *RomanDigit* are specially designed to capture the symbols frequently occurred in biomedical NE.

4. Feature *ATCG sequence* identify the similarity of words according to their word formations, e.g. *AACAAAG*, *CTCAGGA*, etc.

5. Features dealing with mixed alphabets and digits such as *AlphaDigitAlpha*, *CapMixAlpha*, etc. are beneficial for biomedical abbreviations.

Furthermore, we evaluate these features and compare with those used in MUC (Zhou and Su, 2002). The reported result of the simple deterministic features used in MUC can achieve F-measure of 74.1 (Zhou and Su 2002), but when they are used in biomedical domain, they only get F-measure of 24.3. By contrast, using the simple deterministic features we designed for biomedical NER, the system achieves F-measure of 29.4. According to the comparison, some findings may be concluded as follows:

1) Simple deterministic features are domain dependent, which suggests that it is necessary to design special features for biomedical NER.

2) Simple deterministic features have weaker predictive power for NE classes in biomedical domain than in newswire domain.

3.2 Morphological Feature (F_m)

Morphological information, such as prefix/suffix, is considered as an important cue for terminology identification. In our system, we get most frequent 100 prefixes and suffixes from training data as candidates. Then, each of these candidates is evaluated according to formula f1.

$$Wt_i = \frac{(\#IN_i - \#OUT_i)}{N_i} \qquad \text{(f1)}$$

in which, $\#IN_i$ is the number that prefix/suffix i occurs within NEs; $\#OUT_i$ is the number that prefix/suffix i occurs out of NEs; N_i is the total number of prefix/suffix i.

The formula assumes that the particular prefix/suffix, which is most likely inside NEs and least likely outside NEs, may be thought as a good evidence for distinguishing the NEs. The candidates with *Wt* above a certain threshold (0.7 in experiment) are chosen. Then, we calculated the frequency of each prefix/suffix in each NE class and group the prefixes/suffixes with the similar distribution among NE classes into one feature. This is because prefixes/suffixes with the similar distribution have the similar contribution, and it will avoid suffering from the data sparseness problem. Some of morphological features were listed in Table 2.

F_m Name	Prefix/Suffix	Example
sOOC	~cin	actinomycin
	~mide	Cycloheximide
	~zole	Sulphamethoxazole
sLPD	~lipid	Phospholipids
	~rogen	Estrogen
	~vitamin	dihydroxyvitamin
sCTP	~blast	erythroblast
	~cyte	thymocyte
	~phil	eosinophil
sPEPT	~peptide	neuropeptide
sMA	~ma	hybridoma
sVIR	~virus	cytomegalovirus

Table 2: Examples of morphological features

From Table 2, the suffixes *~cin, ~mide, ~zole* have been grouped into one feature *sOOC* because they all have the high frequency in the NE class O*therOrganicCompound* and relatively low frequencies in the other NE classes. In our system, totally 37 prefixes and suffixes were selected and grouped to 23 features.

3.3 Part-of-Speech Features (F_{pos})

In the previous NER research in newswire domain, part-of-speech (POS) features were stated not useful, as POS features may affect the use of some important capitalization information (Zhou and Su 2002). However, since more and more words with lower case are included in NEs, capitalization information in biomedical domain is not as evidential as it in newswire domain (Zhang et al. 2003). Moreover, since many biomedical NEs are descriptive and long, identifying NE boundary is not a trivial task. POS tagging can provide the evidence of noun phrase region based on word syntactic information and the noun phrases are most likely to be NE. Therefore, we reconsidered the POS tagging.

In previous research, (Kazama et al. 2002) make use of POS information and conclude that it only slightly improves performance. Moreover, (Collier et al. 2000; Nobata et al. 2000; Takeuchi and Collier. 2002) don't incorporate POS information in their systems. The probable reason explained by them is that since POS tagger they used is trained on newswire articles, the assigned POS tags are often incorrect in biomedical documents. On the whole, it can be concluded that POS information hasn't been well used in previous work.

