A Progressive Learning Approach to Chinese SRL Using Heterogeneous Data

Qiaolin Xia, Baobao Chang and **Zhifang Sui**
MOE Key Laboratory of Computational Linguistics, Peking University
School of Electronics Engineering and Computer Science, Peking University
{xql,chbb,szf}@pku.edu.cn

arXiv:1702.06740v1 [cs.CL] 22 Feb 2017

Abstract

Previous studies on Chinese semantic role labeling (SRL) have concentrated on single semantically annotated corpus. But the training data of single corpus is often limited. Meanwhile, there usually exists other semantically annotated corpora for Chinese SRL scattered across different annotation frameworks. Data sparsity remains a bottleneck. This situation calls for larger training datasets, or effective approaches which can take advantage of highly heterogeneous data. In these papers, we focus mainly on the latter, that is, to improve Chinese SRL by using heterogeneous corpora together. We propose a novel progressive learning model which augments the Progressive Neural Network with Gated Recurrent Adapters. The model can accommodate heterogeneous inputs and effectively transfer knowledge between them. We also release a new corpus, Chinese SemBank, for Chinese SRL. Experiments on CPB 1.0 show that ours model outperforms state-of-the-art methods.

1 Introduction

Semantic role labeling (SRL) is one of the fundamental tasks in natural language processing because of its important role in information extraction (Bastianelli et al., 2013), statistical machine translation (Aziz et al., 2016; Xiong et al., 2012), and so on.

However, state-of-the-art performance of Chinese SRL is still far from satisfactory. And data sparsity has been a bottleneck that can not be ignored. For English, the most commonly used benchmark dataset PropBank (Xue and Palmer, 2003) has about 54,900 sentences. But for Chinese, there are only 10,364 sentences in Chinese PropBank 1.0 (CPB) (with about 35,700 propositions) (Xue, 2008).

Predicate given: 修改 (revise)

(a) [*ArgM-TMP*在这期间 (Meanwhile)], [*Arg0*全国人大常委会 (the NPC Standing Committee)] ...
广泛 (widely) 征求 (solicit) 意见 (opinions), [*ArgM-ADV*多次 (for many times)] [*ArgM-ADV*反复 (repeatedly)]
[*Rel*修改 (revise)] [*Arg1* *pro* (omitted)] 。

(b) [*agent*他们 (They)] 对 (to) [*patient*系统 (system)]进行了 (made) [*Rel*修改 (revise)] 。

Figure 1: Sentences from (**a**) CPB and (**b**) our heterogeneous dataset. In CPB, each predicate (e.g., 修改) has a specific set of core roles given with numbers (e.g., *Arg0*). While our dataset uses a different semantic role set, and all roles are non-predicate-specific.

To mitigate the data sparsity, models incorporating heterogeneous resources have been introduced to improve Chinese SRL performance (Wang et al., 2015; Guo et al., 2016; Li et al., 2016). The heterogeneous resources introduced by these models include other semantically annotated corpora with annotation schema different to that used in PropBank, and even of a different language. The challenge here lies in the fact that those newly introduced resources are heterogeneous in nature, without sharing the same tagging schema, semantic role set, syntactic tag set and domain. For example, Wang et al. (2015) introduced a heterogeneous dataset, Chinese NetBank, by pretraining word embeddings. Specifically, they learn an LSTM RNN model based on NetBank first, then initialize a new model with the pretrained embeddings obtained from Net-

Bank, and then train it on CPB. Chinese NetBank (Yulin, 2007) is also a corpus annotated with semantic roles, but using a very different roleset and annotation schema. Wang's method can inherit knowledge acquired from other resources conveniently, but only at word representation level, missing more generalized semantic meanings in higher hidden layers. Li (2016) proposed a two-pass training approach to use corpora of two languages, but a few non-common roles are ignored in the first pass. Guo et al. (2016) proposed a unified neural network model for SRL and *relation classification* (RC). It can learn two tasks at the same time, but cannot filter out harmful features learned in incompatible tasks.

