A Syntax-aware Multi-task Learning Framework for Chinese Semantic Role Labeling

Qingrong Xia, Zhenghua Li*, Min Zhang
Institute of Artificial Intelligence, School of Computer Science and Technology,
Soochow University, China
kirosummer.nlp@gmail.com, {zhli13, minzhang}@suda.edu.cn

Abstract

Semantic role labeling (SRL) aims to identify the predicate-argument structure of a sentence. Inspired by the strong correlation between syntax and semantics, previous works pay much attention to improve SRL performance on exploiting syntactic knowledge, achieving significant results. Pipeline methods based on automatic syntactic trees and multi-task learning (MTL) approaches using standard syntactic trees are two common research orientations. In this paper, we adopt a simple unified span-based model for both span-based and word-based Chinese SRL as a strong baseline. Besides, we present a MTL framework that includes the basic SRL module and a dependency parser module. Different from the commonly used hard parameter sharing strategy in MTL, the main idea is to extract implicit syntactic representations from the dependency parser as external inputs for the basic SRL model. Experiments on the benchmarks of Chinese Proposition Bank 1.0 and CoNLL-2009 Chinese datasets show that our proposed framework can effectively improve the performance over the strong baselines. With the external BERT representations, our framework achieves new state-of-the-art 87.54 and 88.5 F1 scores on the two test data of the two benchmarks, respectively. In-depth analysis are conducted to gain more insights on the proposed framework and the effectiveness of syntax.

1 Introduction

Semantic role labeling (SRL) is a fundamental and important task in natural language processing (NLP), which aims to identify the semantic structure (*Who* did *what* to *whom*, *when* and *where*, etc.) of each given predicate in a sentence. Semantic knowledge has been widely exploited in many down-stream NLP tasks, such as information extraction (Bastianelli et al., 2013), machine translation (Liu and Gildea, 2010; Gao and Vogel, 2011) and question answering (Shen and Lapata, 2007; Wang et al., 2015a).

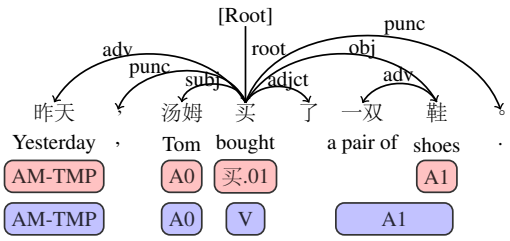

Figure 1: Example of span-based (blue blocks) and word-based (red blocks) SRL formulations in a sentence, where the top part is its dependency tree.

There are two formulations of SRL in the community according to the definition of semantic roles. The first is called *span-based* SRL, which employs a continuous word span as a semantic role and follows the manual annotations in the PropBank (Palmer et al., 2005) and NomBank (Meyers et al., 2004). The second is *word-based* SRL (Surdeanu et al., 2008), also called *dependency-based* SRL, whose semantic role is usually syntactic or semantic head word of the manually annotated word span. Figure 1 gives an example of the two forms in a sentence, where "bought" is the given predicate.

Intuitively, syntax and semantics are strongly correlative. For example, the semantic *A0* and *A1* roles are usually the syntactic subject and object, as shown in Figure 1. Inspired by the correlation, researchers try to improve SRL performance by exploring various ways to integrate syntactic knowledge (Roth and Lapata, 2016; He et al., 2018b; Swayamdipta et al., 2018). In contrast, some recent works (He et al., 2017; Tan et al., 2018; Cai et al., 2018) propose deep neural models for SRL without considering any syntactic in-

*Corresponding author.

formation, achieving promising results. Most recently, He et al. (2018a); Li et al. (2019) extend the span-based models to jointly tackle the predicate and argument identification sub-tasks of SRL.

Compared with the large amount of research for English SRL, Chinese SRL works are rare, mainly because of the limited amount of data and lack of attention of Chinese researchers. For Chinese, the commonly used datasets are Chinese Proposition Bank 1.0 (CPB1.0) (span-based) (Xue, 2008) and CoNLL-2009 Chinese (word-based) (Hajič et al., 2009). The CPB1.0 dataset follows the same annotation guideline with the English PropBank benchmark (Palmer et al., 2005). Wu and Palmer (2015) present a top model based selection preference approach to improve Chinese SRL. Since the amount of CPB1.0 dataset is small, Xia et al. (2017) exploit heterogeneous SRL data to improve the performance via a progressive learning approach. The CoNLL-2009 benchmark is released by the CoNLL-2009 shared task (Hajič et al., 2009). Previous works (Marcheggiani et al., 2017; He et al., 2018b; Cai et al., 2018) mainly focus on building more powerful models or exploring the usage of external knowledge on this dataset.

Inspired by the development of neural models and exploration of syntactic information, this paper proposes a MTL framework to extract syntactic representations as the external input features for the simple unified SRL model. The contributions of our paper are three-folds:

1. We introduce a simple unified model for span-based and word-based Chinese SRL.
2. We propose a MTL framework to extract implicit syntactic representations for SRL model, which significantly outperforms the baseline model.
3. Detailed analysis gains crucial insights on the effectiveness of our proposed framework.

We conduct experiments on the benchmarks of CPB1.0 and CoNLL-2009. The results show that our framework achieves new state-of-the-art 87.54 and 88.5 F1 scores on the two test data, respectively.