In our experiment, a POS tagger was trained using 80% of GENIA V2.1 corpus (536 abstracts, 123K words) and evaluated on the rest 20% (134 abstracts, 29K words). We use GENIA corpus to train the POS tagger in order to let it be adapted for biomedical domain. As for comparison, we also trained the POS tagger on Wall Street Journal articles (2500 articles, 756K words) and tested on the 20% of GENIA corpus. The results are shown in Table 3.

Training set	Testing set	Precision
2500 WSJ articles	134 GENIA abstracts	84.31
536 GENIA abstracts		97.37

Table 3: Comparison of POS tagger using different training data

From Table 3, it can be found that POS tagger trained on the biomedical documents performs much better on the biomedical testing documents than that trained on WSJ articles. This is consistent with earlier explanation for why POS features are not so useful in biomedical NER (Nobata et al. 2000; Takeuchi and Collier 2002).

3.4 Semantic Trigger Features

Semantic trigger features are collected to capture the evidence of certain NE class based on the semantic information of some key words. Initially, we design two types of semantic triggers: head noun triggers and special verb triggers.

3.4.1 Head Noun Triggers (F_{hnt})

Head noun means the main noun or noun phrase of some compound words and describes the function or the property, e.g. "*B cells*" is the head noun for the NE "*activated human B cells*". Compared with the other words in NE, head noun is a much more decisive factor for distinguishing NE classes. For instance,

<OtherName>IFN-gamma treatment</OtherName>
<DNA>IFN-gamma activation sequence</DNA>

In our work, we extract uni-gram and bi-grams of head nouns automatically from training data, and rank them by frequency. According to the experiment, we selected 60% top ranked head nouns as trigger features for each NE class. Some examples are shown in Table 4.

In the future application, we may also extract the head nouns from some public resources to enhance the triggers.

1-gram	2-grams
PROTEIN	
interleukin	activator protein
interferon	binding protein
kinase	cell receptor
ligand	gene product
CELL TYPE	
lymphocyte	blast cell
astrocyte	blood lymphocyte
eosinophil	killer cell
fibroblast	peripheral monocyte
DNA	
DNA	X chromosome
breakpoint	alpha promoter
cDNA	binding motif
chromosome	promoter element

Table 4: Examples of head noun triggers

3.4.2 Special Verb Triggers (F_{svt})

Besides collecting the triggers, such as head noun triggers, from the NEs themselves, we also extract the triggers from the local contexts of the NEs. Recently, some frequently occurred verbs in biomedical document have been proved useful for extracting the interaction between entities (Thomas et al. 2000; Sekimizu et al. 1998). In biomedical NER, we have the intuition that particular verbs may also provide the evidence for boundary and NE class. For instance, the verb *bind* is often used to indicate the interaction between proteins.

In our system, we selected 20 most frequent verbs which occur adjacent to NE from training data automatically as the verb trigger features, which is shown in Table 5.

Special Verb Triggers	
activate	express
bind	induce
inhibit	interact
regulate	stimulate

Table 5: Examples of special verb triggers

4 Method for Abbreviation Recognition

Abbreviations are widely used in biomedical domain. Identifying the class of them constitutes an important and difficult problem (Zhang et al. 2003).

In our current system, we incorporate a method to classify abbreviation by mapping the abbreviation to its full form. This approach is based on the assumption that it is easier to classify the full form than abbreviation. In most cases, this assumption is valid because the full form has more evidences than its abbreviation to capture the NE class. Moreover, if we can map the abbreviation to its full form in the current document, the recognized abbreviation is still helpful for classifying the same forthcoming abbreviations in the same document, as in (Zhou and Su 2002).

