Structure Regularized Learning for Named Entity Recognition

Yang Yang, Xu Sun
MOE Key Laboratory of Computational Linguistics, Peking University

Abstract

We describe a simple approach to named entity recognition (NER) using structure regularized learning based on conditional random fields. The structures of models in NER tasks are often complex, which are actually harmful in structured prediction (structured classification) according to our study, because they can easily cause structure-based overfitting. In order to control this kind of structure-based overfitting, we decompose training samples into mini-samples which are simpler on structures and use a structure regularized learning scheme. Our method achieves record-breaking scores on those tasks and also with substantially faster training speed.

1 Introduction

Named Entity Recognition (NER) is a task that identifies and classifies named entities, including the names of persons, organizations, locations, products, companies, etc. within running text into categories automatically. NER starts as a subproblem of information extraction. Now, it becomes an essential task for question answering, information retrieval , coreference resolution and various other NLP problems.

NER can be considered as a sequence labeling problem where each token in a sentence is annotated with BIO-tags. In addition, the names are annotated with a category. Named entities can be any type of word: nouns, prepositions, adverbs, adjectives, and even some verbs, but the majority are nouns.

Many different approaches can be applied to named entity recognition. Some of representative models are conditional random fields (CRFs), deep neural networks, and structured perceptron models. There are many rare named entities and large amounts of ambiguities in natural languages. So, recent years, in order to more accurately capture structural information, some studies introduce long range dependencies and develop long distance features or global features for the purpose of intensifying structural dependencies in structured prediction.

However, these complex structures in NER tasks are actually harmful to model accuracy according to our study. So, it could be misleading to over-emphasize on intensive structural dependencies (Sun, 2014). Indeed, while it is obvious that intensive structural dependencies can effectively incorporate structural information, it is less obvious that intensive structural dependencies have a drawback of increasing the generalization risk. The trained model tends to overfit the training data if increasing the complexity of structure,because a more complex structure is easier to suffer from overfitting. It can be more serious for NER tasks where complex features are widely used.

To deal with this problem, we employ a simple solution based on structure regularized learning (Sun, 2014) to derive a model with better generalization power. Each training sample is decomposed into multiple mini-samples. The structure of mini-samples is much simpler than the original one. Through experiments, we find that the method improves both accuracy and training time in three named entity recognition tasks.

To our knowledge, this is the first application of the idea of structure regularization (Sun, 2014) in general NER tasks. Our method achieves record-breaking scores on the standard NER tasks in different languages, including Dutch, Spanish, and English, with the error rate reductions of 4.97%, 4.25% and 0.43% on Dutch, Spanish and English standard datasets, respectively. In addition to the significant improvement on accuracy scores, our method also with substantially faster training speed than existing methods.

2 Background

In this section, we will briefly review the named entity recognition problem and some techniques such as conditional random fields which we apply in our work.

2.1 Named Entity Recognition

The early NER systems such as (Fisher et al., 1995; Black et al., 1998) using linguistic tools could be difficult to develop or adapt to other languages. So, a variety of machine learning methods have been applied to NER tasks, such as hidden Markov models (Bikel et al., 1999), maximum entropy models (Malouf, 2002), support vector machines(McNamee and Mayfield, 2002; Hearst et al., 1998), averaged perceptrons (Buitinck and Marx, 2012), various connectionist approaches (Florian, 2002; Hammerton, 2003) or a combination of various classifiers (CRF, SVM, k-NN) (Desmet and Hoste, 2010).

On English biomedical named entity recognition task in the BioNLP-2004 shared task, (Tsuruoka et al., 2011) proposes a method based on lookahead learning and (Yoshida and Tsujii, 2007) based on reranking. (Lin et al., 2004) uses a maximum entropy approach and (Settles, 2004) uses condition random fields in their approaches. On the Dutch and Spanish named entity recognition tasks in the CoNLL-2002 shared task, (Carreras et al., 2002) uses binary AdaBoost and (Wu et al., 2002) uses boosting in their approaches for both tasks. (Buitinck and Marx, 2012) uses averaged perceptrons on Dutch NER in his 2012 approach. On the Spanish NER task, Kozareva proposes a bootstrapping method in (Kozareva, 2006). He also introduces a 'voted co-training' algorithm in (Kozareva et al., 2005),

Performance of NER can be further improved by using gazetteers (Mikheev et al., 1999): lists of persons, locations and organizations. However, we do not refer to any gazetteers in our work, because our work makes no language-specific assumptions. Moreover, there may be no good gazetteers for some languages.

