Recognizing Biomedical Named Entities using Skip-chain Conditional Random Fields

Jingchen Liu Minlie Huang* Xiaoyan Zhu
Department of Computer Science and Technology
Tsinghua University, Beijing 100084, China
liu-jc04@mails.tsinghua.edu.cn
{aihuang, zxy-dcs}@tsinghua.edu.cn

Abstract

Linear-chain Conditional Random Fields (CRF) has been applied to perform the Named Entity Recognition (NER) task in many biomedical text mining and information extraction systems. However, the linear-chain CRF cannot capture long distance dependency, which is very common in the biomedical literature. In this paper, we propose a novel study of capturing such long distance dependency by defining two principles of constructing skip-edges for a skip-chain CRF: linking similar words and linking words having typed dependencies. The approach is applied to recognize gene/protein mentions in the literature. When tested on the BioCreAtIvE II Gene Mention dataset and GENIA corpus, the approach contributes significant improvements over the linear-chain CRF. We also present in-depth error analysis on inconsistent labeling and study the influence of the quality of skip edges on the labeling performance.

1 Introduction

Named Entity Recognition (NER) is a key task in most text mining and information extraction systems. The improvement in NER can benefit the final system performance. NER is a challenging task, particularly in the biomedical literature due to the variety of biomedical terminologies and the complicated syntactic structures.

Many studies have been devoted to biomedical NER. To evaluate biomedical NER systems, several challenge competitions had been held, such as BioNLP/NLPBA in 2004[1], BioCreAtIvE I in 2004 and BioCreAtIvE II in 2006[2]. The overview reports from these competitions, presenting state-of-the-art of biomedical NER studies, show that linear-chain Conditional Random Fields (CRF) is one of the most commonly used models and has the most competitive results (Yeh et al., 2005; Smith et al., 2008). Linear-chain CRF has also been successfully applied to other NLP tasks such as POS-tagging (Lafferty et al., 2001) and sentence chunking (Sha and Pereira, 2003). However, in most of these applications, only linear-chain CRF was fully exploited, assuming that only adjacent words are inter-dependent. The dependency between distant words, which occurs frequently in the biomedical literature, is yet to be captured.

In the biomedical literature, the repeated appearance of same or similar words in one sentence is a common type of long distance dependencies. This phenomenon is due to the complicated syntactic structures and the various biomedical terminologies in nature. See the following example:

> *"Both GH deficiency and impaired spinal growth may result in short stature, whereas the occurrence of early puberty in association with GH deficiency reduces the time available for GH therapy."*

the mentions of *GH* are repeated three times. If the entity are referred by a pronoun, the meaning of the sentence will be confusing and unclear because of the complex sentence structure. In this sentence:

> *"These 8-oxoguanine DNA glycosylases, hOgg1 (human) and mOgg1 (murine), are homologous to each other and to yeast Ogg1."*

the words *hOgg1*, *mOgg1* and *Ogg1* are homologous genes belonging to different species, having

* Corresponding author

[1] http://research.nii.ac.jp/~collier/workshops/JNLPBA04st.htm

[2] http://www.biocreative.org/

Proceedings of the 2010 Workshop on Biomedical Natural Language Processing, ACL 2010, pages 10–18,
Uppsala, Sweden, 15 July 2010. ©2010 Association for Computational Linguistics

very similar entity names. Some other types of long distance dependencies also occur frequently in the biomedical literature. For example, in this sentence

> *"Western immunoblot analysis detected p55gag and its cleavage products p39 and p27 in purified particles derived by expression of gag and gag-pol, respectively."*

the words *p55gag*, *p39* and *p27* conjuncted by *and*, have similar semantic meanings but they are separated by several tokens. A human curator can easily recognize such long distance dependencies and annotate these words consistently. However, when applying the linear-chain CRF, inconsistency errors in annotating these entities could happen due to the inability of representing long distance dependency.

In this paper, we present an approach of capturing long distance dependencies between words. We adopte the skip-chain CRF to improve the performance of gene mention recognition. We define two principles of connecting skip-edges for skip-chain CRF to capture long distance dependencies. The efficacy of the principles is investigated with extensive experiments. We test our method on two data sets and significant improvements are observed over the linear-chain CRF. We present in-depth error analysis on inconsistent labeling. We also investigat whether the quality of connected edges affect the labeling performance.

