Empirical Textual Mining to Protein Entities Recognition from PubMed Corpus

Tyne Liang and Ping-Ke Shih

Department of Computer and Information Science
National Chiao Tung University, Hsinchu, Taiwan
tliang@cis.nctu.edu.tw

Abstract. Named Entity Recognition (NER) from biomedical literature is crucial in biomedical knowledge base automation. In this paper, both empirical rule and statistical approaches to protein entity recognition are presented and investigated on a general corpus GENIA 3.02p and a new domain-specific corpus SRC. Experimental results show the rules derived from SRC are useful though they are simpler and more general than the one used by other rule-based approaches. Meanwhile, a concise HMM-based model with rich set of features is presented and proved to be robust and competitive while comparing it to other successful hybrid models. Besides, the resolution of coordination variants common in entities recognition is addressed. By applying heuristic rules and clustering strategy, the presented resolver is proved to be feasible.

1 Introduction

Nowadays efficient automation of biomedical knowledge bases is urgently demanded to cope with the proliferation of biomedical researches. One crucial task involved in the automation is named entity recognition (NER) from biomedical literature. Similar to the recognition in general domains, the issues associated with biomedical entity recognition are open vocabulary, synonyms, boundaries and sense disambiguation. For example, the number of entries in SwissProt[1], a protein knowledge base, increases 277.36% in recent ten years. Each protein entity contains 2.54 synonyms in average, and each synonym contains 2.74 tokens in average.

Recent textual mining approaches useful to biomedical NER can be divided into rule-based, statistical and hybrid methods. Generally, rule-based approaches employ the information of terms and hand-craft rules to produce candidates which are then verified by using lexical analysis [1, 2, 5]. Yet rule-based methods require more domain knowledge and essentially lack of scalability. On the other hand, statistical models have been widely employed for their portability and scalability, such as Hidden Markov Model (HMM), Support Vector Model (SVM), Maximum Entropy (ME), and etc.. The recognition accuracy achieved by these models generally depends on a well-tagged training corpus and a well set

[1] SwissProt: http://us.expasy.org/sprot/

A. Montoyo et al. (Eds.): NLDB 2005, LNCS 3513, pp. 56–66, 2005.

of features [3, 6, 7, 9, 10]. Recently, hybrid approaches are proposed by combining coded rules, statistical model and dictionaries [4, 9]. As pointed in [10], it is expected that systems on a specified evaluation corpus with help of dictionaries tend to perform better than the general ones without help of any dictionaries. For example, the recognition performance is significantly improved when dictionary and rules are applied at post-processing together with a ME-based recognition mechanism in [4].

In this paper, recognition for protein entities from PubMed[2] corpus is addressed so as to facilitate the automation of protein interaction databases construction. In order to mine more features relevant to protein entities, we assembled a domain-specific protein corpus SRC (SwissProt Reference Corpus) which were extracted from SwissProt reference articles and we tagged it by simply matching SwissProt entry collection. Experimental results show that this new domain corpus is indeed helpful in generating informative patterns used in both rule-based and statistical models. It is also found that though the derived rules are fewer and less complicated than the ones used in the rule-based systems Kex [1] or Yapex [5], the presented model outperforms these two systems in terms of higher F-scores on a general corpus like GENIA 3.02p [3] and the domain-specific SRC.

On the other hand, a concise HMM-based model is presented with a back-off strategy to overcome data sparseness. With a rich set of features, the presented approaches could achieve promising results, by showing 76-77% F-scores on both GENIA corpus and SRC. Compared to the results achieved by some successful systems (the best 78% F-score for protein instances in [9]) which employ dictionaries or semantic lexicon lists, our results are competitive for three reasons. First, the recognition is done without any help of dictionaries or predefined lexicon lists. Second, the presented concise HMM is easily implemented and robust for different corpora. Third, our results are evaluated with strict annotation and enetities with the longest annotation are adopted in case they are in the nested forms.

Besides, this paper addresses the issue of coordination variants while we tackle with NER problems in written texts. To resolve such term variants, a method based on heuristic rules and clustering strategy is presented. Experimental results on GENIA corpus 3.0 proved its feasibility by achieving 88.51% recall and 57.04% precision on a test of 1850 sentences, including 174 variants.