2.2 Conditional Random Fields

The conditional random fields (CRFs), first proposed in (Lafferty et al., 2001), are discriminative probabilistic graphical models aimed at calculating the conditional probability of designated output labels given input observations for a sequence of tokens.

Let $\boldsymbol{O} = (\boldsymbol{o}_{(1)}, \boldsymbol{o}_{(2)}, \ldots, \boldsymbol{o}_{(n)})$ be a sequence of running text words and $\boldsymbol{y} = (\boldsymbol{y}_{(1)}, \boldsymbol{y}_{(2)}, \ldots, \boldsymbol{y}_{(n)})$ be corresponding output labels. We assume they have the same length n. Moreover, a sample is converted to an indexed sequence of feature vectors $\boldsymbol{x} = \{\boldsymbol{x}_{(1)}, \ldots, \boldsymbol{x}_{(n)}\}$. We can use an $n \times d$ matrix to represent $\boldsymbol{x} \in \mathcal{X}^n$.

We use $\boldsymbol{f}$ to represent the global feature vector, and $\boldsymbol{w}$ a vector of learned weights. Such CRFs define conditional probability distributions $P(\boldsymbol{y}|\boldsymbol{x})$ of label sequences given input feature sequences.

$$P(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{w}) = \frac{\exp(\boldsymbol{w}^\top \boldsymbol{f}(\boldsymbol{y}, \boldsymbol{x}))}{\sum_{\boldsymbol{y}'} \exp(\boldsymbol{w}^\top \boldsymbol{f}(\boldsymbol{y}', \boldsymbol{x}))} \quad (1)$$

Let $\mathcal{Z} = (\mathcal{X}^n, \mathcal{Y}^n)$ and $\boldsymbol{z} = (\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{Z}$ denote a sample in the training data. We train the weights of a CRF by setting them to maximize the conditional log-likelihood of labeled sequences in some given training set $S = \{\boldsymbol{z}_1 = (\boldsymbol{x}_1, \boldsymbol{y}_1), \ldots, \boldsymbol{z}_m = (\boldsymbol{x}_m, \boldsymbol{y}_m)\}$. Applying an L_2 prior, the conditional log-likelihood is

$$L(\boldsymbol{w}) = \sum_{i=1}^{m} \log(P(\boldsymbol{y}_i|\boldsymbol{x}_i, \boldsymbol{w})) - \frac{||\boldsymbol{w}||^2}{2\sigma^2} \quad (2)$$

3 Structure Regularized Learning

Following prior work on learning with decomposition (Sun, 2014; Sutton and McCallum, 2007; Samdani and Roth, 2012; Tsuruoka et al., 2011), we apply the structure regularization method (Sun, 2014) which regularizes the complexity of structures. This method can reduce the overfitting risk in the named entity recognition problem with structured prediction problems.

3.1 Structured Learning

Following (Sun, 2014), we use the term *sample* to denote $\boldsymbol{O} = \{\boldsymbol{o}_{(1)}, \ldots, \boldsymbol{o}_{(n)}\}$. So in named entity recognition tasks, a sample corresponds to a sentence of n words with dependencies. Thus, we call n as *tag structure complexity* or simply *structure complexity* below. A learning algorithm is a function $G : \mathcal{Z}^m \mapsto \mathcal{F}$ with the function space $\mathcal{F} \subset \{\mathcal{X}^n \mapsto \mathcal{Y}^n\}$, i.e., G maps a training set S to a function $G_S : \mathcal{X}^n \mapsto \mathcal{Y}^n$.