The remainder of this paper is organized as follows: We survey related studies in Section 2. We introduce linear-chain CRF and skip-chain CRF in Section 3. The method of connecting skip-chain edges is described in Section 4 . In Section 5 we present our experiments and in-depth analysis. We summarize our work in Section 6.

2 Related work

NER is a widely studied topic in text mining research, and many new challenges are seen in domain-specific applications, such as biomedical NER (Zhou et al., 2004). The dictionary based method is a common technique as biomedical thesauruses play a key role in understanding such text. Most dictionary based NER systems focused on: (1) integrating and normalizing different biomedical databases to improve the quality of the dictionary to be used; (2) improving matching strategies that are more suitable for biomedical terminologies; and (3) making filtering rules for post-processing to refine the matching results or to adjust the boundary of entities, see (Fukuda et al., 1998; Narayanaswamy et al., 2003; Yang et al., 2008). Many information extraction systems had a dictionary matching module to perform preliminary detection of named entities (Schuhmann et al., 2007; Kolarik et al., 2007; Wang et al., 2010).

Applying machine learning techniques generally obtains superior performance for the biomedical NER task. The automated learning process can induce patterns for recognizing biomedical names and rules for pre- and post-processing. Generally speaking, there are two categories of machine learning based methods: one treats NER as a classification task, while the other treats NER as a sequence labeling task. For the first category, Support Vector Machine (SVM) was a commonly adopted model (Kazama et al., 2002; Zhou et al., 2004). Lee et al. (2004) proposed a two-step framework to perform biomedical NER using SVM: firstly detecting the boundaries of named entities using classifiers; secondly classifying each named entity into predefined target types. For the second category, a sentence was treated as a sequence of tokens and the objective was to find the optimal label sequence for these tokens. The label space was often defined as {B,I,O}, where B indicates the beginning token of an entity, I denotes the continuing token and O represents the token outside an entity. The sequence labeling task can be approached by Hidden Markov Model (HMM), Conditional Random Field (CRF) , or a combination of different models (Zhou et al., 2005; Tatar and Cicekli, 2009).

Since proposed in (Lafferty et al., 2001), CRF has been applied to many sequence labeling tasks, including recognizing gene mentions from biomedical text (McDonald and Pereira, 2005). The Gene Mention Recognition task was included in both BioCreAtIvE I and BioCreAtIvE II challenges. CRF had been used in most of top performing systems in the Gene Mention Recognition task of BioCreAtIvE II (Smith et al., 2008). Some novel use of linear-chain CRF was proposed. For example, in (Kuo et al., 2007) labeling was performed in forward and backward directions on the same sentence and results were combined from the two directions. Huang et al. (2007) combines a linear-chain CRF and two SVM models to enhance the recall. Finkel et al. (2005) used Gibbs Sampling to add non-local dependencies into linear-chain CRF model for information extraction. However, the CRF models used in these systems were all linear-chain CRFs. To the best of our knowledge, no previous work has been done on using non-linear-chain CRF in the biomedical NER task.

Beyond the biomedical domain, skip-chain CRF has been used in several studies to model long distance dependency. In (Galley, 2006), skip edges were linked between sentences with non-local pragmatic dependencies to rank meetings. In (Ding et al., 2008), skip-chain CRF was used to detect the context and answers from online forums. The most close work to ours was in (Sutton and McCallum, 2004), which used skip-chain CRF to extract information from email messages announcing seminars. By linking the same words whose initial letter is capital, the method obtained improvements on extracting speakers' name. Our work is in the spirit of this idea, but we approach it in a different way. We found that the problem is much more difficult in the biomedical NER task: that is why we systematically studied the principles of linking skip edges and the quality of connected edges.

3 linear-chain and skip-chain CRF

Conditional Random Field is a probabilistic graphic model. The model predicts the output variables $\mathbf{y}$ for each input variables in $\mathbf{x}$ by calculating the conditional probability $p(\mathbf{y}|\mathbf{x})$ according to the graph structure that represents the dependencies between the $\mathbf{y}$ variables. Formally, given a graph structure over $\mathbf{y}$, the CRF model can be written as:

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{C_p \in \zeta} \prod_{\Psi_c \in C_p} \Psi_c(\mathbf{x}_c, \mathbf{y}_c; \theta_p) \quad (1)$$

$Z(\mathbf{x})$ is a normalization factor.

