A Global Deep Reranking Model for Semantic Role Classification*

Haitong YANG†,†††, Guangyou ZHOU†,†††, Tingting HE†,††, Nonmembers, and Maoxi LI†††, Member

SUMMARY The current approaches to semantic role classification usually first define a representation vector for a candidate role and feed the vector into a deep neural network to perform classification. The representation vector contains some lexicalization features like word embeddings, lemmat embeddings. From linguistics, the semantic role frame of a sentence is a joint structure with strong dependencies between arguments which is not considered in current deep SRL systems. Therefore, this paper proposes a global deep reranking model to exploit these strong dependencies. The evaluation experiments on the CoNLL 2009 shared tasks show that our system can outperform a strong local system significantly that does not consider role dependency relations.

key words: deep reranking, representation vector, joint structure, dependencies

1. Introduction

The supervised methods are dominated in the Semantic Role Labeling (SRL) systems. The traditional methods[1]–[6] pay many attentions to feature engineerings and expect to exploit effective handcrafted features to improve SRL. In recent years, deep learning based models have witnessed many great successes in SRL. Deep learning based SRL systems[7]–[12] usually adopt continuous features like word embeddings as the representation vector of a candidate role and the representation vector is fed into a neural network for classification.

Under the view of linguistics, the semantic roles convey the meanings of the sentence as a whole and can be seen as a joint structure in which there are strong dependencies between different arguments. We take the sentence [Yesterday]AM-TMP [they]A0 hold Pred [a conference]A1 [in Beijing]AM-LOC as the example. Its predicate is ‘hold’ and its semantic roles contain A0 (an agent argument), A1 (a patient argument), AM-TMP (a time argument) and AM-LOC (a location argument). The set of all arguments and the predicate convey the whole meanings of the sentence. Therefore, the role label sequence (A0, A1, AM-TMP, AM-LOC) is a reasonable SRL result. If the argument [they] is classified into A1 incorrectly, the role label sequence (A1, A1, AM-TMP, AM-LOC) is not reasonable. Some traditional methods[5],[6] have claimed that dependency relations between different arguments are helpful for SRL. Therefore, in this paper we would like to know whether we can improve a deep SRL system with the help of dependency relations between different arguments.

To address the above problems, we propose a global deep reranking model for SRL which can efficiently explore rich dependencies between arguments of a semantic role frame. We first construct a strong local model to generate the top n SRL results and then design a global deep reranking model to improve the local model by reranking these initial results. The global model takes the whole sequence of role labels as the input and outputs a score for a candidate sequence. The global model is expected to assign a high score to the gold results and a low score to the incorrect results. So, we would obtain better results according to the scores. The key component of the global model is how to encode the sequence of role labels which implies the dependencies among different arguments. In this paper, we try four different ways to encoder it and compare the effectiveness of different ways quantitatively in the experiments. Like most machine learning problems, training data is critical for our global reranking model. But, unfortunately there are not direct available training data for our global model. To create training samples for our global model, we treat the gold labels as the positive samples and sample the incorrect results of the local model as the negative ones. We conduct experiments on the English portion of CoNLL09 shared tasks and results show that our method improves the local model significantly.

2. Related Work

In recent years, deep neural network has achieved great successes in the Natural Language Processing (NLP) community. A lot of neural models[7]–[15] have been proposed for SRL and a lot of promising results are reported. Some works[11]–[15] have investigated role dependency relations in the SRL system. In the works[11],[12], their idea is designing some dependency rules manually and then these
rules are incorporated into an Integer Linear Programming (ILP) framework. But their system is not efficient due to the difficulty of ILP’s optimization. The works [13]–[15] proposed deep learning based system to improve SRL with modeling of dependencies between multiple arguments in Japanese. But, different from them, our method is constructed based on deep reranking, which is simple and effective in implementation.

3. Methodology

3.1 Model Overview

The overview of the proposed model is shown in Fig. 1. The architecture consists of two parts: the global model (the upper) and the local model (the below) which is a strong deep learning based SRL system. Our idea is that the local model generates SRL candidates for the global model and then the global model reranks these candidates with the global features which encoding the dependency relations between different arguments. In the next, we will illustrate the local model and the global model in detail.