In practice, abbreviation and its full form often occur simultaneously with parenthesis when first appear in biomedical documents. There are two cases:

1. full form (abbreviation)
2. abbreviation (full form)

Most patterns conform to the first case and if the content inside the parenthesis includes more than two words, the second case is assumed (Schwartz and Hearst 2003).

In these two cases, the use of parenthesis is both evidential and confusing. On one hand, it is evidential because it can provide the indication to map the abbreviation to its full form. On the other hand, it is confusing because it makes the annotation of NE more complicated. Sometimes, the abbreviation and its full form are annotated separately, such as
<CellType>human mononulear leukocytes</CellType>(<CellType>hMNL</CellType>),
and sometimes, they are all embedded in the whole entity, such as
<OtherName>leukotriene B4 (LTB4) generation</OtherName>.
Therefore, parenthesis needs to be treated specially. We develop an abbreviation recognition algorithm described in Figure 1.

In preprocessing stage, we remove the abbreviations and parentheses from the sentence, when the abbreviation is first defined. This measure will make the annotation simpler and the NE recognizer more effective. The main work in this stage is to judge which case the current pattern belongs to and record the original positions of the abbreviation and parenthesis.

After applying the HMM-based NE recognizer to the sentence, we restore the abbreviation and parenthesis to the original position in the sentence. Next, the abbreviation is classified. There are two priorities of the class (from high to low): the class of its full form identified by the recognizer, and the class of the abbreviation itself identified by the recognizer. At last, the same abbreviation occurring in the rest sentences of the current document are assigned the same NE class.

```
for each sentence S_i in the document{
  if exist parenthesis{
    judge the case of {
      "full form (abbr.)";
      "abbr. (full form)";
    }
    store the abbr. A and position P_a to a list;
    record the parenthesis position P_p;
    remove A and parenthesis from sentence;
    apply HMM-based NE recognizer to S_i;
    restore A and parenthesis into P_a, P_p;
    if P_p within an identified NE E with the class C_E
      parenthesis is included in E;
    else{
      parenthesis is not included;
      classify A to C_E;
      classify A in the rest part of document to C_E;
    }
  }
  else apply HMM-based NE recognizer to S_i;
}
```

Figure 1: Abbreviation recognition algorithm

5 Solution of Cascaded Phenomena

In (Zhang et al. 2003), they state that 16.57% of NEs in GENIA V3.0 have cascaded annotations, such as
<RNA><DNA>CIITA</DNA> mRNA</RNA>.
Currently, we only consider the longest NE and ignore the embedded NEs.

Based on the features described in section 3, our system counters some problems when dealing with cascaded NEs. The probable reason is that

the features we used are not so effective for this kind of NEs.

For instance, POS is based on the assumption that NE is most likely to be a noun phrase. For cascaded NE, this assumption may not always be valid because one NE may consist of two or more noun phrases connected by some special words, such as *TSH receptor specific T cell lines*.

Moreover, in section 3.4.1, we have shown that head noun is the significant clue for distinguishing NE classes. Even for cascaded NEs, head noun features are still effective to some extent, such as *IL-2 mRNA*. However, cascaded NEs sometimes contain two or more head nouns, which belong to different NE classes. For example, *<DNA>IgG Fc receptor type IC gene</DNA>*, in which *receptor* is the head noun of protein and *gene* is the head noun of DNA. In general, the latter head noun will be more important. Unfortunately, it seems that sometimes the shorter NE is more possible to be identified, such as *<protein>IgG Fc receptor</protein> type IC gene*.

On the whole, we have to explore an additional method to cope with the cascaded phenomena separately. In our experiment, we attempt to solve this problem based on some rules.