We define *point-wise cost function* $c : \mathcal{Y} \times \mathcal{Y} \mapsto \mathbb{R}^+$ as $c[G_S(\boldsymbol{x}, k), \boldsymbol{y}_{(k)}]$, which measures the cost on a position k by comparing $G_S(\boldsymbol{x}, k)$ and the

gold-standard tag $\boldsymbol{y}_{(k)}$, and we introduce the pointwise loss as

$$\ell(G_S, \boldsymbol{z}, k) \triangleq c[G_S(\boldsymbol{x}, k), \boldsymbol{y}_{(k)}] \quad (3)$$

Then, we define *sample-wise cost function* $C : \mathcal{Y}^n \times \mathcal{Y}^n \mapsto \mathbb{R}^+$, which is the cost function with respect to a whole sample, and we introduce the sample-wise loss as

$$\mathcal{L}(G_S, \boldsymbol{z}) \triangleq C[G_S(\boldsymbol{x}), \boldsymbol{y}] = \sum_{k=1}^{n} \ell(G_S, \boldsymbol{z}, k) \quad (4)$$

Given G and a training set S, what we are most interested in is the *generalization risk* in structured prediction (i.e., expected average loss) (Taskar et al., 2003; London et al., 2013):

$$R(G_S) = \mathbb{E}_{\boldsymbol{z}}\Big[\frac{\mathcal{L}(G_S, \boldsymbol{z})}{n}\Big] \quad (5)$$

Since the distribution D is unknown, we have to estimate $R(G_S)$ by using the *empirical risk*:

$$\begin{aligned} R_e(G_S) &= \frac{1}{mn}\sum_{i=1}^{m} \mathcal{L}(G_S, \boldsymbol{z}_i) \\ &= \frac{1}{mn}\sum_{i=1}^{m}\sum_{k=1}^{n} \ell(G_S, \boldsymbol{z}_i, k) \end{aligned} \quad (6)$$

Most existing regularization techniques which process named entity recognition regularize model weights/parameters (e.g., L_2 regularizer). We call such regularization techniques as *weight regularization*. Let $N_\lambda : \mathcal{F} \mapsto \mathbb{R}^+$ be a weight regularization function on $\mathcal{F}$ with regularization strength λ, the structured classification based objective function with general weight regularization is as follows:

$$R_\lambda(G_S) \triangleq R_e(G_S) + N_\lambda(G_S) \quad (7)$$

3.2 Structure Regularized Learning

Different from weight regularization which normalizes model weights, the structure regularized learning method normalizes the structural complexity of the training samples (Sun, 2014). As illustrated in Figure 1, our proposal is based on *tag structure decomposition*. We decompose a complete sentence, from which the named entities are to be recognized, into mini-samples.The structured classification based objective function with structure regularized learning is formally defined as follows[1]:

$$\begin{aligned} R_\alpha(G_S) &\triangleq R_e[G_{N_\alpha(S)}] \\ &= \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{\alpha}\sum_{k=1}^{n/\alpha} \ell[G_{S'}, \boldsymbol{z}_{(i,j)}, k] \end{aligned} \quad (8)$$

where $N_\alpha(\boldsymbol{z}_i)$ is a structure regularized learning function and it randomly splits $\boldsymbol{z}_i$ into α mini-samples $\{\boldsymbol{z}_{(i,1)}, \ldots, \boldsymbol{z}_{(i,\alpha)}\}$, so that the mini-samples have a distribution on their sizes (structure complexities) with the expected value $n' = n/\alpha$. Thus, we get

$$\begin{aligned} S' = \quad & \{\underbrace{\boldsymbol{z}_{(1,1)}, \boldsymbol{z}_{(1,2)}, \ldots, \boldsymbol{z}_{(1,\alpha)}}_{\alpha}, \\ & \ldots, \\ & \underbrace{\boldsymbol{z}_{(m,1)}, \boldsymbol{z}_{(m,2)}, \ldots, \boldsymbol{z}_{(m,\alpha)}}_{\alpha}\} \end{aligned} \quad (9)$$

with $m\alpha$ mini-samples with expected structure complexity n/α. We can denote S' more compactly as $S' = \{\boldsymbol{z}'_1, \boldsymbol{z}'_2, \ldots, \boldsymbol{z}'_{m\alpha}\}$ and $R_\alpha(G_S)$ can be simplified as

$$R_\alpha(G_S) = \frac{1}{mn}\sum_{i=1}^{m\alpha}\sum_{k=1}^{n/\alpha} \ell[G_{S'}, \boldsymbol{z}'_i, k] \quad (10)$$

When the structure regularized learning strength $\alpha = 1$, we have $S' = S$ and $R_\alpha = R_e$, which means the sentence to be processed is not decomposed. The structure regularized learning algorithm (with the stochastic gradient descent setting) is summarized in Algorithm 1. Recall that $\boldsymbol{x} = \{\boldsymbol{x}_{(1)}, \ldots, \boldsymbol{x}_{(n)}\}$ represents feature vectors. Thus, it should be emphasized that the decomposition of $\boldsymbol{x}$ is the decomposition of the feature vectors, not the original observed words. Actually the decomposition of the feature vectors is more convenient and has no information loss — decomposing observations needs to regenerate features and may lose the features which have information from different mini-samples. These features can be important for named entity recognition.