2 Corpus Preparation

In order to boost protein entities recognition by mining more relevant information, we assembled a domain-specific corpus 'SwissProt Ref Corpus' ('SRC' for short), other than the widely-used tagged corpus like GENIA 3.02p. The new corpus was processed by employing Sentence Splitter[4] and Penn Treebank

[2] PubMed: http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=Pubmed
[3] http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/
[4] Sentence Splitter: http://l2r.cs.uiuc.edu/~cogcomp/

Tokenizer[5] for sentence segmentation and tokenization respectively. The POS-tagging is processed by a HMM-based POS tagger which was developed in our lab. By using GENIA 3.02p as training set, our POS-tagger could yield 95% F-score. For the sake of saving human efforts, annotating SRC with all the target entities was simply implemented with the following steps:

1. Tokens are split by space and hyphen.
2. Each token is converted to lower case except its initial character.
3. Entity is recognized if it matches an entity from SwissProt version 42.0.

The final specific SRC corpus is composed of 2,894 abstracts, which were particularly selected from SWISSPORT 82,740 reference articles in such a way that each of them contains at least six target entities. Table 1 lists the basic statistics for SRC and GENIA 3.02p.

Table 1. The statistics of SRC corpus and GENIA corpus 3.02p.

	SRC		**GENIA**	
	count	average	count	average
Abstract (a)	2,894		1,999	
Sentence (s)	28,154	9.73 (s/a)	18,572	9.29 (s/a)
Token (t)	740,001	255.70 (s/a)	490.469	245.36 (t/a)
		26.28 (t/s)		26.41 (t/s)
Protein (p)	31,977	11.05 (p/a)	32,525	11.05 (p/a)
Entity		1.14 (p/s)		1.14 (p/a)
Entity Token (t)	57,878	1.81(t/p)	58,200	1.79 (t/p)

3 Coordination Variants Resolution

Coordination variants are one common type of variants in general written texts like MEDLINE records. For example there are 1598 coordination variants in GENIA 3.02p corpus and each variant contains 2.1 entities in average. Table 2lists three types of the regular expressions generalized from the GENIA 3.02p training corpus of 16,684 sentences (in which 1421 coordination variants are distributed in 1329 sentences). There #, H, T, and R indicate core, head, tail, and coordinate terms respectively. For example, in the coordination '91 and 84 kDa proteins', '91' and '84' are the core terms, 'kDa proteins' is the tail term, and 'and' is the coordinate term.

The variant resolution was implemented with finite state machines (FSM) which are verified by a test set of 1850 sentences in which 174 variants are distributed in 165 sentences. Experimental results showed that this approach yielded 91.38% recall and 42.06% precision (indicated as ̈baseline approachïn Table 3). In practice, the precision can be improved by presenting more number of FSMs so as to cover all possible variant patterns, yet it will slow down the resolving throughput. In order to increase the sensitivity of coordination identification, a simple term clustering is employed. Suppose terms t_i, t_j co-occur

[5] http://www.cis.upenn.edu/~treebank/tokenization.html

Table 2. Original patterns, expanded patterns, and examples.

	Regular Expression		Example
Type 1	Original	H#(R#)$^+$	human chromosomes 11p15 and 11p13
	Expanded	(H#R)$^+$H#	humman chromosomes 11p15 and human chromosome 11p13
Type 2	Original	#(R#)$^+$T	c-fos, c-jun, and EGR2 mRNA
	Expanded	#T(R#)$^+$T	c-fos mRNA, c-jun mRNA, and EGR2 mRNA
Type 3	Original	H#(R#)$^+$T	human T and B lymphocytes
	Expanded	#T(R#)$^+$T	human T lymphocytes and human B lymphocytes

in one coordination variant, and terms t_i, t_k co-occur in another one. Then we put t_i, t_j and t_k into one cluster. The clustering procedure was implemented recursively. With such term clustering strategy (indicated as 'unlimited-distance' in Table 3), the resolution precision is increased by 4%. This showed that the clustering approach is helpful to restrict the path movement in FSMs. To distinguish the closeness of the terms in the same cluster, we furthermore applied the Floyd-Warshall algorithm to cluster sets. That is, if terms t_i, t_j co-occur in a sentence and terms t_i, t_k co-occur in another one but t_j, t_k do not co-occur in any sentence, then the $dist(t_j, t_k) = 2$. With this clustering strategy, the precision became 57.04% (increasing 15% with respect to the baseline method) at the expense of lower recall.