2 Basic SRL Model

Motivated by the recently presented span-based models (He et al., 2018a; Li et al., 2019) for jointly predicting predicates and arguments, we introduce a simple unified span-based model. Formally, given a sentence $s = w_1, w_2, ..., w_n$, the span-based model aims to predict a set of labeled predicate-argument relationships $\mathcal{Y} \subseteq \mathcal{P} \times \mathcal{A} \times \mathcal{R}$, where $\mathcal{P} = \{w_1, w_2, ..., w_n\}$ is the set of all candidate predicates, $\mathcal{A} = \{(w_i, ..., w_j) | 1 \leq i \leq j \leq n\}$ is the set of all candidate arguments, and $\mathcal{R}$ is the set of the semantic roles. Following He et al. (2018a), we also include a null label ϵ in the role set $\mathcal{R}$ indicating no relation between the focused predicate and argument. The model objective is to optimize the probability of the predicate-argument-role tuples $y \in \mathcal{Y}$ in a sentence s, which is formulated as:

$$\begin{aligned} P(y|s) &= \prod_{p \in \mathcal{P}, a \in \mathcal{A}, r \in \mathcal{R}} P(y_{(p,a,r)}|s) \\ &= \prod_{p \in \mathcal{P}, a \in \mathcal{A}, r \in \mathcal{R}} \frac{e^{\phi(p,a,r)}}{\sum_{r' \in \mathcal{R}} e^{\phi(p,a,r')}} \end{aligned} \quad (1)$$

where $\phi(p, a, r) = \phi_p(p) + \phi_a(a) + \phi_r(p, a)$ is the score of the predicate-argument-relation tuple. We directly adopt the model architecture of He et al. (2018a) as our basic SRL model with a modification on the argument representation. The architecture of the basic SRL module is shown in the right part of Figure 2, and we will describe it in the following subsections.

2.1 Input Layer

Following He et al. (2018a); Li et al. (2019), we employ CNNs to encode Chinese characters for each word w_i into its character representation, denoted as $\boldsymbol{rep}_i^{char}$. Then, we concatenate $\boldsymbol{rep}_i^{char}$ with the word embedding $\boldsymbol{emb}_i^{word}$ to represent the word-level features as our basic model input. In addition, we also employ BERT representations (Devlin et al., 2019) to boost the performance of our baseline model, which we denote as $\boldsymbol{rep}_i^{BERT}$. Formally, the input representation of w_i is:

$$\boldsymbol{x}_i = \boldsymbol{rep}_i^{char} \oplus \boldsymbol{emb}_i^{word} \oplus \boldsymbol{rep}_i^{BERT} \quad (2)$$

, where $\oplus$ is the concatenation operation. Our basic SRL model and BERT-enhanced baseline depend on whether including the BERT representation $\boldsymbol{rep}_i^{BERT}$ or not.

2.2 BiLSTM Encoder

Over the input layer, we employ the BiLSTMs with highway connections (Srivastava et al., 2015;

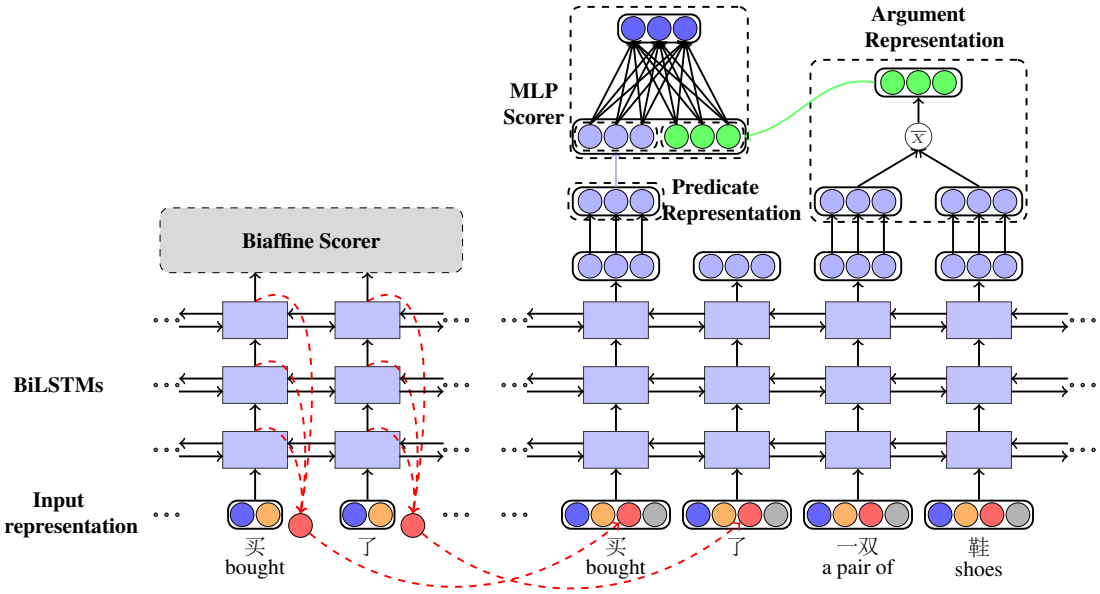

Figure 2: The detailed architecture of our proposed framework, where the left part is the dependency parser and the right part is the basic SRL module, respectively.

Zhang et al., 2016b) to encode long-range dependencies and obtain rich representations denoted as $\boldsymbol{h}_i$ for time stamp i. The highway connections are used to alleviate the gradient vanishing problem when training deep neural networks.