Recently, *Progressive Neural Networks* (PNN) model was proposed by Rusu et al. (2016) to transfer learned reinforcement learning policies from one game to another, or from simulation to the real robot. PNN "freezes" learned parameters once starting to learn a new task, and it uses lateral connections, namely adapter, to access previously learned features.

Inspired by the PNN model, we propose a progressive learning model to Chinese semantic role labeling in this paper. Especially, we extend the model with Gated Recurrent Adapters (GRA). Since the standard PNN takes pixels as input, policies as output, it is not suitable for SRL task we focus in this paper. Moreover, to handle long sentences in the corpus, we enhance adapters with internal memories, and gates to keep the gradient stable. The contributions of this paper are threefold:

1. We reconstruct PNN columns with bidirectional LSTMs to introduce heterogeneous corpora to improve Chinese SRL. The architecture can also be applied to a wider range of NLP tasks, like event extraction and relation classification, etc.

2. We further extend the model with GRA to remember and take advantage of what has been transferred, thus improve the performance on long sentences.

3. We also release a new corpus, Chinese SemBank, which was annotated with the schema different to that used in CPB, in hope that it will be helpful for future work on SRL tasks.

We use our new corpus as a heterogeneous resource, and evaluate the proposed model on the benchmark dataset CPB 1.0. The experiment shows that our approach achieves 79.67% F1 score, significantly outperforms existing state-of-the-art systems by a large margin (Section 5).

Subjective roles: agent(施事), co-agent(同事), experiencer(当事) , indirect experiencer(接事)
Objective roles: patient(受事), relative(系事), dative(与事) , result(结果), content(内容), target(对象)
Space roles: a point of departure(起点) , a point of arrival(终点) , path(路径), direction(方向), location(处所)
Time roles: start time(起始), end time(结束), time point(时点) , duration(时段)
Comparison roles: comparison subject(比较主体), comparison object(比较对象) , comparison range(比较范围), comparison thing(比较项目) , comparison result(比较结果)
Others: instrument(工具) , material(材料) , manner(方式) , quantity(物量) , range(范围) , reason(原因) , purpose(目的)

Table 1: Semantic roles in Chinese SemBank

2 Heterogeneous Corpora for Chinese SRL

In this paper, we provide a new SRL corpus Chinese SemBank (CSB) and use it as an example of heterogeneous data in our experiments. In this section, we first briefly introduce the corpus, then compare it to existing corpora.

Sentences in CSB are from a variety of sources including online articles and news. The goal of this project is to build a very large and complete Chinese semantic corpus in the future. Currently, it only focus on the predicate-argument structures in a sentence without annotation of the temporal relations and coreference. Compared to the most commonly used dataset CPB, CSB is different in the following aspects:

- In terms of predicate, CSB takes wider range of predicates into account. We not only annotated common verbs, but also nominal verbs, as NomBank does, and state words as well. While CPB only annotate common verbs as predicates.

- In terms of semantic roles, CSB has a more fined-grained semantic role set. we defined 31 roles in five types (as Table. 1 shows). While In CPB, there are totally 23 roles, including core roles and non-core roles.

- CSB does not have any pre-defined frames for predicates because all roles are set to be non-predicate-specific. The reason for not defining frames is that frames may lead inconsistencies in labels. For example, in CPB, an *agent* of a verb is often marked as its *Arg0*, but not all *Arg0* are agents, according to Chinese verb formation theory (Sun et al., 2009). Therefore, roles are defined for predicates with similar syntactic and semantic regularities, rather than single predicate.

Two direct benefits of using stand-alone non-predicate-specific roles are: First, meanings of all semantic roles can be directly inferred from their labels. For instance, roles of things that people are telling (谈) or looking (看) are labeled as 内容/*content*, because verbs like 谈 and 看 are often followed by an object. Second, we can easily annotate sentences with new predicates without defining new frame files.