Table 3. Accuracy of coordination variants identification in GENIA 3.02p.

	dist.	Variants	tp+fp	tp	Recall	Precision	F-Score
Baseline	N/A	174	378	159	91.38%	42.06%	57.61%
Term Clustering	unlimited	174	338	158	90.80%	46.75%	61.72%
	1	174	270	154	88.51%	57.04%	69.37%

4 Protein Entity Recognition

In this paper, protein entity recognition is approached and investigated by both rule-based and HMM models. The performance verification is implemented by using both SRC and GENIA 3.02p corpora in such a way that the corpora are divided into 90% for training phase and 10% for testing phase.

4.1 Rule-Based Approach

The rule-based recognition is implemented by employing the patterns of the protein nomenclature mined from SRC and GENIA corpora. The patterns are formed in terms of core, function or predefined terms. Core terms show the closest resemblance to regular proper names. Function terms describe the functions or characteristics of a protein. Table 4 shows the frequent regular expressions which 'C' indicates core term, 'F' indicates function term, and 'P' indicates predefined term, namely specifier, amino acid and unit.

Table 4. Top 5 regular expressions of protein entities in SRC and GENIA 3.02p.

Regular Expression	SRC	Regular Expression	GENIA
C^+	25.70%	C^+	69.64%
C^+F^+	21.22%	C^+F^+	8.14%
F^+	15.57%	C^+P^+	5.84%
F^+P^+	12.62%	F^+C^+	2.91%
C^+P^+	9.36%	F^+	2.35%

The function terms may be head or tail function term depending on the position they appear texts. From our observation of SRC, 58.48% head function terms appear before an initial uppercase token, and 74.07% tail function terms appear after an initial uppercase token or a specifier. We define 217 head function terms and 127 tail function terms. The rest of the terms other than predefined and function terms are treated as core terms candidates. The candidates may be the composition of common strings which are useful for identifying unknown words. For example, a common string 'CD' is acquired from a core term 'CD23', and then an unknown word 'CD25' will be seen as a core term.

The extraction of protein entities is done by six steps. The first three steps are aimed to produce the candidates by using term information. If a token is one of the three type terms, it will be annotated. Steps 4-6 are aimed to acquire protein entities as many as possible.

Step 1: boundary confirmation We scan the chunk forward (left to right) and backward (right to left) to fix entity boundaries by exploiting POS pattern information of protein entities, as shown in Tables 5 and 6.

Table 5. Top 5 POS patterns in SRC and GENIA.

POS Pattern	SRC	POS Pattern	GENIA
NN	79.38%	NN	67.57%
NN,CD	12.94%	JJ,NN	7.13%
JJ,NN	3.13%	NNS	7.11%
JJ,NN	3.02%	JJ,NNS	2.94%
CD,NN	0.26%	NN,CD	0.96%

Table 6. The top frequent POS tags at the first and the last positions of chunks.

	First POS tag		Last POS tag	
POS	SRC	GENIA	SRC	GENIA
CD	0.27%	0.43%	13.12%	1.91%
JJ	6.32%	13.23%	3.03%	0.57%
NN	93.12%	83.20%	83.43%	83.50%
NNS	0.01%	2.28%	0.08%	13.66%
VBN	0.14%	0.31%	0.08%	0.01%

Step 2: remove invalid single-token chunks A single-token chunk will be treated as invalid if (a) its characters are in lower case, and the token is not a protein entity in training data or (b) it is a predefined term only.

Step 3: remove invalid multi-token chunks by using a general set of domain-independent rules. A chunk will be removed if it composes of the followings: (a) the predefined terms, (b) the single uppercase English letters, (c) the punctuation marks, and (d) the conjunctions. After the three steps, 68.21% and 52.63% invalid tokens in SRC and GENIA are removed 98.58% and 96.93% accuracy rates respectively.