2.3 Predicate and Argument Representations

We directly employ the output of the top BiLSTM as the predicate representation at each time stamp. For all the candidate arguments, we simplify the representations by employing the mean operation over the BiLSTM outputs within the corresponding argument spans, which achieves similar results compared with the attention-based span representations (He et al., 2018a) on English SRL in our preliminary experiments. Formally,

$$\begin{aligned} \boldsymbol{rep}^p_i &= \boldsymbol{h}_i \\ \boldsymbol{rep}^a_{j,k} &= mean(\boldsymbol{h}_j, ..., \boldsymbol{h}_k) \\ & (1 \leq i \leq n; 1 \leq j \leq k \leq n) \end{aligned} \tag{3}$$

Specifically, for word-based SRL, we only need to set the length of candidate arguments to be 1.

2.4 MLP Scorer

We employ the MLP scorers as the scoring functions to determine whether the candidate predicates or arguments need to be pruned. Another MLP scorer is employed to compute the score of whether the focused candidate predicate and argument can compose a semantic relation.

$$\begin{aligned} \phi_p(p) &= \mathbf{w}_\mathbf{p}^\top \mathbf{MLP}_\mathbf{p}(\boldsymbol{rep}^p_i) \\ \phi_a(a) &= \mathbf{w}_\mathbf{a}^\top \mathbf{MLP}_\mathbf{a}(\boldsymbol{rep}^a_{j,k}) \\ \phi_r(p,a) &= \mathbf{w}_\mathbf{r}^\top \mathbf{MLP}_\mathbf{r}([\boldsymbol{rep}^p_i; \boldsymbol{rep}^a_{j,k}]) \end{aligned} \tag{4}$$

3 Proposed Framework

Our framework includes two modules, a basic SRL module and a dependency parser module, as shown in Figure 2. In this section, we will first describe the architecture of the employed dependency parser, and then illustrate the integration of the syntactic parser into the basic SRL model.

3.1 Dependency Parser Module

We employ the state-of-the-art biaffine parser proposed by Dozat and Manning (2017) as the dependency parser module in our framework, as shown by the left part of Figure 2. In order to better fit the dependency parser into our framework, we make some modifications on the original model architecture. First, we remove the Part-of-Speech (PoS) tagging embeddings and add the Chinese character CNN representations, so the resulting input representation is the same as the SRL module. Second, we substitute the BiLSTMs in the original biaffine parser with the same 3-layer highway BiLSTMs used in our SRL module. The biaffine scorer is proposed to compute the score of candidate syntactic head and modifier, which remains unchanged.

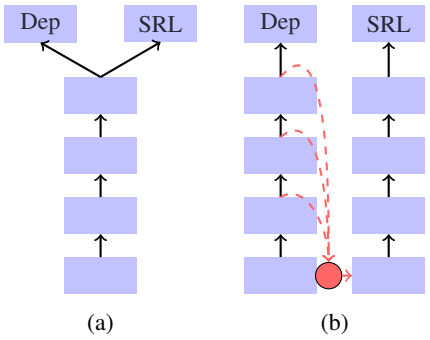

Figure 3: Comparison between the architectures of hard parameter sharing (a) and our proposed implicit representation integration (b).

3.2 Details of Integration

Multi-task learning (MTL) approaches can effectively exploit the standard dependency trees to improve the SRL performance, regarding dependency parsing as the auxiliary task. Hard parameter sharing (Ruder, 2017) is the most commonly used method in MTL, which shares several common layers between tasks and keeps the task-specific output layers, as illustrated by the sub-figure a in Figure 3. We propose a better way to integrate the focused two tasks in this work. In the following, we will describe the intuition and details of the integration on the two tasks.

As is well known, the hard parameter sharing approach can provide representations for all the shared tasks and reduce the probability of overfitting on the main task. However, this kind of sharing strategy somewhat weakens the representation framework maintains distinct model parameters for each task, due to the neutralization of knowledge introduced by the auxiliary task. Different from the hard parameter sharing strategy, we propose to integrate the syntactic information into the input layer of the basic SRL module, as illustrated by sub-figure b of Figure 3. And Figure 2 shows the detailed architecture. First, we extract all the 3 BiLSTM hidden outputs of the dependency parser as the syntactic representations. Second, we employ the normalized weights to sum the extracted representations as the final syntactic representation for word w_i, denoted as $\boldsymbol{rep}_i^{syn}$. Formally,

$$\boldsymbol{rep}_i^{syn} = \sum_{1 \le j \le N} \alpha_j * \boldsymbol{h}_i^j \tag{5}$$

where N is the layer number of the highway BiLSTMs, and α_j is the $j-$th softmax weight. Finally, the extracted syntactic representations are fed into the input layer of the SRL module, and concatenated with the original SRL module input. We design this framework for several considerations: 1) the proposed framework keeps the own model parameters for each task, thereby maximizing task-specific information for the main task, 2) the dependency parser module can be updated by the gradients returned from the extracted syntactic representations, which can encourage it to produce semantic preferred representations.

3.3 Training Objective

Given the sets of SRL data $\mathcal{S}$ and dependency data $\mathcal{D}$, the framework loss function is defined as the sum of the negative log-likelihood loss of the two tasks:

$$-\Big(\sum_{(Y_s^*, X_s) \in \mathcal{S}} \log P(Y_s^* | X_s) + \alpha \sum_{(Y_d^*, X_d) \in \mathcal{D}} \log P(Y_d^* | X_d) \Big) \tag{6}$$

where Y_s^* and Y_d^* are gold semantic and syntactic structures respectively, and α is a corpus weighting factor to control the loss contribution of the dependency data in each batch as discussed in the experiments.