3.2 Local Model

We first construct a strong SRL system as our local model. It is a strong comparison system and also generates SRL candidates for the global model. From Fig. 1, our local model consists of four layers: BERT layer, BiLSTM layer, hidden layer, softmax layer.

**BERT layer** Recently, the pretraining based language models such as BERT have shown big improvement in a variety of natural language tasks. We choose BERT as the pretraining model in this paper. Following previous works [16], we first encode the sentence in a predicate-aware way, [[CLS] sentence [SEP] predicate [SEP]], which allows the representation of the predicate to interact with the entire sentence via appropriate attention mechanisms. Then, the above sequence is fed to the BERT encoder.

**BiLSTM layer** The task of SRL is usually treated as a sequence-to-sequence task. BiLSTM is a dominant sequence-to-sequence model widely used in the SRL systems. Thus, in this paper, we use BiLSTM to encode the input sequence. The BiLSTM processes the input sequence from both directions and we can obtain two separated hidden states $\bar{h}_t, \tilde{h}_t$. Then, we concatenate $\bar{h}_t$ and $\tilde{h}_t$ as the output vector and feed this vector into the next layer.

**Hidden Layer** The hidden layer is a full-connected layer. The ReLU activation function is employed in this layer.

**Softmax Layer** The softmax layer is the last one of the local model. It takes the outputs from the hidden layer and predict the final semantic roles by maximizing the likelihood of role labels.

3.3 Global Model

From the above illustration of the local model, we can see that the word representation layer only contains some independent features like word embedding while semantic roles of a sentence can be seen as a joint structure conveying the meaning of the sentence. Therefore, if we introduce the sequence of role labels into the model which encodes the dependency relations between different arguments, we should obtain better SRL results. Inspired by this idea, we propose a global reranking model.

In the global reranking model, we define a function $f(\cdot)$ and it computes a score $f(r)$ for any candidate role sequence $r$. Our goal is learning a good scoring function $f(\cdot)$ that assigns a high score to the gold results and a low score to the incorrect results, which can be formulated as the following.

$$f(r^+) > f(r^-), \quad \forall r^-$$

where $r^+$ is the gold role label sequence and $r^-$ is the incorrect one.

In this paper, we employ a neural network for the scoring function $f$. As shown in Fig. 1, the neural network consists of four layers: role embedding layer, feature encoder layer, hidden layer and sigmoid layer.

**Role embedding layer** We introduce an embedding representation for every role label. We concatenate the role representation and word representation as the first layer. This layer turns the sequence of role labels into a sequence of vectors.
Feature Encoder layer This layer encodes a sequence of vectors into a global vector. How to encode the input sequence is crucial for the global model. In this paper, we try the following four different manners,

1) Averaging the input nodes
2) Maximizing the input nodes
3) Convolutional Neural Network (CNN)
4) BiLSTM The hidden state of the last node is the final output.

Hidden layer This layer is a full-connected layer.

Sigmoid layer This layer generates the final score of a candidate role label sequence \( r \).

3.4 Training and Inference

There are not direct available training data for our global model. To address this problem, this paper proposes a simple method. Given a sentence, we choose its gold label as the positive sample \( r^+ \). The softmax layer of the local model can be seen as a discrete distribution, and thus we can sample an incorrect result as the negative sample \( r^- \) from the distribution.

For a pair sample \( (r^+, r^-) \), we define the following hinge loss for the model,

\[
\ell(r^+, r^-) = \max(0, e + f(r^+) - f(r^-))
\]

where \( e \) is a margin parameter that regularizes the distance of \( f(r^+) \) and \( f(r^-) \). The hinge loss is a convex loss function used for training classifiers like the SVM and has been proved to enhance the generalization of the ranking task[17]. However, the 'max' function in the hinge loss makes the network indifferentiable at the zero point. To smooth the hinge loss, we replace the max function with the LogSumExp\(^\dagger\) function which is a smooth approximation to the maximum function, mainly used in the machine learning community. Then, we can get the following derivation.

\[
\ell(r^+, r^-) = \log(e^0 + e^{\epsilon + f(r^+) - f(r^-)})
\]

\[
= \log(e^{f(r^-)} + e^{\epsilon + f(r^+)}) - f(r^-)
\]

Our objective function turns into

\[
\min \log(e^{f(r^-)} + e^{\epsilon + f(r^+)}) - f(r^-) + \lambda||W||
\]

where the third term is the regularization to improve the generalization and \( \lambda \) is a hyper-parameter to make the tradeoff between the hinge loss and the regularization.