Step 4: mine the tokens surrounding protein entities This step is to acquire more protein entities. The pattern is formulated as '$< T_{-2}, T_{-1}, \#, T_1, T_2 >$', where '#' is token's number of the protein entity, and the token 'T_i' is the i^{th} token relative to the protein entity. Two measurements namely, confidence and occurrence are used to justify the usefulness of the patterns. Confidence is the ratio of the number of correct instances divided by the number of all instances in training data, and occurrence is the number of all instances in training data. Patterns are selected whenever their occurrence and confidence are greater than one and 0.8 respectively, because our system is expected to achieve 80% correct rate, which is the ratio of the number of correct instances divided by the number of all retrieved instances.

Step 5: mine the bag-of-word surrounding protein entities For each protein entity we collect its preceding two tokens and following two tokens. The non-confidence is used to filter the candidates and it is defined as the ratio of the negative instances to all instances. Patterns are recognized whenever non-confidence is greater than 0.8 since our system is expected to yield 80% correct rate.

Step 6: employ syntactic rules Hypernyms may appear in front of hyponyms, and one common pattern is 'NP_0 such as $\{\mathrm{NP}_1, \mathrm{NP}_2, \ldots, (\mathrm{and}|\mathrm{or})\ \}\ \mathrm{NP}_n$'. So we can mine those clue words by collecting the tokens preceding 'such as' and 'e.g.'. For example, 'protein' is the clue token of '... proteins, such as CBL and VAV, were phosphorylated on ...'. The clue words are the tokens of UMLS concepts and their corresponding synonyms which are tagged with 'protein' semantic type.

The model performance is evaluated in terms of precision (P), recall (R) and F-score (F) which is 2PR/(R+P). To present performance of rule-based systems, we use the notations of correct matching defined in [5]. Table 7 shows that the strict measure, which the proposed hit matches one answer key exactly, can yield 51%-52% F-Score. Table 7 shows that we can get higher F-score if we measure the performance with PNP ('protein name parts'), meaning each proposed token matches any token of the answer key. For example 'CD surface receptor' is treated as 'PNP' of 'activation of the CD28 surface receptor'. In practice, such kind of annotation result is acceptable. In addition, Table 7 also shows that the terms, mined from SRC, are adaptable since we can obtain almost the same performance results from GENIA corpus. Table 8 shows the improvement is obvious for steps 1 to 3, but steps 4 to 6 have little effect. On the other hand, the precision can be boosted obviously but not much for recall.

Table 7. Experimental results by rule-based approach.

	Notation	tp+sn	tp+fp	tp	recall	precision	F-Score
SRC	SLOPPY	3234	4782	2987	92.36%	62.46%	74.53%
	PNP	3234	4782	2859	88.40%	59.79%	71.33%
	STRICT	3234	4782	2077	64.22%	43.43%	51.82%
	LEFT	3234	4782	2620	81.01%	54.79%	65.37%
	RIGHT	3234	4782	2363	73.07%	49.41%	58.96%
	LorR	3234	4782	2907	89.89%	60.79%	72.53%
	Notation	tp+sn	tp+fp	tp	recall	precision	F-Score
GENIA	SLOPPY	3451	4923	3010	87.22%	61.14%	71.89%
	PNP	3451	4923	2837	82.21%	57.63%	67.76%
	STRICT	3451	4923	2123	61.52%	43.12%	50.70%
	LEFT	3451	4923	2765	80.12%	56.16%	66.04%
	RIGHT	3451	4923	2296	66.53%	46.64%	54.84%
	LorR	3451	4923	2938	85.13%	59.68%	70.17%

Table 8. The intermediate results of rule-based approach.