4 Experiments

4.1 Settings

We evaluate the proposed MTL framework on two commonly used benchmark datasets of Chinese: Chinese Proposition Bank 1.0 (CPB1.0) (span-based) (Xue, 2008) and CoNLL-2009 (word-based) (Hajič et al., 2009). Following previous works, we report the results of span-based SRL in two setups: *pre-identified predicates* and *end-to-end*. For word-based SRL, we only report the results in the *pre-identified predicates* setting. Following Roth and Lapata (2016), we employ the mate-tools[1] (Björkelund et al., 2010) for the predicate disambiguation, which achieves 94.87% and 94.91% F1 scores on the CoNLL-2009 Chinese development and test data respectively.

Dependency Parsing Data. We employ the Chinese Open Dependency Treebank[2] constructed at Soochow University. The treebank construction

[1] https://code.google.com/archive/p/mate-tools/
[2] http://hlt.suda.edu.cn/index.php/CODT

project aims to continually build a large-scale Chinese dependency treebank that covers up-to-date texts from different domains and sources (Peng et al., 2019). So far, CODT contains 67,679 sentences from 9 different domains or sources.

BERT Representations. Recently, BERT (Bidirectional Encoder Representations from Transformers) is proposed by Devlin et al. (2019), which makes use of Transformers to learn contextual representations between words. In this paper, we use the pre-trained Chinese model[3] to extract the BERT representations for our span-based and word-based SRL datasets. We extract the fixed BERT representations from the last four hidden layers of the pre-trained model. Finally, we also employ the normalized weighted sum operation to obtain the final BERT representation for each word w_i, denoted as $\boldsymbol{rep}_i^{BERT}$.

Hyperparameters. We employ word2vec (Mikolov et al., 2013) to train the Chinese word embeddings on the Chinese Gigaword dataset[4]. The Chinese char embeddings are randomly initialized, and the dimension is 100. We employ the CNN to get the Chinese char representations, which has window size of 3, 4 and 5, and the output channel size is 100. For other parameter settings in the SRL module, we mostly follow the work of He et al. (2018a). As for the pruning of candidate predicates and arguments, we choose the pruning ratios according to the training data, using $\lambda_p = 0.4$ for predicates and $\lambda_a = 0.8$ for arguments with up to 30 words.

Training Criterion. We choose Adam (Kingma and Ba, 2015) optimizer with 0.001 as the initial learning rate and 0.1% as the decay rate for every 100 steps. Each data batch is composed of both SRL and dependency instances. We randomly shuffle the SRL and dependency training datasets if the smaller SRL data is used up. All baseline models are trained for at most 180,000 steps, and 100,000 steps for other models. In addition, we pick the best model on the development data for testing. We apply 0.5 dropout to the word embeddings and Chinese character representations and 0.2 dropout to all hidden layers. We employ variational dropout masks that are shared across all timesteps (Gal and Ghahramani, 2016) for the highway BiLSTMs, with 0.4 dropout rate.

Evaluation. We adopt the official scripts provided by CoNLL-2005[5] and CoNLL-2009[6] for span-based and word-based SRL evaluation, respectively. We conduct significant tests using the Dan Bikel's randomized parsing evaluation comparer.

4.2 Syntax-aware Methods

To illustrate the effectiveness and advantage of our proposed framework[7] (Integration of Implicit Representations, IIR), we conduct several experiments with the recently employed syntax-aware methods on CPB1.0 dataset for comparison:

- **Tree-GRU** Xia et al. (2019) investigate several syntax-aware methods for the English span-based SRL, showing the effectiveness of introducing syntactic knowledge into the SRL task. We only compare with the Tree-GRU method, since the other methods are all predicate-specific and hence not fit into our basic SRL model.

- **FIR** Following Yu et al. (2018) and Zhang et al. (2019), we extract the outputs of BiLSTMs as the fixed implicit representations (FIR) from a pre-trained biaffine parser. In detail, we train the biaffine parser with the same training data used in our framework, and employ the combination of development data of CDT (997 sentences) and PCTB7 (998 sentences) as the development data. The biaffine parser achieves 79.71% UAS and 74.74% LAS on the combined development data.

- **HPS** We employ the commonly used hard parameter sharing (HPS) strategy of MTL as a strong baseline, which shares the word and char embeddings and 3-layer BiLSTMs between the dependency parser and the basic SRL module.

4.3 Main Results

Results of Syntax-aware Methods. Table 1 shows the results of these syntax-aware methods on CPB1.0 dataset. First, the first line shows the results of our baseline model, which only employs the word embeddings and char representations as the inputs of the basic SRL model. Second, the

[3] https://github.com/google-research/bert#pre-trained-models

[4] https://catalog.ldc.upenn.edu/LDC2003T09

[5] http://www.cs.upc.edu/~srlconll/st05/st05.html

[6] https://ufal.mff.cuni.cz/conll2009-st/scorer.html

[7] https://github.com/KiroSummer/A_Syntax-aware_MTL_Framework_for_Chinese_SRL

Methods	Dev			Test		
	P	R	F1	P	R	F1
Baseline	81.52	82.17	81.85	80.95	80.01	80.48
Baseline + Dep (Tree-GRU)	82.35	80.24	81.28	82.10	78.11	80.06
Baseline + Dep (FIR)	**83.56**	83.05	83.30	83.38	81.93	82.65
Baseline + Dep (HPS)	82.58	**84.15**	83.36	83.22	**83.81**	83.51
Baseline + Dep (IIR)	83.12	83.66	**83.39**	**84.49**	83.34	**83.91**

Table 1: Experimental results of syntax-aware methods we compare on CPB1.0 dataset.