In this definition, the graph is partitioned into a set of cliques $\zeta = \{C_1, C_2, \ldots C_p\}$, where each C_p is a clique template. Each Ψ_c, called a factor, is corresponding to one edge in the clique c, and can be parameterized as:

$$\Psi_c(\mathbf{x}_c, \mathbf{y}_c; \theta_p) = \exp \sum_{k=1} \lambda_{pk} f_{pk}(\mathbf{x}_c, \mathbf{y}_c) \quad (2)$$

Each feature function $f_{pk}(\mathbf{x}_c, \mathbf{y}_c)$ represents one feature of $\mathbf{x}$ and the λ_{pk} is the feature weight.

In the training phrase, the parameters is estimated using an optimization algorithm such as limited memory BFGS etc. In the testing phrase, CRF finds the most likely label sequence for an unseen instance by maximizing the probability defined in (1).

In the NER task, one sentence is firstly tokenized into a sequences of tokens and each token can be seen as one word. Each node in the graph is usually corresponding to one word in a sentence. Each x variable represents a set of features for one word, and each y is the variable for the label of one word. Note that when one edge is linked between two words, the edge is actually linked between their corresponding y variables. The $\mathbf{y}$ label is one of $\{$B,I,O$\}$, in which B means the beginning word of an entity, I means the inside word of an entity, and O means outside an entity.

If we link each word with its immediate preceding words to form a linear structure for one sentence, we get a linear-chain CRF, defined as:

$$p_\theta(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{t=1}^{T} \Psi_t(y_t, y_{t-1}, \mathbf{x}) \quad (3)$$

This structure contains only one clique template. If we add an extra clique template that contains some skip edges between nonadjacent words, the CRF become a skip-chain CRF, formulated as follows:

$$p_\theta(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{t=1}^{T} \Psi_t(y_t, y_{t-1}, \mathbf{x}) \cdot \prod_{(u,v) \in \tau} \Psi_{uv}(y_u, y_v, \mathbf{x}) \quad (4)$$

τ is the edge set of the extra clique template containing skip edges. An illustration of linear-chain and skip-chain CRF is given in Figure 1. It is straightforward to change a linear-chain CRF to a skip-chain CRF by simply linking some additional skip edges. However, it must be careful to add such edges because different graph structures require different inference algorithms. Those inference algorithms may have quite different time complexity. For example, for the linear-chain CRF, inference can be performed efficiently and exactly by a dynamic-programming algorithm. However, for the non-linear structure, approximate inference algorithms must be used. Solving arbitrary CRF graph structures is NP-hard. In other word, we must be careful to link too many

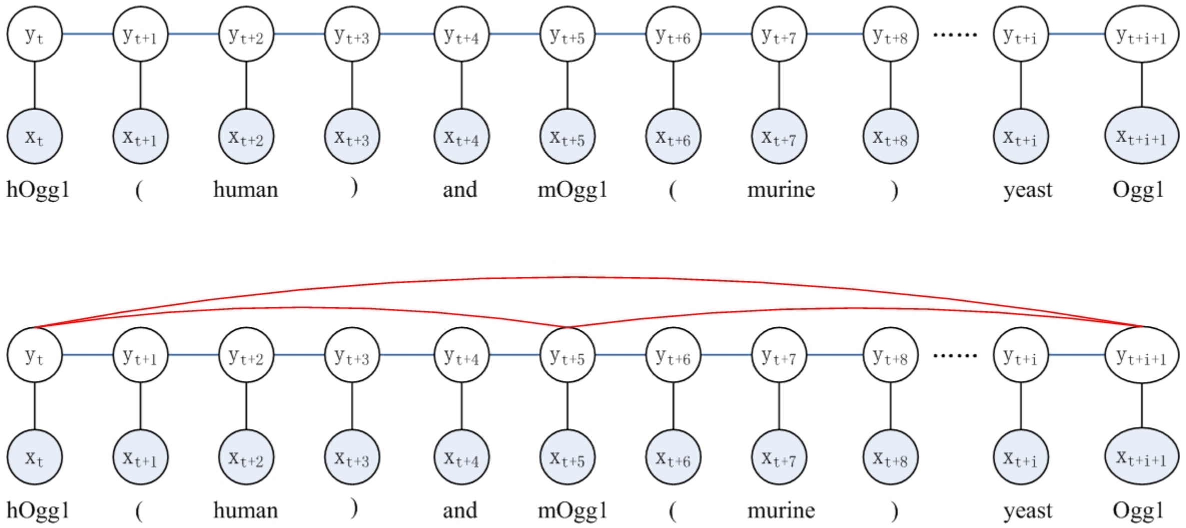

Figure 1: The illustration of linear-chain CRF and skip-chain CRF. The blue edges represent the linear-chain edges belonging to one clique template, while the red edges represent the skip edges belonging to another clique template.

skip edges to avoid making the model impractical. Therefore, it is absolutely necessary to study which kinds of edges will contribute to the performance while avoiding over-connected edges.