In processing named entity recognition, the structure regularized learning has no conflict with the weight regularization, and in our work, we use the structure regularized learning together with the weight regularization.

[1]The notation N is overloaded here. For clarity throughout, N with subscript λ refers to weight regularization function, and N with subscript α refers to structure regularized learning function.

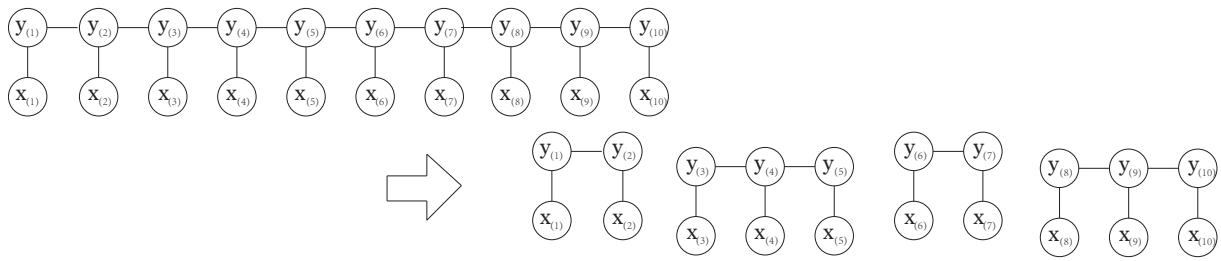

Figure 1: An illustration of structure regularized learning in simple linear chain case, which randomly decompose a training sample $\boldsymbol{z}$ with structure complexity 10 into three mini-samples with structure complexity 2 or 3. The complexities have a distribution with the expected value 2.5. In the figure, $\boldsymbol{x}$ represents extracted features of the training sample and $\boldsymbol{y}$ represents the corresponding tags.

Algorithm 1 Learning with structure regularized learning (an illustration based on SGD training)

Input: model weights $\boldsymbol{w}$, training set S, structure regularized learning strength α
repeat
  $S' \leftarrow \emptyset$
  for $i = 1 \rightarrow m$ **do**
    Randomly decompose $\boldsymbol{z}_i \in S$ into mini-samples $N_\alpha(\boldsymbol{z}_i) = \{\boldsymbol{z}_{(i,1)}, \ldots, \boldsymbol{z}_{(i,\alpha)}\}$
    $S' \leftarrow S' \cup N_\alpha(\boldsymbol{z}_i)$
  end for
  SGD_Train_oneIter(S')
until Convergence
return $\boldsymbol{w}$

By combining structure regularized learning and weight regularization, the structured classification based objective function is as follows:

$$R_{\alpha,\lambda}(G_S) \triangleq R_\alpha(G_S) + N_\lambda(G_S) \qquad (11)$$

When $\alpha = 1$, we have $R_{\alpha,\lambda} = R_e(G_S) + N_\lambda(G_S) = R_\lambda$.

4 Experiments

4.1 Tasks

Dutch Named Entity Recognition (Dutch-NER) We use the dataset from the shared task of CoNLL-2002. The dataset concentrates on four types of named entities: persons, locations, organizations and names of miscellaneous entities that do not belong to the previous three groups. The corpus consists of four editions of the Belgian newspaper 'De Morgen' of 2000 (June 2, July 1, August 1 and September 1). The data was annotated as a part of the Atranos project at the University of Antwerp. There are altogether 271,925 tokens, including 17,285 entities in training and test data.

Spanish Named Entity Recognition (Spanish-NER) Spanish-NER is based on CoNLL-2002 dataset. The four types entities to be recognized in this task is the same as the types in Dutch-NER. The data is a collection of news wire articles made available by the Spanish EFE News Agency consisting of 316,248 tokens, including 22,355 entities.