Tree-GRU method only achieves 80.06 F1 score on the test data, which even didn't catch up with the baseline model. We think this is caused by the relatively low accuracy in Chinese dependency parsing. Third, the FIR approach outperforms the baseline by 2.17 F1 score on the test data, demonstrating the effectiveness of introducing fixed implicit syntactic representations. Forth, the HPS strategy achieves more significant performance by 83.51 F1 score. Finally, our proposed framework achieves the best performance of 83.91 F1 score among these methods, outperforming the baseline by 3.43 F1 score. All the improvements are statistically significant ($p < 0.0001$). From these experimental results, we can conclude that: 1) the quality of syntax has a crucial impact on the methods which depend on the systematic dependency trees, like Tree-GRU, 2) the implicit syntactic features have the potential to improve the down-stream NLP tasks, and 3) learning the syntactic features with the main task performs better than extract them from a fixed dependency parser.

Results on CPB1.0. Table 2 shows the results of our baseline model and proposed framework using external dependency trees on CPB1.0, as well as the corresponding results when adding BERT representations. It is clear that adding dependency trees into the baseline SRL model can effectively improve the performance ($p < 0.0001$), no matter whether employ the BERT representations or not. Especially, our proposed framework (IIR) consistently outperforms the hard parameter sharing strategy. So we only report the results of our proposed framework in later experiments. Our final results outperforms the best previous model (Xia et al., 2017) by 7.87 and 4.24 F1 scores with BERT representations or not, respectively.

Table 3 shows the results of our framework in the *end-to-end* setting. To our best knowledge, we are the first to present the results of *end-to-end* on the CPB1.0 dataset. We achieve the result of 85.57 in F1 score, which is a strong baseline for later works. It is clear that our framework can still achieve better results compared with the strong baseline, which employs BERT representations as the external input.

Methods	**F1**
Previous Works	
Sun et al. (2009)	74.12
Wang et al. (2015b)	77.59
Sha et al. (2016)	77.69
Xia et al. (2017)	79.67
Ours	
Baseline	80.48
Baseline + Dep (HPS)	83.51
Baseline + Dep (IIR)	**83.91**
Baseline + BERT	86.62
Baseline + BERT + Dep (HPS)	87.03
Baseline + BERT + Dep (IIR)	**87.54**

Table 2: Results and comparison with previous works on CPB1.0 test set.

Results on CoNLL-2009. Table 4 shows the results of our framework and comparison with previous works on the CoNLL-2009 Chinese test data. Our baseline achieves nearly the same per-

Methods	**Dev** **F1**	**Test** **F1**
Ours		
Baseline	80.37	79.29
Baseline + Dep (IIR)	**82.39**	**81.73**
Baseline + BERT	85.30	85.26
Baseline + BERT + Dep (IIR)	**85.92**	**85.57**

Table 3: F1 scores of end-to-end settings on CPB1.0 test set.

Methods	P	R	F1
Previous Works			
Roth and Lapata (2016)	83.2	75.9	79.4
Marcheggiani et al. (2017)	84.6	80.4	82.5
He et al. (2018b)	84.2	81.5	82.8
Cai et al. (2018)	84.7	84.0	84.3
Ours			
Baseline	83.7	84.8	84.2
Baseline + Dep (IIR)	**84.6**	**85.7**	**85.1**
Baseline + BERT	87.8	**89.2**	**88.5**
Baseline + BERT + Dep (IIR)	**88.0**	89.1	**88.5**

Table 4: Results and comparison with previous works on CoNLL-2009 Chinese test set.

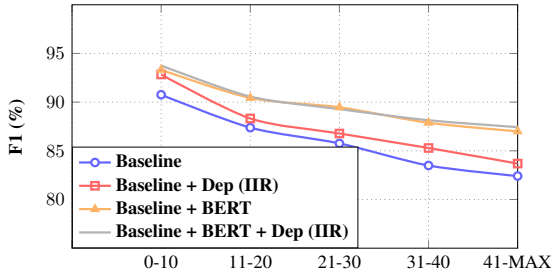

Figure 4: F1 scores regarding to the sentence length of the CoNLL-2009 Chinese dev data.

formance with Cai et al. (2018), which is an end-to-end neural model that consists of BiLSTM encoder and biaffine scorer. Our proposed framework outperforms the best reported result (Cai et al., 2018) by 0.8 F1 score and brings a significant improvement ($p < 0.0001$) of 0.9 F1 score over our baseline model. Our experimental result boosts to 88.5 F1 score when the framework is enhanced with BERT representations. However, compared with the results in the settings without BERT, the improvement is fairly small (88.53 - 88.47 = 0.06 F1 score, $p > 0.1$)[8] of the proposed framework, which we will discuss in Section 5.3.

5 Analysis

In this section, we conduct detailed analysis to understand the improvements introduced by our proposed framework.

5.1 Long-distance Dependencies

To analyze the effect of the proposed framework regarding to the distance of sentence lengths, we report the F1 scores of different sets of sentence lengths, as shown in Figure 4. We can see that improvements are obtained for nearly all sets of sentences, especially on the sentences with long-distance. It demonstrate that *syntactic knowledge is beneficial for SRL and effective to capture long-distance dependencies*.