In the inference stage, we utilize beam search algorithm to generate top \( n \) SRL results of the local model. Then, we use the global model to compute the scores of these \( n \) candidates. The candidate with the highest score is the final output of the global model. In the experiments, we set \( n \) to 20.

\(^\dagger\)https://en.wikipedia.org/wiki/LogSumExp

4. Experiments

4.1 Data Set

We measure the performance of our SRL system on a benchmark: the English portion of CoNLL 2009 shared tasks. We follow the standard split of CoNLL 2009: section 2-21 of the Wall Street Journal (WSJ) corpus as training set, section 24 as development set, section 23 as the in-domain test set and 3 sections from the Brown corpus as the out-domain test set. We use the official evaluation script\(^\dagger\dagger\) to compute the scores of all SRL systems.

4.2 Model Setup

In our experiments, the dimension of the label embedding is set to 100 and initialized by sampling from a uniform distribution between \(-0.1\) and \(0.1\). The BiLSTM layers contains 256-dimensional hidden units. The dropout rate is set to 0.1, the batch size is 32, the learning rate is 0.001 and the max epoch is 50. The hyper-parameter \( \lambda \) in Eq. (3) is set to 0.2.

4.3 Results

We compare the proposed method with the below systems,

- Zhao09. The system [3] reaches the best results on English portion of the CoNLL 2009 shared task.
- Local+ILP. Following [11], we implement Integer Linear Programming on the local system.

The results of all comparison systems are shown in Table 1. Zhao09 is a traditional SRL system and the other systems are based on the deep learning. The results show that the deep learning based SRL systems perform better than the traditional system. We first compare our global model with the local model. The results show that our global improve the SRL results significantly and outperforms the local model by 0.6 points \( F_1 \), which indicates that dependency relations between different arguments are helpful for a deep learning based SRL system and our method provides a simple and effective way to integrate dependency relations.

Next, we compare our method with Local+ILP. The two systems are constructed on the local model and can

| Corpus | Method   | P(%) | R(%) | F1(%)  |
|--------|----------|------|------|--------|
| WSJ    | Local    | 92.0 | 91.6 | 91.8   |
|        | Zhao09   | -    | -    | 85.4   |
|        | Local+ILP| 92.4 | 91.9 | 92.1   |
|        | Ours     | 92.6 | 92.2 | 92.4†  |
| Brown  | Local    | 85.3 | 84.4 | 84.8   |
|        | Zhao09   | -    | -    | 73.3   |
|        | Local+ILP| 85.9 | 84.9 | 85.4   |
|        | Ours     | 86.1 | 85.3 | 85.7†  |

\(^\dagger\dagger\)https://ufal.mff.cuni.cz/conll2009-st/eval09.pl
improve the local model. Our method achieves better results than Local+ILP. Besides, in Local+ILP we have to cost much human labors to design constraint rules manually while our system runs automatically. Lastly, ILP runs slowly especially when the problem has more than 20 variables. In our experiments, our system runs faster 30 times than ILP.

We also conduct experiments in the out-of-domain test set (Brown). The results are shown in Table 2. Many works [3], [4] have proved that the SRL task suffers severe performance drops on out-of-domain test data. Form the table, we also can see the $F_1$ of all systems drop heavily. The performance drop of our system on the out-of-domain data is smaller than other systems, which suggests that our method is also helpful for the out-of-the-domain data.

We propose four ways to encode the sequence of the role labels. Here, we discuss the effects of different encoding ways. The width of the CNN filters is set to 1, 2, 3 and 4. The results are shown in the Table 2. First, we can see that all SRL systems achieve higher $F_1$ scores than the local model which suggests that the global sequential label features are discriminative for the SRL task and can help improve the local model. Second, the BiLSTM way reach the better results than other ways. We think the reason is that BiLSTM performs powerfully in encoding the long term dependency, especially in the sequential data.