English Biomedical Named Entity Recognition (Bio-NER) This task is from the BioNLP-2004 shared task, which is for recognizing 5 kinds of biomedical named entities, namely, DNA, RNA, protein, cell-type and cell-line. This task is on the MEDLINE biomedical text corpus. There are 564,646 tokens, including 56,988 entities.

The evaluation metric of all tasks is balanced F-score: $F_1 = \frac{2PR}{P+R}$, where P means precision, R means recall.

Dutch-NER	LOC	ORG	PER	MISC
Train	3,208	2,082	4,716	3,338
Test	774	882	1,098	1,187
Spanish-NER	LOC	ORG	PER	MISC
Train	4,913	7,390	4,321	2,173
Test	1,084	1,400	735	339

Bio-NER	Protein	C-line	C-type	RNA	DNA
Train	28,505	3,590	6,382	887	8,962
Test	5,067	500	1,921	118	1,056

Table 1: Entity information of the datasets.

4.2 Feature Set

The context of a word *w* is called a window of *w*. The size of a window can be determined by the furthermost word selected in the window. Together with the relative positive to *w*, a feature is represented.

For each word in the sequence, we apply following features for Dutch-NER and Spanish-NER tasks:

- The word form, and its part-of-speech(POS) tag, within w_{i-2}, w_{i-1}, w_i, that is to say, from the word with relative position of -2 to the current word. POS tag features are only available for Dutch-NER.
- Prefixes and suffixes of the current word and the previous word up to a length of 4 characters.
- Word patterns in w_{i-2}, w_{i-1}, w_i. Whether each character in the word is a lower case letter, a upper case letter or a non-letter character.
- Orthographic features of the current word:
  - Whether or not the initial character is a capital letter. e.g. , England, Germany
  - Whether or not the word has only one character. e.g. , a, y
  - Whether or not all characters of the word are capital letters. e.g. , UNICEF, IFAW
  - Whether or not the word contains a digital.
  - Whether or not all characters of the word are digitals. e.g. , 2005 in phrase 'the year of 2005'
  - Whether or not the word contains a dot.
  - Whether or not the word contains a hyphen. e.g. , on-line
  - Whether or not the word is a punctuation.

For the Bio-NER task, we apply following features:

- The word form, in $\{w_{i-2}, w_{i-1}, w_i, w_{i+1}, w_{i+2}, w_{i-1}w_i, w_iw_{i+1}\}$.
- POS tag features, in $\{w_{i-1}, w_i, w_{i+1}, w_{i-1}w_i, w_iw_{i+1}, w_{i-2}w_{i-1}w_i, w_{i-1}w_iw_{i+1}, w_iw_{i+1}w_{i+2}\}$
- Chunking features, in $\{w_{i-1}, w_i, w_{i+1}, w_{i-1}w_i, w_iw_{i+1}, w_{i-2}w_{i-1}w_i, w_iw_{i+1}w_{i+2}\}$
- Word patterns of words in $\{w_{i-1}, w_i, w_{i+1}\}$.
- Prefixes and suffixes up to a length of 4 characters in $\{w_{i-1}, w_i, w_{i+1}\}$.

We apply in total 349,073 raw features for Dutch-NER, 387,130 raw features for Spanish-NER and 403,192 raw features for Bio-NER.

4.3 Experimental Setting

For the purpose of testing the robustness of the structure regularized learning (SR) method, we perform our experiments on different models including both probabilistic and non-probabilistic structure prediction models . We choose the conditional random fields (CRFs) (Lafferty et al., 2001) and structured perceptrons (Perc) (Collins, 2002), which are arguably the most popular probabilistic and non-probabilistic structured prediction models, respectively. The CRFs are trained using the SGD (Huang et al., 2011) and BFGS (Wright and Nocedal, 1999) algorithms. For the structured perceptrons, we choose naïve perceptron method (Freund and Schapire, 1999; Collins, 2002) and averaged perceptron method (Collins, 2002). They have very fast training speed due to the avoidance of the computation on gradients (Sun et al., 2013). The rich edge features (Sun et al., 2014; Sun et al., 2012) are employed for all methods.