Other Corpora for Chinese SRL Other popular semantic role labeling corpora include Chinese NomBank (Xue, 2006), Peking University Chinese NetBank (Yulin, 2007). NomBank, often used as a complement to PropBank, annotates nominal predicates and semantic roles according to the similar semantic schema as PropBank does. Peking University Chinese NetBank was created by adding a semantic layer to Peking University Chinese TreeBank (Zhou et al., 1997). It only use non-predicate-specific roles as we does. But the role set is smaller, which has 20 roles.

3 Challenges in Inheriting Knowledge from Heterogeneous Corpora

Although there are lots of annotated corpora for Chinese SRL as we mentioned in the previous section, most of them are quite small compared to that of English. And *data sparsity* remains a bottleneck. This situation calls for larger training dataset, or effective approaches which can take advantage of highly heterogeneous datasets. In this paper, we focus on the second problem, that is, *to improve Chinese SRL by using heterogeneous corpora together within one model*.

We will consider the combination of the standard benchmark, CPB 1.0 dataset (Xue and Palmer, 2003), with the new corpus, CSB, because there are huge differences between them, as we discussed in Section 2. Consequently, a number of challenges arise for this task. Now we describe them as below.

Inheriting from Different Schema and Role Sets. CPB was annotated with PropBank-style frames and roles, while Chinese FrameNet uses its own frames and roles. And our dataset has no frame files and use different role set. Therefore, it is hard to find explicit mapping or hierarchical relationships among their role sets, or decide which system is better, especially when there are more than two resources.

Inheriting from Different Domain/Genre. The datasets mentioned above composed of sentences from various sources, including news and stories, etc. However, it is well known that adding data in very different genre to training data may hurts parser performance (Bikel, 2004). Therefore, we also need to deal with domain adaptation problem when using heterogeneous data. In other words, the proposed approach should be robust to harmful features learned on incompatible datasets, and can accommodate potentially different model structures and inputs in the procedure of knowledge fusion.

Inheriting from Different Syntactic Annotation. Different from English, previous works (Ding and Chang, 2009; Sun et al., 2009) on Chinese SRL task often use both correct segmentation and part-of-speech tagging, and even treebank gold-standard parses (Xue, 2008) as their features. But some corpora like CPB and NetBank do not share the same PoS tag set, or do not have correct PoS tagging and gold treebank parses at all, like CSB. And in real application scenarios, it is more convenient to use automatic PoS tagging instead of gold-standard tagging on large datasets, as they can be obtained quickly. So to deal with the absence of syntactic features, we adopt automatic PoS tagging when training on CSB in this work.

Some previous techniques, such as finetuning after pretraining (Wang et al., 2015; Li et al., 2016) and multi-task learning (Guo et al., 2016), have been used to deal with these challenges. Though they can also leverage knowledge from different

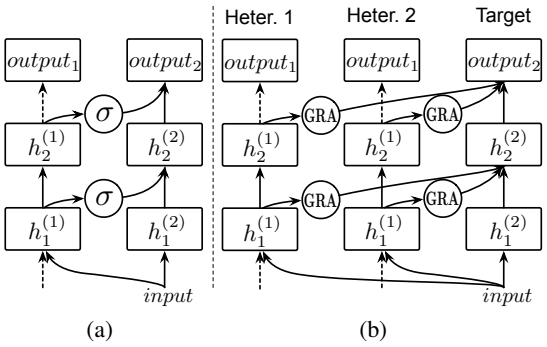


Figure 2: Depiction of the standard Progressive Neural Network architecture (a) and ours PNN GRA model (b). Our model uses Gated Recurrent Adapters (GRA), instead of sigmoid adapters to access previous knowledge in previous columns learned on heterogeneous data. If there are more than one heterogeneous resources available, more columns can be added on the left.

domains, they have following drawbacks: finetuning cannot avoid catastrophic forgetting because learned parameters, whether embeddings or other hidden weights, will be tuned after the model has been initialized; And multi-task learning cannot ignore previously learned harmful features because some features are learned in shared layers, although it avoids forgetting by randomly select a task to learn at each iteration. Therefore, to solve the above-mentioned challenges, we further introduce progressive learning which we believe is more suitable for the task.