	Procedure	tp+sn	tp+fp	tp	recall	precision	F-Score
SRC	step1	3234	10480	2051	63.42%	19.57%	29.91%
	step1-2	3234	5493	2043	63.17%	37.19%	46.82%
	step1-3	3234	4911	2040	63.08%	41.54%	50.09%
	step1-4	3234	4977	2104	65.06%	42.27%	51.25%
	step1-5	3234	4781	2077	64.22%	43.33%	51.83%
	step1-6	3234	4782	2077	64.22%	43.43%	51.82%
	Procedure	tp+sn	tp+fp	tp	recall	precision	F-Score
GENIA	step1	3451	7911	2160	62.59%	27.30%	38.02%
	step1-2	3451	5173	2129	61.69%	41.16%	49.37%
	step1-3	3451	5082	2127	61.63%	41.85%	49.85%
	step1-4	3451	5164	2155	62.45%	41.73%	50.03%
	step1-5	3451	4915	2120	61.43%	43.13%	50.68%
	step1-6	3451	4923	2123	51.52%	43.12%	50.70%

4.2 HMM-Based Approaches

The statistical approach for NER is implemented by a concise HMM model (Concise-HMM) which employs a rich set of input features. Its performance is verified with SRC and GENIA 3.02p by comparing two other models, namely, traditional model (Traditional-HMM) and mutual information model (MI-HMM) which was presented in [9] and produced high F-scores in MUC-6 and MUC-7. The comparison is made in the same environment settings.

In this paper, all the models are trained with the same set of useful features including internal, external and global features. Internal features are those surface clues in tokens (e.g. initial character is upper case). There are 17 internal features mined from the training corpus. External features indicate the external information associated with tokens. We treated POS tags as our external features. Global features are the trigger nouns extracted from whole training

corpus by using Chi-square test. Besides, the complete-link clustering algorithm is applied to the mined nouns so as to reduce their dimensions. For window size of three sentences, we have 214 and 142 noun clusters in SRC and GENIA corpus respectively.

Traditional HMM. Given a token sequence $T_1^n = t_1 t_2 \dots t_n$, the goal is to find an optimal state sequence $S_1^n = s_1 s_2 \dots s_n$ that maximizes $\log Pr(S_1^n | T_1^n)$, the logarithm probability of state sequence S_1^n corresponding to the given token sequence T_1^n. By applying Bayes's rule to

$$Pr(S_1^n | T_1^n) = \frac{Pr(S_1^n | T_1^n)}{Pr(T_1^n)} \tag{1}$$

we have

$$\arg\ {}^{\max}_{S} \log Pr(S_1^n | T_1^n) = \arg\ {}^{\max}_{S} \log Pr(S_1^n | T_1^n) + \log Pr(S_1^n)) \tag{2}$$

where

$$Pr(T_1^n | S_1^n) = \prod_{i=1}^{n} Pr(t_i | s_i) \tag{3}$$

and

$$Pr(S_1^n) = \prod_{i=1}^{n} Pr(s_i | s_{i-1}) \tag{4}$$

with the assumption of conditional probability independence and considering preceding state. Therefore equation (2) can be rewritten as:

$$\arg\ {}^{\max}_{S} \log Pr(S_1^n | T_1^n) = \arg\ {}^{\max}_{S} \left(\sum_{i-1}^{n} (\log Pr(t_i | s_i) + \log Pr(s_i | s_{i-1})) \right) \tag{5}$$

MI-HMM. Different from traditional HMM, MI-HMM is aimed to maximize the equation:

$$\arg\ {}^{\max}_{S} \log Pr(S_1^n | T_1^n) = \arg\ {}^{\max}_{S} \left(\log Pr(S_1^n) + \log \frac{Pr(S_1^n, T_1^n)}{Pr(S_1^n) \bullet Pr(T_1^n)}) \right) \tag{6}$$

In order to simplify the computation, the mutual information independence is assumed to be:

$$MI(S_1^n, T_1^n) = \sum_{i=1}^{n} MI(s_i, T_1^n) \tag{7}$$

or

$$\log \frac{Pr(S_1^n, T_1^n)}{Pr(S_1^n) \bullet Pr(T_1^n)} = \sum_{i=1}^{n} \log \frac{Pr(s_i, T_1^n)}{Pr(s_i) \bullet Pr(T_1^n)} \tag{8}$$

Applying it to equation (6), we have:

$$\arg\ {}^{\max}_{S} \log Pr(S_1^n | T_1^n) = \arg\ {}^{\max}_{S} \left(\log Pr(S_1^n) - \sum_{i=1}^{n} \log Pr(s_i) + \sum_{i=1}^{n} \log Pr(s_i | T_1^n) \right) \tag{9}$$