3.1 Features

As our interest is in modifying the CRF graph structure rather than evaluating the effectiveness of features, we simply adopted features from the state-of-the-art such as (McDonald and Pereira, 2005) and (Kuo et al., 2007).

- **Common Features**: the original word, the stemmed word, the POS-tag of a word, the word length, is or not the beginning or ending word of the sentence etc.
- **Regular Expression Features**: a set of regular expressions to extract orthographic features for the word.
- **Dictionary Features**: We use several lexicons. For example, a protein name dictionary compiled from SWISS-PROT, a species dictionary from NCBI Taxonomy, a drug name dictionary from DrugBank database, and a disease name dictionary from several Internet web site.
- **N-gram Features**: For each token, we extract the corresponding 2-4 grams into the feature set.

Each word will include the adjacent words' features within $\{-2, -1, 0, 1, 2\}$ offsets. The features used in the linear-chain CRF and skip-chain CRF are all the same in our experiment.

4 Method

As the limitations discussed above, detecting the necessary nodes to link should be the first step in constructing a skip-chain CRF. In the speaker name extraction task (Sutton and McCallum, 2004), only identical capitalized words are linked, because there is few variations in the speaker's name. However, gene mentions often involve words without obvious orthographic features and such phenomena are common in the biomedical literature such as *RGC DNA sequence* and *multisubunit TFIID protein*. If we link all the words like *DNA*, *sequence* and *protein*, the efficiency and performance will drop due to over-connected edges. Therefore, the most important step of detecting gene mentions is to determine which edges should be connected.

4.1 Detect keywords in gene mention

We found that many gene mentions have at least one important word for the identification of gene mentions. For example, the word, *Gal4*, is such a keyword in *Gal4 protein* and *NS1A* in *NS1A protein*. These words can distinguish gene mentions from other common English words and phrases, and can distinguish different gene mentions as well. We define such words as the keyword of a gene mention. The skip edges are limited to only connect these keywords. We use a rule-based method to detect keywords. By examining the annotated data, we defined keywords as those containing at least one capital letter or digit. And at the same time, keywords must conform to the following rules:

- Keywords are not stop words, single letters, numbers, Greek letters, Roman numbers or nucleotide sequence such as *ATTCCCTGG*.
- Keywords are not in the form of an uppercase initial letter followed by lowercase letters, such as *Comparison* and *Watson*. These words have capital letters only because they are the first word in the sentences, or they are the names of people or other objects. This rule will miss some correct candidates, but reduces noise.
- Keywords do not include some common words with capital letters such as *DNA*, *cDNA*, *RNA*, *mRNA*, *tRNA* etc. and some frequently appearing non-gene names such as *HIV* and *mmHg*. We defined a lexicon for such words on the training data.

4.2 Link similar keywords

After keyword candidates are detected, we judge each pair of keywords in the same sentence to find similar word pairs. Each word pair is examined by these rules:

- They are exactly the same words.
- Words only differ in digit letters, such as *CYP1* and *CYP2*.
- Words with the same prefix, such as *IgA* and *IgG*, or with the same suffix, such as *ANF* and *pANF*.

The token pair will be linked by a skip edge if they match at least one rule.

4.3 Link typed dependencies

Some long distance dependency cannot be detected simply by string similarity. To capture such dependency, we used stanford parser[3] to parse sentences and extract typed dependencies from parsed results. The typed dependencies are a set of binary relations belonging to 55 pre-defined types to provide a description of the grammatical relationships in a sentence (Marneffe and Manning, 2008). Some examples of typed dependencies are listed in Table 1.

Type	**Description**
conj	conjuncted by the conjunction such as *and*
prep	prepositional modifier
nn	noun compound modifier
amod	adjectival modifier
dep	uncertain types

Table 1: Examples for typed dependencies.