For SGD, we perform automatic tuning for the L_2 regularization strengths based on the training data via 4-fold cross-validation, testing with 0.005, 0.01, 0.05, 0.1, respectively, and the optimal value is chosen based on the best F-score of cross-validation. Via this automatic tuning, we find it is proper to set 0.05 for Dutch-NER and Spanish-NER and 0.005 for Bio-NER. Our SR method adopts the same L_2 regularization setting as SGD. Experiments are performed on a computer with an Intel(R) Xeon(R) 2.0-GHz CPU.

4.4 Results

The experimental results are shown in Figure 2. Results in terms of F-score are shown in the first row and the third row. The first row compares SR with probabilistic models, and the third row compares SR with non-probabilistic models. For CRF models, the training is convergent, the results on the 100'th iteration where the convergence state (decided by relative objective change with

	Number of sentences		Number of Tokens		Averaged Length of Sentences	
	Test	Train	Test	Train	Test	Train
Dutch-NER	5,195	15,806	68,994	202,931	13.28	12.84
Spanish-NER	1,517	8,323	51,533	264,715	33.97	31.81
Bio-NER	3,856	17,484	101,039	463,607	26.20	26.52

Table 2: Information of sentences and tokens about the datasets.

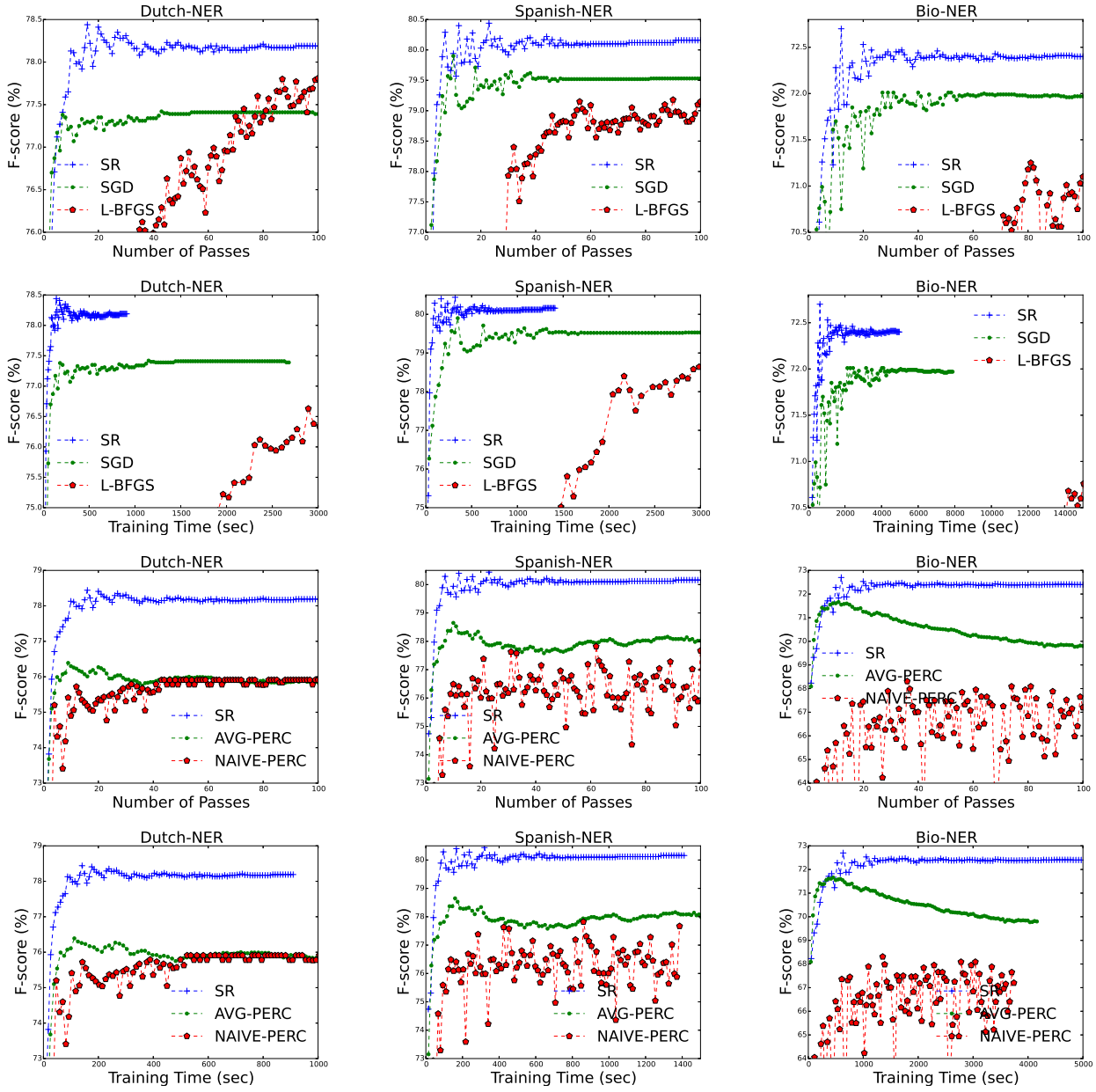

Figure 2: The first two rows compare SR with probabilistic training methods(SGD and L-BFGS). The first row shows F-score curves on the three tasks based on training passes while the second row based on wall-clock time. The third and the fourth row compare SR with non-probabilistic training methods(naïve perceptrons and averaged perceptrons). The third row shows F-score curves on the three tasks based on training passes while the fourth row based on wall-clock time.

the threshold value of 0.0001) is already reached are shown. For perceptron models, the training is typically not convergent, and the results on the 20'th iteration are shown. For stability of the curves, the results of the structured perceptrons are averaged over 10 repeated runs.

The second row and the fourth row of Figure 2 show the experimental results of these different methods in terms of wall-clock training time. As we can see, the training speed of structure regularized learning is substantially improved. Theoretically, mini-samples get a faster convergence rates and the faster processing time on the structures, because it is more efficient to process the decomposed samples. The smaller the size of mini-samples are, the faster the processing is.

5 Conclusions