In GENIA corpus, we find that there are four basic types of cascaded NEs:

1. < <NE> head noun >
2. < modifier <NE> >
3. < <NE1> <NE2> >
4. < <NE1> word <NE2> >

Moreover, these cascaded NEs may be generated iteratively. For instance,

5. < modifier <NE> head noun >
6. < <NE1> <NE2> head noun >

The rules are constructed automatically from the cascaded NEs in training data. Corresponding to the four basic types of cascaded NEs mentioned before, we propose four patterns and apply them iteratively in each sentence:

1. <entity1> head noun → <entity2>
   e.g. <Protein> binding motif → <DNA>
2. <entity1> <entity2> → <entity3>
   e.g. <Lipid> <Protein> → <Protein>
3. modifier <entity1> → <entity2>
   e.g. anti <Protein> → <Protein>
4. <entity1> word <entity2> → <entity3>
   e.g. <Virus> infected <Multicell> → <Multicell>

In our system, 102 rules are incorporated to classify the cascaded NEs.

6 Experiments

6.1 GENIA Corpus

GENIA corpus is the largest annotated corpus in molecular biology domain available to public (Ohta et al. 2002). In our experiment, three versions are used:

- GENIA Version 1.1 (V1.1) -- It contains 670 MEDLINE abstracts. Since a lot of previous related work used this version, we use it to compare our result with others'.
- GENIA Version 2.1 (V2.1) -- It contains the same 670 abstracts as V1.1 and POS tagging. We use it to train and evaluate our POS tagger.
- GENIA Version 3.0 (V3.0) -- It contains 2000 abstracts, which is the superset of V1.1. We use it to get the latest result and find out the effect of training data size.

The annotation of NE is based on the GENIA ontology. In our task, we use 23 distinct NE classes. As for the conjunctive and disjunctive NEs, we ignore such cases and take the whole construction as one entity. In addition, for the cascaded annotations in V3.0, currently, we only consider the longest one level of the annotations.

6.2 Experimental Results

The system is evaluated using standard "precision/recall/F-measure", in which "F-measure" is defined as F-measure = (2PR) / (P+R).

We evaluate our NER system on both V3.0 and V1.1, each of which has been split into a training set and a testing set. As for V1.1, we divide the corpus into 590 abstracts (136K words) as training set and the rest 80 abstracts (17K words) as testing set. As for V3.0, we use the same testing set as V1.1 and the rest 1920 abstracts (447K words) as training set.

Corpus	P	R	F
Our system on V3.0	66.5	65.7	66.1
Our system on V1.1	63.8	61.3	62.5
Kazama's on V1.1	56.2	52.8	54.4

Table 6: Comparison of overall performance

Table 6 shows the overall performance of our system on V3.0 and V1.1, and the best reported system on V1.1 described in (Kazama et al. 2002). On V1.1, we use the same training and testing data and capture the same NE classes as (Kazama et al.

2002). Our system (62.5 F-measure) outperforms Kazama's (54.4 F-measure) by 8.1 F-measure. This probably benefits from the various evidential features and the effective methods we proposed. Furthermore, as our expectation, the performance achieved on V3.0 (66.1 F-measure) is better than that on V1.1 (62.5 F-measure), which indicate that our system still has some room for improvement with the larger training data set.