4 Progressive Learning Approach

We propose a progressive learning approach which is ideal for combining heterogeneous SRL data for multiple reasons. First, it can accommodate dissimilar inputs with different schema, syntactic information and domain, because it allow models for heterogeneous resources to be extremely different, such as different network structures, different width, and different learning rates, etc. Second, it is immune to forgetting by freezing learned weights and can leverage prior knowledge via lateral connections. Third, the lateral connections can be extended with recurrent structure and gate mechanism to handle with forgetting problem over long distance.

Our model is mainly inspired by Rusu et al. (2016). They proposed *progressive neural networks* for a wide variety of reinforcement learning tasks (e.g. Atari games and robot simulation). In their cases, inputs are pixels, outputs are learned policies, and each column, consisting of simple layers and convolutional layers, is trained to solve a particular Markov Decision Process. But in our case, inputs are sentences annotated using different syntactic tagsets and outputs are semantic role sequences. So we change the structure of columns to recurrent neural networks with LSTM, similar to the model proposed by Wang et al. (2015). Below we first introduce basic progressive neural network architecture, then describe our model, PNN with gated recurrent adapters.

4.1 Progressive Neural Networks

Fig. 2a is an illustration of the basic progressive neural network model. It starts with single column (a neural network), in which there are L hidden layers and the output for ith layer ($i \leq L$) with n_i units is $h_i^1 \in \mathbb{R}^{n_i}$. Θ^1 denotes the parameters to be learned in the first column. When switching to a second corpus, it "freezes" the parameter Θ^1 and randomly initialize a new column with parameters Θ^2 and several lateral connections between two columns so that layer h_i^2 can receive input from both h_{i-1}^2 and h_{i-1}^1. In this straightforward manner, progressive neural networks can make use of columns with any structures or to compile lateral connections in an ensemble setting. To be more general, we calculate the output of ith layer in kth column h_i^k by:

$$h_i^k = f(W_i^k h_{i-1}^k + \sum_{j<k} U_i^{(k:j)} h_{i-1}^j) \qquad (1)$$

where $W_i^k \in \mathbb{R}^{n_i^k \times n_{i-1}^k}$ is the weight matrix of layer i of column k, $U_i^{(k:j)} \in \mathbb{R}^{n_i^k \times n_{i-1}^j}$ are the lateral connections to transfer information from layer $i-1$ of column j to layer i of column k, h_0 is the input of the network. f can be any activation function, such as element-wise non-linearity. Bias term was omitted in the equation.

Adapters. With implicit assumption that there is some "overlap" between the first task and the second task, pretrain-and-finetune learning paradigm is effective, as only slight adjustment to parameters is needed to learn new features. Progressive networks also have ability to transfer knowledge from previous tasks to improve convergence speed. On the one hand, the model reuse previously learned features from left columns via lateral connections (i.e., *adapters*). On the other