We describe a simple approach to NER using structure regularized learning. In order to control the structure-based overfitting, we decompose training samples into mini-samples which are simpler on structures and use a structure regularized learning scheme. Our method achieves record-breaking scores on those tasks and also with substantially faster training speed.

References

Daniel M Bikel, Richard Schwartz, and Ralph M Weischedel. 1999. An algorithm that learns what's in a name. *Machine learning*, 34(1-3):211–231.

William J. Black, Fabio Rinaldi, and David Mowatt. 1998. Facile: Description of the ne system used for muc-7. In *Proceedings of the 7th Message Understanding Conference*.

Lars Buitinck and Maarten Marx. 2012. Two-stage named-entity recognition using averaged perceptrons. In Gosse Bouma, Ashwin Ittoo, Elisabeth Métais, and Hans Wortmann, editors, *NLDB*, volume 7337 of *Lecture Notes in Computer Science*, pages 171–176. Springer.

Xavier Carreras, Lluís Marquez, and Lluís Padró. 2002. Named entity extraction using adaboost. In Dan Roth and Antal van den Bosch, editors, *CoNLL*. ACL.

Michael Collins. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 1–8. Association for Computational Linguistics.

Bart Desmet and Véronique Hoste. 2010. Dutch named entity recognition using classifier ensembles. *LOT Occasional Series*, 16:29–41.

David Fisher, Stephen Soderland, Fangfang Feng, and Wendy Lehnert. 1995. Description of the umass system as used for muc-6. In *Proceedings of the 6th conference on Message understanding*, pages 127–140. Association for Computational Linguistics.

Radu Florian. 2002. Named entity recognition as a house of cards: Classifier stacking. In Dan Roth and Antal van den Bosch, editors, *CoNLL*. ACL.

Yoav Freund and Robert E Schapire. 1999. Large margin classification using the perceptron algorithm. *Machine learning*, 37(3):277–296.

James Hammerton. 2003. Named entity recognition with long short-term memory. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 172–175. Association for Computational Linguistics.

Marti A. Hearst, Susan T Dumais, Edgar Osman, John Platt, and Bernhard Scholkopf. 1998. Support vector machines. *Intelligent Systems and their Applications, IEEE*, 13(4):18–28.

Junzhou Huang, Tong Zhang, and Dimitris Metaxas. 2011. Learning with structured sparsity. *The Journal of Machine Learning Research*, 12:3371–3412.

Zornitsa Kozareva, Boyan Bonev, and Andres Montoyo. 2005. Self-training and co-training applied to spanish named entity recognition. In *MICAI 2005: Advances in Artificial Intelligence*, pages 770–779. Springer.