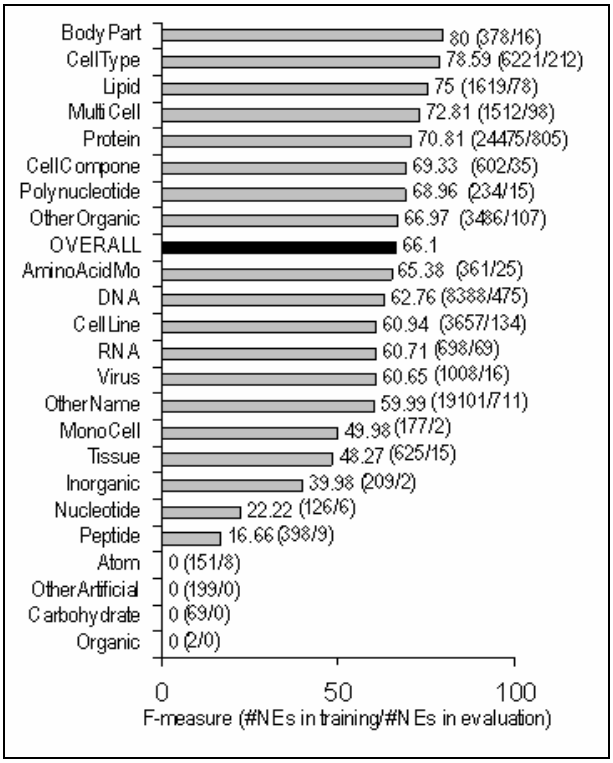

Figure 2: Performance of each NE class

In addition, Figure 2 shows the detailed performance chart of each NE class on V3.0. In the figure, the numbers in the parenthesis are the number that NEs of that class occur in training/testing data. It can be found that the performances vary a lot among the NE classes. Some NE classes that have very few training data, such as *Carbohydrate* and *Organism*, get extremely low performance.

In order to evaluate the contributions of different features, we evaluate our system using different combinations of features (Table 7).

From Table 7, several findings are concluded:

1) With only F_{sd}, our system achieves a basic level F-measure of 29.4.

2) F_m shows the positive effect with 2.4 F-measure improvement based on the basic level. However, it only can slightly improve the performance (+1.2 F-measure) based on F_{sd}, F_{pos} and F_{hnt}. The probable reason is that the evidences included in F_m have already been captured by F_{hnt}. Moreover, the evidences captured by F_{hnt} are more accurate than that captured by F_m. The contribution made by F_m may come from where there is no indication of F_{hnt}.

F_{sd}	F_m	F_{pos}	F_{hnt}	F_{svt}	P	R	F
√					42.4	22.5	29.4
√	√				44.8	24.6	31.8
√	√	√			58.3	50.9	54.3
√		√	√		62.0	61.6	61.8
√	√	√	√		**64.4**	**61.7**	**63.0**
√	√	√	√	√	60.6	59.3	60.0

Table 7: Effects of different features on V3.0

3) F_{pos} is proved very beneficial as it makes great increase on F-measure (+22.5) based on F_{sd} and F_m.

4) F_{hnt} leads to an improvement of 8.7 F-measure based on F_{sd}, F_m and F_{pos}.

5) Out of our expectation, the use of F_{svt} decreases both precision and recall, which may be explained as the present and past participles of some special verbs often play the adjective-like roles inside biomedical NEs, such as *IL10-inhibited lymphocytes*.

	P	R	F
$F_{sd}+F_m+F_{pos}+F_{hnt}$	64.4	61.7	63.0
+abbr. recog. algorithm	64.6	62.5	63.5
+rule-based casc. method	66.2	65.8	66.0
+both	**66.5**	**65.7**	**66.1**

Table 8: Effects of solution for abbr. and casc.

From Table 8, it can be found that the abbreviation recognition method slightly improves the performance by 0.5 F-measure. The probable reason is that the recognition of abbreviation relies too much on the recognition of its full form. Once the full form is wrongly classified, the abbreviation and the forthcoming ones throughout the document are wrong altogether. In the near future, the predefined abbreviation dictionary may be incorporated to enhance the decision of NE class.

Moreover, it can be found that the rule-based method effectively solves the problem of cascaded phenomena and shows prominent improvement

(+3.0 F-measure) based on the performance of "F_{sd}+F_m+F_{pos}+F_{hnt}".

7 Conclusion

In the paper, we describe our exploration on how to adapt a general HMM-based named entity recognizer to biomedical domain. We integrate various evidences for biomedical NER, including lexical, morphological, syntactic and semantic information. Furthermore, we present a simple algorithm to solve the abbreviation problem and a rule-based method to deal with the cascaded phenomena. Based on such evidences and methods, our system is successfully adapted to biomedical domain and achieves significantly better performance than the best published system. In the near future, more effective abbreviation recognition algorithm and some pre-defined NE lists for some classes may be incorporated to enhance our system.