The output of the parser is pairs of dependent words, along with typed dependencies between two words in a pair. For example, in the sentence:

> *“. . . and activate transcription of a set of genes that includes G1 cyclins CLN1, CLN2, and many DN, synthesis genes.”*

a typed dependency *nn(G1,CLN1)* is extracted by the parser, meaning the words *G1* and *CLN1* has a typed dependency of *nn* because they form a noun phrase under a dependency grammar: modification. Similarly, in the sentence

> *“Using the same approach we have shown that hFIRE binds the stimulatory proteins Sp1 and Sp3 in addition to CBF.”*

the words *Sp1* and *Sp3* can be detected to have a typed dependency of *conj_and*, and the two words have a typed denpendency of *prep_in_addition_to* with *CBF*, respectively. The most common type dependencies are *conj_and*, *nn* and *dep*. The keywords having typed dependencies will be linked by a skip edge.

5 Experiment

We tested our method on two datasets: the Gene Mention (GM) data in BioCreAtIvE II (BCIIGM)

[3] http://nlp.stanford.edu/software/lex-parser.shtml

[4]and GENIA corpus[5]. The BCIIGM dataset was used in the BioCreAtIvE II Gene Mention Recognition task in 2006. It was built from the GENETAG corpus (Tanabe et al., 2005) with some modification of the annotation. The dataset contains 15000 sentences for training and 5000 sentences for testing. Two gold-standard sets, GENE and ALTGENE, were provided for evaluation and an official evaluation procedure in Perl script was provided. The ALTGENE set provides alternate forms for genes in the GENE set. In the official evaluation, each identified string will be looked up in both GENE and ALTGENE. If the corresponding gene was found in either GENE or ALTGENE, the identified string will be counted as a correct answer.

The GENIA corpus is a widely used dataset in many NER and information extraction tasks due to its high quality annotation. The GENIA corpus contains 2000 abstracts from MEDLINE, with approximately 18500 sentences. The corpus was annotated by biomedical experts according to a predefined GENIA ontology. In this work, we only used the annotated entities that have a category of protein, DNA, or RNA. These categories are related to the definition of gene mention in BioCreAtIvE II. We only used strict matching evaluation (no alternate forms check) for the GENIA corpus as no ALTGENE-like annotation is available.

The performance is measured by precision, recall and F score. Each identified string is counted as a true positive (TP) if it is matched by a gold-standard gene mention, otherwise the identified string is a false positive (FP). Each gold standard gene mention is counted as a false negative (FN) if it is not identified by the approach. Then the precision, recall and their harmonic average F score is calculated as follows:

$$precision = \frac{TP}{TP + FP}$$

$$recall = \frac{TP}{TP + FN}$$

$$F = \frac{2 \cdot precision \cdot recall}{precision + recall}$$

To implement both linear-chain CRF and skip-chain CRF, we used the GRMM Java package[6] which is an extended version of MALLET. The package provides an implement of arbitrary structure CRF.

5.1 Result Comparison

We evaluated our approach on the BCIIGM dataset and GENIA corpus. For the BCIIGM dataset, two evaluation criteria were used: *official* - exactly the same as that used in the BioCreAtIvE II competition, with the official evaluation procedure; and *strict* - strict matching for each identified string without checking its alternate forms in ALTGENE. The GENIA dataset were randomly divided into 10 parts to perform a 10-fold cross validation. However, we didn't do cross validation on the BCIIGM dataset because the BioCreAtIvE II competition annotations and evaluation procedure were tailored to evaluating participating systems.

The comparative results are listed in Table 2. We compared the two edge linking principles, linking similar words and linking words having typed dependencies. The F score from the skip-chain CRF is better than that from the linear-chain CRF. Significance tests were performed to check whether these results have significant differences. Paired two-tail t-tests were conducted with respect to the F scores of linear-chain CRF vs. those of the two skip-chain CRFs, respectively. The p-value was 1.989×10^{-7} for the skip-chain CRF linked by similar words vs. linear-chain CRF. The p-value was 3.971×10^{-5} for the skip-chain CRF linked by typed dependencies vs. linear-chain CRF. This shows that the improvement is significant.