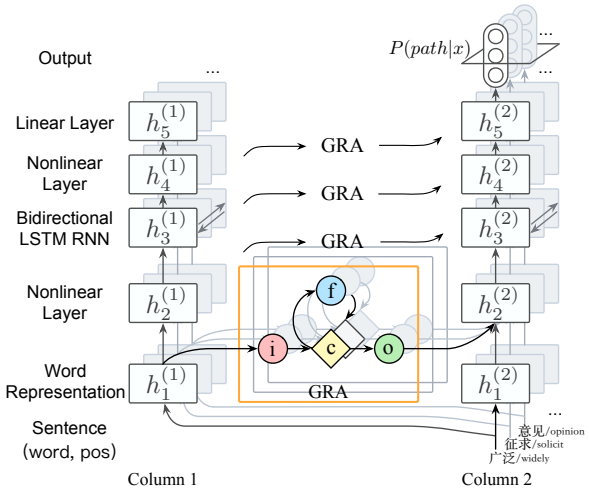


Figure 3: Each column is a stacked bidirectional LSTM RNN model. Two columns are connected by GRAs. There are three gates in each GRA: g_i, g_f, and g_o. The input gate g_i and the forget gate g_f can also be coupled as one uniform gate, that is $g_i = 1 - g_f$.

hand, new features can be learned by adding more columns incrementally. Moreover, when the "overlap" between two tasks is small, lateral connections can filter out harmful features by sigmoid functions. So in practice, the output of adapters can also be calculated by

$$a_i^{(k:j)} = \sigma(A_i^{(k:j)} \alpha_{i-1}^j h_{i-1}^j) \quad (2)$$

where $A_i^{(k:j)}$ is a matrix to be learned. We treat Equation 2 as one of baseline settings in experiments.

4.2 PNN with Gated Recurrent Adapter for Chinese SRL

We reconstruct PNN with bidirectional LSTM to solve SRL problems. Our model is illustrated in Fig. 3.

First, each column in the PNN architecture is a stacked bidirectional LSTM RNN, rather than convolutional neural networks, because inputs are sentences not pixels, and bi-LSTM RNN has proved powerful for Chinese SRL (Wang et al., 2015).

Second, we enhance the adapter with recurrent structure and gate mechanism, because the simple Multi-Layer Perceptron (MLP) adapters has a limitation: their weights are learned word after word independently. For tasks like transferring reinforcement learning policies, this is enough because there are little dependencies among actions. But in NLP domain, things are different. Therefore, we add internal memory to adapters to help them remember what has been inherited from heterogeneous resource.

Third, to keep gradient stable and balance between long-term and short-term memory, we introduce gate mechanism which has been widely used in RNN models. Intuitively, we call the new adapter *Gated Recurrent Adapter* (GRA).

Formally, let $h_{i-1}^{(<k)} = [h_{i-1}^1, ..., h_{i-1}^j, ..., h_{i-1}^{k-1}]$ be the outputs of $i-1$ layers from the first column to the $(k-1)$th column. The dimensionality of them is $n_{i-1}^{(<k)} = [n_{i-1}^1, ..., n_{i-1}^{k-1}]$. $a^{(<k)}$ is the outputs of $k-1$ adapters with dimension $m^{(<k)} = [m^1, ..., m^{k-1}]$. The output vector is multiplied by a learned matrix W_a initialized by random small values before going to GRAs. Its role is to adjust for the different scales of the different inputs and reduce the dimensionality. Formally, the candidate outputs is

$$\hat{a}_t = f(W_a^j h_t^j + U_a^j a_{t-1}^j) \quad (3)$$

where a_{t-1} is the output of the adapter at the previous time-step. U_a is a weight matrix to learn. The output of an adapter a_t^j of layer i at time t can be formalized as follows,

$$g_i = \sigma(W_i^j h_t^j + U_i^j a_{t-1}^j) \quad (4)$$
$$g_f = \sigma(W_f^j h_t^j + U_f^j a_{t-1}^j) \quad (5)$$
$$g_o = \sigma(W_o^j h_t^j + U_o^j a_{t-1}^j) \quad (6)$$
$$\tilde{a}_t = g_i \odot \hat{a}_t + g_f \odot \tilde{a}_{t-1}^j \quad (7)$$
$$a_t = g_o \odot f(\tilde{a}_{t-1}) \quad (8)$$

where $h^j \in \mathbb{R}^{m_{i-1}^j \times n_{i-1}^j}$ is the outputs of previous layers, $W_f, W_o, W_a \in \mathbb{R}^{m_{i-1} \times n_{i-1}}$, $U_f, U_o, U_a \in \mathbb{R}^{m_{i-1} \times d_{i-1}}$. d_{i-1} is the dimension of the inner memory in adapters. $\tilde{a}_t$ represents the inner state of the adapter. f is an activation function, like *tanh*. The input gate and the forget gate can be coupled as a uniform gate, that is $g_i = 1 - g_f$ to alleviate the problem of information redundancy and reduce the possibility of overfitting (Greff et al., 2015).