[8] Following previous works, we only retrain the experimental results with one decimal point

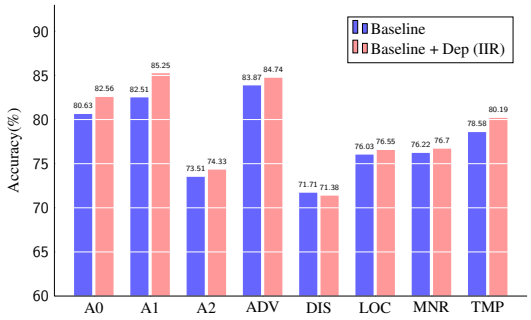

Figure 5: Accuracy comparison of different semantic roles between *Baseline* and *Baseline + Dep (IIR)* on CoNLL-2009 dev data.

5.2 Improvements on Semantic Roles

To find which semantic roles benefit from our syntax-aware framework, we report the F1 scores on several semantic role labels in Figure 5. We can see that syntax helps most on the *A0* and *A1* roles, which is consistent with the intuition that the semantic *A0* and *A1* roles are usually the syntactic subject and object of a verb predicate. Other adjunct semantic roles like *ADV*, *LOC*, *MNR* and *TMP* all benefit from the introduction of syntactic information. There is an interesting finding that the *DIS* role obtains worse performance when introduce syntactic information. We conduct error analysis on this phenomena, and we found that the framework mostly confuses *DIS* with *ADV*. The possible reason is that the two semantic roles are both labeled as *adv* in syntax.

5.3 Integration with BERT

BERT is employed to boost the performance of our basic SRL model and our proposed framework. Compared with results in the settings without BERT, the improvements of our framework over the BERT-enhanced baseline are fairly small on CoNLL-2009, as shown by the last two lines in Table 4. To analyze the difference between the two models (*Baseline + BERT* and *Baseline + BERT + Dep (IIR)*), we conduct an analysis on the sentence performance comparison between them, which is inspired by Zhang et al. (2016a). As shown in Fig-

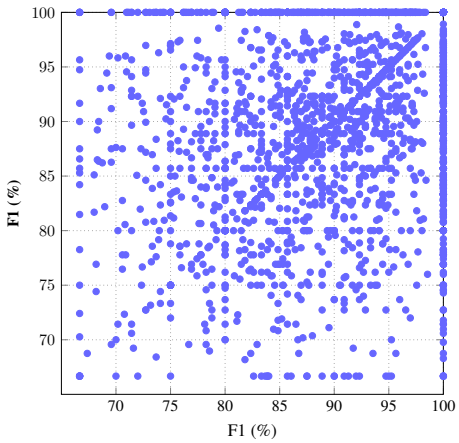

Figure 6: Sentence F1 scores comparison on CoNLL-2009 Chinese test data, where the x axis presents the F1 scores of *Baseline + BERT* and y axis shows the F1 scores of *Baseline + BERT + Dep (IIR)*, respectively.

ure 6, we can see that most of the scatter points are off the diagonal line, demonstrating strong differences between the two models. Based on this finding, how to better integrate syntactic knowledge and BERT representations becomes an interesting and meaningful question, and we leave it for future work.

6 Related Work

Traditional discrete-feature-based SRL works (Swanson and Gordon, 2006; Zhao et al., 2009) mainly make heavy use of syntactic information. Along with the impressive development of neural-network-based approaches in the NLP community, much attention has been paid to build more powerful neural model without considering any syntactic information. Zhou and Xu (2015) employ deep stacked BiLSTMs and achieve strong performance for span-based English SRL. He et al. (2017) extend their work (Zhou and Xu, 2015) by employing several advanced practices in recent deep learning literature, leading to significant improvements. Tan et al. (2018) present a strong self-attention based model, achieving significant improvements. Inspired by the span-based model proposed by Lee et al. (2017) for coreference resolution, He et al. (2018a); Ouchi et al. (2018) present similar span-based models for SRL which can exploit span-level features. For word-based SRL, Marcheggiani et al. (2017) propose a simple and fast syntax-agnostic model with rich input representations. Cai et al. (2018) present an end-to-end model with BiLSTMs and biaffine scorer to jointly handle the predicate disambiguation and the argument labeling sub-tasks.

Apart from the above syntax-free works, researchers also pay much attention on improving the neural-based SRL approaches by introducing syntactic knowledge. Roth and Lapata (2016) introduce the dependency path embeddings to the neural-based model and achieve substantial improvements. Marcheggiani and Titov (2017) employ the graph convolutional neural networks on top of the BiLSTM encoder to encode syntactic information. He et al. (2018b) propose a k-th order argument pruning algorithm based on systematic dependency trees. Strubell et al. (2018) propose a self-attention based neural MTL model which incorporate dependency parsing as a auxiliary task for SRL. Swayamdipta et al. (2018) propose a MTL framework using hard parameter strategy to incorporate constituent parsing loss into semantic tasks, i.e. SRL and coreference resolution, which outperforms their baseline by +0.8 F1 score. Xia et al. (2019) investigate and compare several syntax-aware methods on span-based SRL, showing the effectiveness of integrating syntactic information.