### 5. Conclusions

This paper investigates how to improve the deep learning based SRL system through exploiting dependencies relations of different semantic roles. We propose a global reranking model which can improve the SRL system by incorporating the global sequential features of roles. Experimental results demonstrate the effectiveness of our method.

### References

[1] D. Gildea and D. Jurafsky, “Automatic labeling for semantic roles,” Comput. Linguist., vol.28, no.3, pp.245–288, 2002.

[2] S. Pradhan, K. Hacioglu, V. Krugler, W. Ward, J.H. Martin, and D. Jurafsky, “Support Vector Learning for Semantic Argument Classification,” Mach. Learn., vol.60, no.1-3, pp.11–39, Sept. 2005.

[3] H. Zhao, W. Chen, C. Kit, and G. Zhou, “Multilingual dependency learning: Exploiting rich features for tagging syntactic and semantic dependencies,” Proc. Thirteenth Conference on Computational Natural Language Learning, Boulder, Colorado, pp.61–66, June 2009.

[4] N. Xue, “Labeling Chinese Predicates with Semantic Roles,” Comput. Linguist., vol.34, no.2, pp.225–255, June 2008.

[5] R. Collobert and J. Weston, “Fast semantic extraction using a novel neural network architecture,” Proc. 45th Annual Meeting of the Association of Computational Linguistics, Prague, Czech Republic, pp.560–567, June 2007.

[6] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, “Natural language processing (almost) from scratch,” J. Mach. Learn. Res., vol.12, pp.2493–2537, 2011.

[7] J. Zhou and W. Xu, “End-to-end learning of semantic role labeling using recurrent neural networks,” Proc. 53rd Annual Meeting of the Association for Computational Linguistics, Beijing, China, pp.1127–1137, July 2015.

[8] D. Marcheggiani and I. Titov, “Encoding Sentences with Graph Convolutional Networks for Semantic Role Labeling,” Proc. 2017 Conference on Empirical Methods in Natural Language Processing, Copenhagen, Denmark, pp.1506–1515, Sept. 2017.

[9] M. Roth and M. Lapata, “Neural Semantic Role Labeling with Dependency Path Embeddings” Proc. 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, pp.1192–1202, Aug. 2016.

[10] Z. Li, S. He, J. Cai, Z. Zhang, H. Zhao, G. Liu, L. Li, and L. Si, “A Unified Syntax-aware Framework for Semantic Role Labeling,” Proc. 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, pp.2401–2411, Oct. 2018.

[11] V. Punyakanok, D. Roth, W.-T. Yih, and D. Zimak, “Semantic Role Labeling via Integer Linear Programming Inference,” Proc. 20th International Conference on Computational Linguistics, Switzerland, Article no.1346, Aug. 2004.

[12] K. Toutanova, A. Haghighi, and C. Manning, “Joint Learning Improves Semantic Role Labeling,” Proc. 43th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Ann Arbor, Michigan, pp.589–596, June 2005.

[13] H. Ouchi, H. Shindo, K. Duh, and Y. Matsumoto, “Joint Case Argument Identification for Japanese Predicate Argument Structure Analysis,” Proc. 53rd Annual Meeting of the Association for Computational Linguistics, Beijing, China, pp.961–970, July 2015.

[14] H. Ouchi, H. Shindo, and Y. Matsumoto, “Neural Modeling of Multi-Predicate Interactions for Japanese Predicate Argument Structure Analysis,” Proc. 55rd Annual Meeting of the Association for Computational Linguistics, Vancouver, Canada, pp.1591–1600, July 2017.

[15] Y. Matsubayashi and K. Inai, “Distance-free modeling of multi-predicate interactions in end-to-end Japanese predicate-argument structure analysis,” Proc. 27th International Conference on Computational Linguistics, Santa Fe, New Mexico, USA, pp.94–106, Aug. 2018.

[16] P. Shi and J. Lin, “Simple bert models for relation extraction and semantic role labeling,” arXiv preprint arXiv:1904.05255, 2019. 3.

[17] J. Wang, Y. Song; T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu, “Learning Fine-Grained Image Similarity with Deep Ranking,” 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, pp.1386–1393, 2014.