Note that we did not compare our results on the BCIIGM dataset to those submitted to the competition. There are two reasons for this: First, our focus is on comparing the skip-chain CRF with the linear-chain CRF. Second, in the competition, most participating systems that used CRF also applied other algorithms, or sophisticated rules for adjusting detected boundaries or refining the recognized results, to achieve competitive performance. By contrast, we did not employ any post-processing rule or algorithm to further improve the performance. In this sense, comparing our results to those has become unfair.

[4] http://sourceforge.net/projects/biocreative/files/

[5] http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/home/wiki.cgi?page=Technical+Term+Annotation

[6] http://mallet.cs.umass.edu/grmm/index.php

Data	Model	Precision(%)	Recall(%)	F score(%)
BCIIGM official	linear-chain CRF	85.16	81.50	83.29
	skip-chain CRF linked by sim-words	86.68	82.75	84.67
	skip-chain CRF linked by typed-dep	86.73	82.36	84.49
BCIIGM strict	linear-chain CRF	74.09	69.49	71.73
	skip-chain CRF linked by sim-words	76.26	71.53	73.82
	skip-chain CRF linked by typed-dep	75.99	70.49	73.14
GENIA	linear-chain CRF	76.77	74.92	75.83
	skip-chain CRF linked by sim-words	78.57	77.12	77.82
	skip-chain CRF linked by typed-dep	78.18	76.87	77.52

Table 2: The result comparison between the linear-chain CRF and skip-chain CRF. *BCIIGM* is the BioCreAtIvE II Gene Mention Recognition dataset. *official* means using the official provided evaluation procedure and *strict* means using strict matching to evaluate the results. *sim-words* means similar words and *typed-dep* means typed dependencies. The results for GENIA are averaged over 10-fold cross validation.

5.2 Discussion

We provided in-depth analysis of our results on the BCIIGM dataset. As one of our motivations for connecting words with skip edges is to enhance the consistency of labeling, we firstly examined whether the proposed approach can provide consistent labeling. Let us start from two typical examples. In the first sentence

> *"The response sequences were localized between -67 and +30 in the simian cytomegalovirus IE94 promoter and upstream of position +9 in the HCMV IE68 promoter."*

the word *IE94* is missed (not labeled) while its similar word *IE68* is labeled correctly by the linear-chain CRF. In the second sentence

> *"It is suggested that biliary secretion of both TBZ and FBZ and their metabolites may contribute to this recycling."*

the word *TBZ* is labeled as a gene mention incorrectly (false positive) while its similar word *FBZ* is not labeled at all (true negative) by the linear-chain CRF. Both sentences are correctly labeled by the skip-chain CRF. Similar improvements are also made by the skip-chain CRF model linked by typed dependencies. To study labeling consistency, we counted the statistics of inconsistency errors, as shown in Table 3. Two kinds of inconsistency errors were counted: false negatives correctable by consistency (FNCC) and false positives correctable by consistency (FPCC). An FNCC means that a gold-standard mention is missed by the system while its skip edge linked gene mention is correctly labeled, which is similar to the *inconsistent miss* in (Sutton and McCallum, 2004), as the *IE94* in the first example. An FPCC means a non-gene mention is labeled as a gene while its skip edge linked mention (also non-gene mention) is not recognized, as *TBZ* in the second example. These two kinds of inconsistency errors lead to inconsistent false negatives (FN) and false positives (FP). A good model should reduce as much inconsistency errors as possible. The inconsistency errors are reduced substantially as we expected, showing that the reduction of inconsistency errors is one reason for the performance improvements.

The skip-chain CRF linked by similar words had better performance than the skip-chain CRF linked by typed dependencies. This may infer that the quality of skip edges has impact on the performance. In order to study this issue, the quality of skip edges was examined. The statistics of skip edges in the BCIIGM dataset for the two skip-chain CRF models (linked by similar words and by typed dependencies respectively) is shown in the first two rows of Table 4. A skip edge is counted as a correct edge if the edge links two words that are both gene mentions in the gold-standard annotation. The statistics shows that the skip-chain CRF linked by similar words has a higher precision than the model by typed dependencies. To make the comparison more evident, we built another skip-chain CRF whose skip edges were randomly connected. The number of skip edges in this model

Skip edge type	**Model**	**FPCC**	**FNCC**
sim-words	linear-chain	112	70
	skip-chain	48	20
Percentage of reduction		57.14%	71.43%
typed-dep	linear-chain	32	29
	skip-chain	9	5
Percentage of reduction		71.88%	82.76%