Zornitsa Kozareva. 2006. Bootstrapping named entity recognition with automatically generated gazetteer lists. In *Proceedings of the eleventh conference of the European chapter of the association for computational linguistics: student research workshop*, pages 15–21. Association for Computational Linguistics.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML '01: Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Yi-Feng Lin, Tzong-Han Tsai, Wen-Chi Chou, Kuen-Pin Wu, Ting-Yi Sung, and Wen-Lian Hsu. 2004. A maximum entropy approach to biomedical named entity recognition. In *BIOKDD*, pages 56–61. Citeseer.

B. London, B. Huang, B. Taskar, and L. Getoor. 2013. Pac-bayes generalization bounds for randomized structured prediction. In *NIPS Workshop on Perturbation, Optimization and Statistics*.

Shuming Ma and Xu Sun. 2017. A generic online parallel learning framework for large margin models. *CoRR*, abs/1703.00786.

Robert Malouf. 2002. A comparison of algorithms for maximum entropy parameter estimation. In *proceedings of the 6th conference on Natural language learning-Volume 20*, pages 1–7. Association for Computational Linguistics.

Paul McNamee and James Mayfield. 2002. Entity extraction without language-specific resources. In *proceedings of the 6th conference on Natural language learning-Volume 20*, pages 1–4. Association for Computational Linguistics.

Andrei Mikheev, Marc Moens, and Claire Grover. 1999. Named entity recognition without gazetteers. In *Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics*, pages 1–8. Association for Computational Linguistics.

Rajhans Samdani and Dan Roth. 2012. Efficient decomposed learning for structured prediction. In *ICML'12*.

Burr Settles. 2004. Biomedical named entity recognition using conditional random fields and rich feature sets. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications*, pages 104–107. Association for Computational Linguistics.

Xu Sun and Jun'ichi Tsujii. 2009. Sequential labeling with latent variables: An exact inference algorithm and its efficient approximation. In *EACL 2009, 12th Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference, Athens, Greece, March 30 - April 3, 2009*, pages 772–780.

Xu Sun, Takuya Matsuzaki, Daisuke Okanohara, and Jun'ichi Tsujii. 2009. Latent variable perceptron algorithm for structured classification. In *IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, July 11-17, 2009*, pages 1236–1242.

Xu Sun, Houfeng Wang, and Wenjie Li. 2012. Fast online training with frequency-adaptive learning rates for chinese word segmentation and new word detection. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 253–262. Association for Computational Linguistics.

Xu Sun, Takuya Matsuzaki, and Wenjie Li. 2013. Latent structured perceptrons for large-scale learning with hidden information. *Knowledge and Data Engineering, IEEE Transactions on*, 25(9):2063–2075.

Xu Sun, Wenjie Li, Houfeng Wang, and Qin Lu. 2014. Feature-frequency–adaptive on-line training for fast and accurate natural language processing. *Computational Linguistics*, 40(3):563–586.

Xu Sun. 2014. Structure regularization for structured prediction. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2402–2410.

Charles A. Sutton and Andrew McCallum. 2007. Piecewise pseudolikelihood for efficient training of conditional random fields. In *ICML'07*, pages 863–870. ACM.

Ben Taskar, Carlos Guestrin, and Daphne Koller. 2003. Max-margin markov networks. In *NIPS'03*.

Yoshimasa Tsuruoka, Yusuke Miyao, and JunÂąÂśichi Kazama. 2011. Learning with lookahead: Can history-based models rival globally optimized models? In *Conference on Computational Natural Language Learning*.

Stephen J Wright and Jorge Nocedal. 1999. *Numerical optimization*, volume 2. Springer New York.

Dekai Wu, Grace Ngai, Marine Carpuat, Jeppe Larsen, and Yongsheng Yang. 2002. Boosting for named entity recognition. In *proceedings of the 6th conference on Natural language learning-Volume 20*, pages 1–4. Association for Computational Linguistics.

Kazuhiro Yoshida and Jun'ichi Tsujii. 2007. Reranking for biomedical named-entity recognition. In *Proceedings of the Workshop on BioNLP 2007: Biological, Translational, and Clinical Language Processing*, pages 209–216. Association for Computational Linguistics.

Yi Zhang, Xu Sun, and Yang Yang. 2017. Does higher order LSTM have better accuracy in chunking and named entity recognition? *CoRR*, abs/1711.08231.