Compared with the large amount of works on English SRL, Chinese SRL works are rare, mainly because of the limitation of datasize and lack of attention of Chinese researchers. Sun et al. (2009) treat the Chinese SRL as a sequence labeling problem and build a SVM-based model by exploiting morphological and syntactic features. Wang et al. (2015b) build a basic BiLSTM model and introduce a way to exploit heterogeneous data by sharing word embeddings. Xia et al. (2017) propose a progressive model to learn and transfer knowledge from heterogeneous SRL data. The above works are all focus on the span-based Chinese SRL, and we compare with their results in Table 2. Different from them, we propose a MTL framework to integrate implicit syntactic representations into a simple unified model on both span-based and word-based SRL, achieving substantial improvements.

In addition to the hard parameter sharing strategy that we discuss in Section 3.2, partial parameter sharing strategy is also a commonly studied approach in MTL and domain adaptation. Kim et al. (2016) introduce simple neural extensions of feature argumentation by employing a global LSTM used across all domains and independent LSTMs used within individual domains. Peng

et al. (2017) explore a multitask learning approach which shares parameters across formalisms for semantic dependency parsing. In addition, Peng et al. (2018) present a multi-task approach for frame-semantic parsing and semantic dependency parsing with latent structured variables.

7 Conclusion

This paper proposes a syntax-aware MTL framework to integrate implicit syntactic representations into a simple unified SRL model. The experimental results show that our proposed framework can effectively improve the basic SRL model, even when the basic model is enhanced with BERT representations. Especially, our proposed framework is more effective at utilizing syntactic information, compared with the hard parameter sharing strategy of MTL. By utilizing BERT representations, our framework achieves new state-of-the-art performance on both span-based and word-based Chinese SRL benchmarks, i.e. CPB1.0 and CoNLL-2009 respectively. Detailed analysis shows that syntax helps most on the long sentences, because of the long-distance dependencies captured by syntax trees. Moreover, the comparison of sentence performance indicates that there is still a lot of work to do to better integrate syntactic information and BERT representation.

Acknowledgments

We thank our anonymous reviewers for their helpful comments. This work was supported by National Natural Science Foundation of China (Grant No. 61525205, 61876116, 61432013) and a project funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions.

References

Emanuele Bastianelli, Giuseppe Castellucci, Danilo Croce, and Roberto Basili. 2013. Textual inference and meaning representation in human robot interaction. In *Proceedings of JSSP*, pages 65–69.

Anders Björkelund, Bernd Bohnet, Love Hafdell, and Pierre Nugues. 2010. A high-performance syntactic and semantic dependency parser. In *Proceedings of COLING*, pages 33–36.

Jiaxun Cai, Shexia He, Zuchao Li, and Hai Zhao. 2018. A full end-to-end semantic role labeler, syntactic-agnostic over syntactic-aware? In *Proceedings of COLING*, pages 2753–2765.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

Timothy Dozat and Christopher D Manning. 2017. Deep biaffine attention for neural dependency parsing. In *Proceedings of ICIR*.

Yarin Gal and Zoubin Ghahramani. 2016. A theoretically grounded application of dropout in recurrent neural networks. In *Proceedings of NIPS*, pages 1019–1027.

Qin Gao and Stephan Vogel. 2011. Corpus expansion for statistical machine translation with semantic role label substitution rules. In *Proceedings of ACL*, pages 294–298.

Jan Hajič, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria Antònia Martí, Lluís Màrquez, Adam Meyers, Joakim Nivre, Sebastian Padó, Jan Štěpánek, et al. 2009. The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of CoNLL: Shared Task*, pages 1–18. ACL.

Luheng He, Kenton Lee, Omer Levy, and Luke Zettlemoyer. 2018a. Jointly predicting predicates and arguments in neural semantic role labeling. In *Proceedings of ACL*, pages 364–369.

Luheng He, Kenton Lee, Mike Lewis, and Luke Zettlemoyer. 2017. Deep semantic role labeling: What works and whats next. In *Proceedings of ACL*, pages 473–483.

Shexia He, Zuchao Li, Hai Zhao, and Hongxiao Bai. 2018b. Syntax for semantic role labeling, to be, or not to be. In *Proceedings of ACL*, pages 2061–2071.

Young-Bum Kim, Karl Stratos, and Ruhi Sarikaya. 2016. Frustratingly easy neural domain adaptation. In *Proceedings of COLING*, pages 387–396.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of ICLR*.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of EMNLP*, pages 188–197.

Zuchao Li, Shexia He, Hai Zhao, Yiqing Zhang, Zhuosheng Zhang, Xi Zhou, and Xiang Zhou. 2019. Dependency or span, end-to-end uniform semantic role labeling. In *Proceedings of AAAI*.

Ding Liu and Daniel Gildea. 2010. Semantic role features for machine translation. In *Proceedings of COLING*, pages 716–724.

Diego Marcheggiani, Anton Frolov, and Ivan Titov. 2017. A simple and accurate syntax-agnostic neural model for dependency-based semantic role labeling. In *Proceedings of CoNLL*, pages 411–420.

Diego Marcheggiani and Ivan Titov. 2017. Encoding sentences with graph convolutional networks for semantic role labeling. In *Proceedings of EMNLP*, pages 1506–1515.

Adam Meyers, Ruth Reeves, Catherine Macleod, Rachel Szekely, Veronika Zielinska, Brian Young, and Ralph Grishman. 2004. The nombank project: An interim report. In *Proceedings of HLT-NAACL*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of NIPS*, pages 3111–3119.