Table 3: Statistics of inconsistency errors for the linear-chain CRF and skip-chain CRF. *FPCC* is false positives correctable by consistency and *FNCC* is false negatives correctable by consistency in the table. The percentage is calculated by dividing the reduction of errors by the error number of linear-chain CRF, for example $(112 - 48)/48 = 57.14\%$.

approximately equals to that in the skip-chain CRF linked by similar words. The percentage of correct skip-edges in this model is small, as shown in the last row of Table 4. We tested this skip-chain CRF model on the BCIIGM dataset under the strict matching criterion. The performance of the randomly linked skip-chain CRF is shown in Table 5. As can be seen from the table, the performance of the randomly connected skip-chain CRF droped remarkably, even worse than that of the linear-chain CRF. This confirms that the quality of skip edges is a key factor for the performance improvement.

Model	**Edges**	**Correct edges**	**Percentage**
sim-words	1912	1344	70.29%
typed-dep	728	425	53.38%
random	1906	41	2.15%

Table 4: Statistics of skip edges and correct skip edges for the skip-chain CRF models. *sim-words* means the skip-chain CRF linked by similar words, *typed-dep* means the CRF linked by typed dependencies and *random* means the skip-chain CRF has randomly connected skip edges. The edges are counted in the BCIIGM testing data.

From the above discussion, we summarize this section as follows: (1) the skip-chain CRF with high quality skip edges can reduce inconsistent labeling errors, and (2) the quality of skip edges is crucial to the performance improvement.

Model	**P (%)**	**R(%)**	**F(%)**
linear	74.09	69.49	71.73
sim-words	76.26	71.53	73.82
typed-dep	75.99	70.49	73.14
random	73.66	69.13	71.32

Table 5: Performance comparison between the randomly linked skip-chain CRF and other models. The result was tested on the BCIIGM dataset under the strict matching criterion. *P*, *R* and *F* denote the precision, recall and F score respectively. *linear* denotes the linear-chain CRF. *sim-words* denotes the skip-chain CRF linked by similar words. *typed-dep* denotes the skip-chain CRF linked by typed dependencies. *random* denotes the skip-chain CRF having randomly linked skip edges.

6 Conclusion

This paper proposed a method to construct a skip-chain CRF to perform named entity recognition in the biomedical literature. We presented two principles to connect skip edges to address the issue of capturing long distance dependency: linking similar keywords and linking words having typed dependencies. We evaluated our method on the BioCreAtIvE II GM dataset and GENIA corpus. Significant improvements were observed. Moreover, we presented in-depth analysis on inconsistent labeling errors and the quality of skip edges. The study shows that the quality of linked edges is a key factor of the system performance.

The quality of linked edges plays an important role in not only performance but also time efficiency. Thus, we are planning to apply machine learning techniques to automatically induce patterns for linking high-quality skip-edges. Furthermore, to refine the recognition results, we are planning to employ post-processing algorithms or construct refinement rules.

Acknowledgments

This work was partly supported by the Chinese Natural Science Foundation under grant No. 60803075 and No.60973104, and partly carried out with the aid of a grant from the International Development Research Center, Ottawa, Canada IRCI project from the International Development.

References

Shilin Ding, Gao Cong, Chin-Yew Lin and Xiaoyan Zhu. 2008. *Using Conditional Random Fields to Extract Contexts and Answers of Questions from Online Forums*. In Proceedings of 46th Annual Meeting of the Association for Computational Linguistics (ACL'08), pp 710-718.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. *Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling*. Proceedings of the 43rd Annual Meeting of the ACL, pages 363C370.

K. Fukuda, A. Tamura, T. Tsunoda and T. Takagi. 1998. *Toward information extraction: identifying protein names from biological papers*. Pacific Symposium on Biocomputing. 1998.

Michel Galley. 2006. *A Skip-Chain Conditional Random Field for Ranking Meeting Utterances by Importance*. Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing (EMNLP 2006), pages 364-372.

Han-Shen Huang, Yu-Shi Lin, Kuan-Ting Lin, Cheng-Ju Kuo, Yu-Ming Chang, Bo-Hou Yang, I-Fang Chung and Chun-Nan Hsu. 2007. *High-recall gene mention recognition by unification of multiple backward parsing models*. Proceedings of the Second BioCreative Challenge Evaluation Workshop, pages 109-111.