Concise HMM. The presented concise HMM is based on the idea of maximizing the fundamental $\log Pr(S_1^n|T_1^n)$. In the equation (9), $\log Pr(S_1^n|T_1^n)$ and $\sum_{i=1}^{n} \log Pr(s_i)$ are found to carry less meaning because the weak probabilities of states and state transitions are merely 3-by-3 and 3-by-1 matrices respectively. Thus, a concise HMM can be obtained by simplifying the formula (9) to be equation (10):

$$\arg \max_S \log Pr(S_1^n|T_1^n) = \arg \max_S \log Pr(S_1^n) - \sum_{i=1}^{n} \log Pr(s_i|T_1^n) \qquad (10)$$

Since the concise HMM does not take its state transition into account, we put previous state in the model to ensure correct state induction. Because the presented HMM approach concerned many features mentioned above, it is possible to train a high-accuracy probability model. To overcome spareseness problem, we use a back-off strategy which aims at the token sequence T_1^n in $Pr(S_1^n|T_1^n)$ or in $Pr(s_i|T_1^n)$ where T_1^n represents not only a token sequence but also the full set of sequence's features. There are two back-off levels. First level is based on different combinations of tokens and their features, and T_1^n will be assigned in the descending order:

$< s_{-1}, t_{-1}, t_0, f_0 >, < s_{-1}, t_0, f_0 >, < s_{-1}, t_{-1}, f_0 >, < s_{-1}, f_0 >$

where f_i represents the feature set including internal, external and global features. t_i is a token, s_i expresses a HMM state, and i is the i^{th} one relative to current token. Second level is based on different combinations of features, and f_i in first level is assigned in the descending order:

$< f_i^I, f_i^E, f_i^G >, < f_i^I, f_i^E >, < f_i^I >$

where f_i^I , f_i^E and f_i^G represent internal, external and global features, respectively.

4.3 Method Comparisons

Method comparisons for the three HMM-based models were made on both SRC corpus and GENIA corpus in the same environment settings. We used the same back-off model for concise and mutual information HMM, but not for traditional HMM. Table 9 shows that concise HMM with rule-based features (i.e. concise-ruled) yielded the best result. Traditional HMM obtains good high precision, but low recall since we chose a severe probability model to get the best F-score. It is also noticed that the performance of MI-HMM turned out to be the worst because the back-off model was used to optimize concise HMM. On the other hand, Table 10 shows all kinds of features turned out to be positive effect ($f^E > f^I > f^G$) for concise HMM. Such result is similar to that concluded from [10]. Table 11 lists the comparisons of the presented approaches to other well-known approaches on the public evaluation GENIA 3.x corpus. It is noticed that the presented rule-based approach with its simple general rules outperformed the other two complicated rule-based systems. On the other hand, the performance of the presented concise HMM-based models is comparable to the best model presented in [4]. However, we do not need any dictionary or rules in our model.

Table 9. HMM-based model comparison.

	HMM	tp+sn	tp+fp	tp	recall	precision	F-Score
SRC	Concise	3234	2953	2355	72.82%	79.75%	76.13%
	Concise-ruled	3234	2949	2391	73.93%	81.08%	77.34%
	MI	3234	3439	2384	73.72%	69.32%	71.45%
	Traditional	3234	2396	2086	64.50%	87.06%	74.10%
	HMM	tp+sn	tp+fp	tp	recall	precision	F-Score
GENIA	Concise	3451	3285	2553	73.98%	77.72%	75.80%
	Concise-ruled	3451	3323	2596	75.22%	78.12%	76.65%
	MI	3451	3415	2305	66.79%	67.50%	67.14%
	Traditional	3451	2863	2263	65.58%	79.04%	71.68%

Table 10. The effects of features in concise HMM.

	Features	tp+sn	tp+fp	tp	recall	precision	F-Score	Diff.
SRC	All	3234	2953	2355	72.82%	79.75%	76.13%	
	All-f^G	3234	2951	2335	72.20%	79.13%	75.51%	-0.62%
	All-f^E	3234	2894	2284	70.62%	78.92%	74.54%	-1.59%
	All-f^I	3234	2941	2303	71.21%	78.31%	74.59%	-1.54%
	Features	tp+sn	tp+fp	tp	recall	precision	F-Score	Diff.
GENIA	ALL	3451	3285	2553	73.98%	77.72%	75.80%	
	All-f^G	3451	3267	2534	73.43%	77.56%	75.44%	-0.36%
	All-f^E	3451	3176	2442	70.76%	76.89%	73.70%	-2.10%
	All-f^I	3451	3213	2467	71.49%	76.78%	74.04%	-1.76%

Table 11. Comparison to other systems on GENIA corpus.