Hiroki Ouchi, Hiroyuki Shindo, and Yuji Matsumoto. 2018. A span selection model for semantic role labeling. In *Proceedings of EMNLP*, pages 1630–1642.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational linguistics*, 31(1):71–106.

Hao Peng, Sam Thomson, and Noah A Smith. 2017. Deep multitask learning for semantic dependency parsing. In *Proceedings of ACL*, pages 2037–2048.

Hao Peng, Sam Thomson, Swabha Swayamdipta, and Noah A Smith. 2018. Learning joint semantic parsers from disjoint data. In *Proceedings of NAACL-HLT*, pages 1492–1502.

Xue Peng, Zhenghua Li, Min Zhang, Rui Wang, Yue Zhang, and Luo Si. 2019. Overview of the nlpcc 2019 shared task: Cross-domain dependency parsing. In *Proceedings of The 8th CCF International Conference on Natural Language Processing and Chinese Computing (NLPCC)*.

Michael Roth and Mirella Lapata. 2016. Neural semantic role labeling with dependency path embeddings. In *Proceedings of ACL*, pages 1192–1202.

Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*.

Lei Sha, Sujian Li, Baobao Chang, Zhifang Sui, and Tingsong Jiang. 2016. Capturing argument relationship for chinese semantic role labeling. In *Proceedings of EMNLP*, pages 2011–2016.

Dan Shen and Mirella Lapata. 2007. Using semantic roles to improve question answering. In *Proceedings of EMNLP-CoNLL*, pages 12–21.

Rupesh K Srivastava, Klaus Greff, and Jürgen Schmidhuber. 2015. Training very deep networks. In *Proceedings of NIPS*, pages 2377–2385.

Emma Strubell, Patrick Verga, Daniel Andor, David Weiss, and Andrew McCallum. 2018. Linguistically-informed self-attention for semantic role labeling. In *Proceedings of EMNLP*, pages 5027–5038.

Weiwei Sun, Zhifang Sui, Meng Wang, and Xin Wang. 2009. Chinese semantic role labeling with shallow parsing. In *Proceedings of EMNLP*, pages 1475–1483.

Mihai Surdeanu, Richard Johansson, Adam Meyers, Lluís Màrquez, and Joakim Nivre. 2008. The conll-2008 shared task on joint parsing of syntactic and semantic dependencies. In *Proceedings of CoNLL*, pages 159–177.

Reid Swanson and Andrew S Gordon. 2006. A comparison of alternative parse tree paths for labeling semantic roles. In *Proceedings of COLING/ACL*, pages 811–818.

Swabha Swayamdipta, Sam Thomson, Kenton Lee, Luke Zettlemoyer, Chris Dyer, and Noah A Smith. 2018. Syntactic scaffolds for semantic structures. In *Proceedings of EMNLP*, pages 3772–3782.

Zhixing Tan, Mingxuan Wang, Jun Xie, Yidong Chen, and Xiaodong Shi. 2018. Deep semantic role labeling with self-attention. In *Proceedings of AAAI*, pages 4929–4936.

Hai Wang, Mohit Bansal, Kevin Gimpel, and David McAllester. 2015a. Machine comprehension with syntax, frames, and semantics. In *Proceedings of ACL-IJCNLP*, pages 700–706.

Zhen Wang, Tingsong Jiang, Baobao Chang, and Zhifang Sui. 2015b. Chinese semantic role labeling with bidirectional recurrent neural networks. In *Proceedings of EMNLP*, pages 1626–1631.

Shumin Wu and Martha Palmer. 2015. Can selectional preferences help automatic semantic role labeling? In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics*, pages 222–227.

Qiaolin Xia, Lei Sha, Baobao Chang, and Zhifang Sui. 2017. A progressive learning approach to chinese srl using heterogeneous data. In *Proceedings of ACL*, pages 2069–2077.

Qingrong Xia, Zhenghua Li, Min Zhang, Zhang Meishan, Guohong Fu, Rui Wang, and Luo Si. 2019. Syntax-aware neural semantic role labeling. In *Proceedings of AAAI*.

Nianwen Xue. 2008. Labeling chinese predicates with semantic roles. *Computational linguistics*, 34(2):225–255.

Nan Yu, Meishan Zhang, and Guohong Fu. 2018. Transition-based neural rst parsing with implicit syntax features. In *Proceedings of COLING*, pages 559–570.

Meishan Zhang, Zhenghua Li, Guohong Fu, and Min Zhang. 2019. Syntax-enhanced neural machine translation with syntax-aware word representations. In *Proceedings of NAACL-HLT*, pages 1151–1161.

Meishan Zhang, Yue Zhang, and Guohong Fu. 2016a. Transition-based neural word segmentation. In *Proceedings of ACL*, pages 421–431.

Yu Zhang, Guoguo Chen, Dong Yu, Kaisheng Yaco, Sanjeev Khudanpur, and James Glass. 2016b. Highway long short-term memory rnns for distant speech recognition. In *Proceedings of ICASSP*, pages 5755–5759.

Hai Zhao, Wenliang Chen, and Chunyu Kit. 2009. Semantic dependency parsing of nombank and propbank: An efficient integrated approach via a large-scale feature selection. In *Proceedings of EMNLP*, pages 30–39.

Jie Zhou and Wei Xu. 2015. End-to-end learning of semantic role labeling using recurrent neural networks. In *Proceedings of ACL-IJCNLP*, pages 1127–1137.