Acknowledgements

We would like to thank Mr. Tan Soon Heng for his support of biomedical knowledge.

References

M. Bikel Danie, R.Schwartz and M. Weischedel Ralph. 1999. An Algorithm that Learns What's in a Name. In *Proc. of Machine Learning (Special Issue on NLP)*.

A. Borthwick. 1999. A Maximum Entropy Approach to Named Entity Recognition. *Ph.D. Thesis. New York University*.

N. Collier, C. Nobata, and J. Tsujii. 2000. Extracting the names of genes and gene products with a hidden Markov model. In *Proc. of COLING 2000*, pages 201-207.

K. Fukuda, T. Tsunoda, A. Tamura, and T. Takagi. 1998. Toward information extraction: identifying protein names from biological papers. In *Proc. of the Pacific Symposium on Biocomputing'98 (PSB'98)*, pages 707-718, January.

R. Gaizauskas, G. Demetriou and K. Humphreys. Term Recognition and Classification in Biological Science Journal Articles. 2000. In *Proc. of the Computional Terminology for Medical and Biological Applications Workshop of the 2nd International Conference on NLP*, pages 37-44.

J. Kazama, T. Makino, Y.Ohta, and J. Tsujii. 2002. Tuning Support Vector Machines for Biomedical Named Entity Recognition. In *Proc. of the Workshop on Natural Language Processing in the Biomedical Domain (at ACL'2002)*, pages 1-8.

C. Nobata, N. Collier, and J. Tsujii. 1999. Automatic term identification and classification in biology texts. In *Proc. of the 5th NLPRS*, pages 369-374.

C. Nobata, N. Collier, and J. Tsujii. 2000. Comparison between tagged corpora for the named entity task. In *Proc. of the Workshop on Comparing Corpora (at ACL'2000)*, pages 20-27.

T. Ohta, Y. Tateisi, J. Kim, H. Mima, and J. Tsujii. 2002. The GENIA corpus: An annotated research abstract corpus in molecular biology domain. In *Proc. of HLT 2002*.

D. Proux, F. Rechenmann, L. Julliard, V. Pillet and B. Jacq. 1998. Detecting Gene Symbols and Names in Biological Texts: A First Step toward Pertinent Information Extraction. In *Proc. of Genome Inform Ser Workshop Genome Inform*, pages 72-80.

A.S. Schwartz and M.A. Hearst. 2003. A Simple Algorithm for Identifying Abbreviation Definitions in Biomedical Text. In *Proc. of the Pacific Symposium on Biocomputing (PSB 2003) Kauai*.

T. Sekimizu, H. Park, and J. Tsujii. 1998. Identifying the interaction between genes and gene products based on frequently seen verbs in medline abstracts. In *Proc. of Genome Informatics*, Universal Academy Press, Inc.

K. Takeuchi and N. Collier. 2002. Use of Support Vector Machines in Extended Named Entity Recognition. In *Proc. of the Sixth Conference on Natural Language Learning (CONLL 2002)*, pages 119-125.

J. Thomas, D. Milward, C. Ouzounis, S. Pulman, and M. Carroll. 2000. Automatic extraction of protein interactions from scientific abstracts. In *Proc. of the Pacific Symposium on Biocomputing'2000 (PSB'2000)*, pages 541-551, Hawaii, January.

A. J. Viterbi. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. In *Proc. of IEEE Transactions on Information Theory*, pages 260-269.

J. Zhang, D. Shen, G. Zhou, J. Su and C. Tan. 2003. Exploring Various Evidences for Recognition of Named Entities in Biomedical Domain. Submitted to EMNLP 2003.

G. Zhou and J. Su. 2002. Named Entity Recognition using an HMM-based Chunk Tagger. In *Proc. of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 473-480.