Jun'ichi Kazama, Takaki Makino, Yoshihiro Ohta and Jun'ichi Tsujii. 2002. *Tuning support vector machines for biomedical named entity recognition*. Proceedings of the ACL-02 workshop on Natural language processing in the biomedical domain - Volume 3.

Corinna Kolarik, Martin Hofmann-Apitius, Marc Zimmermann and Juliane Fluck. 2007. *Identification of new drug classification terms in textual resources*. Bioinformatics 2007 23(13):i264-i272

Cheng-Ju Kuo, Yu-Ming Chang, Han-Shen Huang, Kuan-Ting Lin, Bo-Hou Yang, Yu-Shi Lin, Chun-Nan Hsu and I-Fang Chung. 2007. *Rich Feature Set, Unification of Bidirectional Parsing and Dictionary Filtering for High F-Score Gene Mention Tagging*. Proceedings of the Second BioCreative Challenge Evaluation Workshop, pages 105-107.

Ki-Joong Lee, Young-Sook Hwang, Seonho Kim and Hae-Chang Rim. 2004. *Biomedical named entity recognition using two-phase model based on SVMs*. Journal of Biomedical Informatics, Volume 37, Issue 6, December 2004, Pages 436-447.

John Lafferty, Andrew McCallum and Fernando Pereira. 2001. *Conditional random fields: Probabilistic models for segmenting and labeling sequence data*. In Proc. ICML-01, pages 282-289, 2001.

Ryan McDonald and Fernando Pereira. 2005. *Identifying gene and protein mentions in text using conditional random fields*. BMC Bioinformatics 2005, 6(Suppl 1):S6.

Marie-Catherine de Marneffe and Christopher D. Manning. 2008. *Stanford typed dependencies manual*.

M. Narayanaswamy, K.E. Ravikumar and K. Vijay-Shanker. 2003. *A biological named entity recognizer*. Pacific Symposium on Biocomputing. 2003.

Dietrich Rebholz-Schuhmann, Harald Kirsch, Miguel Arregui, Sylvain Gaudan, Mark Riethoven and Peter Stoehr. 2007. *EBIMedltext crunching to gather facts for proteins from Medline*. Bioinformatics 2007 23(2):e237-e244

Fei Sha and Fernando Pereira. 2003. *Shallow Parsing with Conditional Random Fields*. Proceedings of HLT-NAACL 2003, Main Papers, pp.134-141

Larry Smith, Lorraine K Tanabe, et al. 2008. *Overview of BioCreative II gene mention recognition*. Genome Biology 2008, 9(Suppl 2):S2.

Charles Sutton and Andrew McCallum. 2004. *Collective Segmentation and Labeling of Distant Entities in Information Extraction*. ICML workshop on Statistical Relational Learning, 2004.

Lorraine Tanabe, Natalie Xie, Lynne H Thom, Wayne Matten and W John Wilbur. 2005. *GENETAG: a tagged corpus for gene/protein named entity recognition* . BMC Bioinformatics 2005, 6(Suppl 1):S3

Serhan Tatar and Ilyas Cicekli. 2009. *Two learning approaches for protein name extraction*. Journal of Biomedical Informatics 42(2009) 1046-1055

Xinglong Wang, Jun'ichi Tsujii and Sophia Ananiadou. 2010. *Disambiguating the species of biomedical named entities using natural language parsers*. Bioinformatics 2010 26(5):661-667

Zhihao Yang, Hongfei Lin and Yanpeng Li. 2008. *Exploiting the performance of dictionary-based bioentity name recognition in biomedical literature*. Computational Biology and Chemistry 32(2008) 287-291.

Alexander Yeh, Alexander Morgan, Marc Colosimo and Lynette Hirschman. 2005. *BioCreAtIvE Task 1A: gene mention finding evaluation*. BMC Bioinformatics 2005, 6(Suppl 1):S2.

GuoDong Zhou, Jie Zhang, Jian Su, Dan Shen, ChewLim Tan. 2004. *Recognizing names in biomedical texts: a machine learning approach*. Bioinformatics 2004, Vol.20(7),pp.1178C1190.

GuoDong Zhou, Dan Shen, Jie Zhang, Jian Su1 and SoonHeng Tan. 2005. *Recognition of protein/gene names from text using an ensemble of classifiers*. BMC Bioinformatics 2005, 6(Suppl 1):S7.