System	Method	GENIA	Recall	Precision	F-Score
Lee et. al. [3]	SVM	3.0p	78.80%	61.70%	69.20%
Lin et. al. [4]	ME-hybrid	3.01	77.00%	80.00%	78.50%
KeX	Rule-based	3.02p	43.67%	37.40%	40.29%
Yapex	Rule-based	3.02p	45.06%	54.17%	47.48%
Ours	Rule-based	3.02p	61.52%	43.12%	50.70%
	concise-HMM	3.02p	73.98%	77.72%	75.80%
	concise-ruled	3.02p	75.22%	78.12%	76.64%

5 Conclusions and Future Work

In this paper, we presented different textual mining strategies applicable to supporting full automation of protein entities recognition. Recognition for the entities in coordination variants is also concerned. To our best knowledge, our approach is the first one to cope with the term variants in the named entity extraction from biomedical texts. On the other hand, practical textual mining to protein entities recognition were presented by both rule and statistical models. Without the help of any dictionaries, the kernel recognition based on a concise HMM-based model turns out to be promising for protein entity extraction.

Future work includes the manual annotation correction of SRC for fine classification, exploitation of dictionaries for better recognition performance and the improvement of the resolution for coordination variants by using the semantic type information of biomedical thesaurus like UMLS. In addition, novel mining techniques to resolve other types of term variants should be explored for full NER automation.

Acknowledgements

This research is partially supported by MediaTek Research Center, National Chiao Tung University, Taiwan and partially supported by National Science Council under the contract NSC 93-2213-E-009-074.

References

1. Fukuda, K. and Tsunoda, T. and Tamura, A. and Takagi, T.: Towards Information Extraction: identifying Protein Names from Biological Papers. The 3rd Pacific Symposium on Biocomputing. (1998) 707-718.
2. Hou, W. J. and Chen, H. H.: Enhancing Performance of Protein Name Recognizers using Collocation. ACL 2003 Workshop on Natural Language Processing in Biomedicine, (2003) 25-32.
3. Lee, K.J. and Hwang, Y.S. and Rim, H.C.: Two-Phase Biomedical NE Recognition based on SVMs. ACL 2003 Workshop on Natural Language Processing in Biomedicine, (2003) 33-40.
4. Lin, Y. and Tsai, T. and Chiou, W. and Wu K. and Sung, T.-Y. and Hsu, W-L.: A Maximum Entropy Approach to Biomedical Named Entity Recognition. 4th Workshop on Data Mining in Bioinformatics (2004).
5. Olsson, F., Eriksson, G., Franzen, K., Asker, L., and Liden, P.: Notions of Correctness when Evaluating Protein Name Taggers. 19th International Conference on Computational Linguistics. (2002) 765-771.
6. Settles, B.: Biomedical Named Entity Recognition Using Conditional Random Fields and Rich Feature Sets. Int'l Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA), Geneva, Switzerland (2004).
7. Takeuchi, K. and Collier, N.: Bio-Medical Entity Extraction using Support Vector Machines. ACL 2003 Workshop on Natural Language Processing in Biomedicine, (2003) 57-64.
8. Tsuruoka, Y. and Tsujii, J.: Boosting Precision and Recall of Dictionary-based Protein Name Recognition. ACL 2003 Workshop on Natural Language Processing in Biomedicine (2003) 41-48.
9. Zhou, G.D. and Su, J.: Named Entity Recognition using an HMM-based Chunk Tagger. 40th Annual Meeting of the Association for Computational Linguistics (2002).
10. Zhou, G., Zhang, J., Su, J., Shen, D. and Tan, C. L.: Recognizing Names in Biomedical Texts: A Machine Learning Approach. Bioinformatics, Vol. 20, (2004)1178-1190.