Finally, we calculate the output of the next layer i of column k by

$$h_i^k = f(W_i^k concat[a^{(<k)}, h_{i-1}^k]) \quad (9)$$

where $W_i \in \mathbb{R}^{n_i^{(k)} \times \sum m_{i-1}^{(<k)}}$ is the parameters in ith layer.

4.3 Training Criteria

We adopt the sentence tagging approach as Wang et al. (2015) did, because words in a sentence may closely related with each other, independently labeling each word is inappropriate. Sentence tagging approach only consider valid transition paths of tags when calculating the cost. For example, when using IOBES tagging schema, tag transition from *I-Arg0* to *B-Arg0* is invalid, and transition from *I-Arg0* to *I-Arg1* is also invalid because the type of the role changed *inside* the semantic chunk. For each task (column), the log likelihood of sentence x and its correct path y is

$$\log p(y|x,\Theta) = \log \frac{\exp \sum_t^N o_{t,y_t}}{\sum_z \exp \sum_t^{N_i} o_{t,z_t}} \quad (10)$$

where N is the number of words, $o_t \in \mathbb{R}^M$ is the output of the last layer at time t. $y_t = k$ means the tth word has the kth semantic role label. z ranges from all the valid paths of tags.

The negative log likelihood of the whole training set D is

$$J(\Theta) = \sum_{(x,y)\in D} \log p(y|x,\Theta) \quad (11)$$

We minimize $J(\Theta)$ using stochastic gradient descent to learn network parameters Θ. When testing, the best prediction of a sentence can be found using Viterbi algorithm.

5 Experiments

5.1 Experiment Settings

To compare our approach with others, we designed four experimental setups:

(1) A simple LSTM setup on CSB and CPB with automatic PoS tagging. Since CPB is about two times as large as the new corpus, we need to know whether CSB can be used for training good semantic parsers and how much information can be learned from CSB by machine. So we conduct this experiment to provide two baselines for CSB and CPB respectively. In this setup we train and evaluate a one-column LSTM model on CSB.

(2) A simple LSTM setup on CPB with pretrained word embedding on CSB (marked as bi-LSTM+CSB embedding). Previous work found that using pretrained word embeddings can improve performance (Wang et al., 2015) on Chinese SRL. So we conduct this experiment to compare with the method using embeddings trained on large-scale unlabeled data like Gigaword [1], and NetBank.

(3) A two-column finetuning setup where we pretrain the first column on CSB and finetune both two columns on CPB. Clearly, finetuning is a traditional method for continual learning scenarios. But the disadvantage of it is that learned features will be gradually forgotten when the model is adapting to new tasks. To assess this empirically, we design this experiment. The model use the same network structure as PNN does, but it does not "freeze" parameters in the first column when tuning two columns.

(4) A progressive network setup where we train column 1 on CSB, then train column 2 and adapters on CPB. We conduct this experiment to evaluate the proposed model and compare it to all previous methods. To further analyze effectiveness of the new adapter structure, we also conduct an experiment for progressive nets with GRA.

We apply grid-search technique to explore hyper-parameters including learning rates and width of layers.

Preprocessing. We follow the same data setting as previous work (Xue, 2008; Sun et al., 2009), which divided CPB dataset[2] into three parts: 648 files, from `chtb_081.fid` to `chtb_899.fid`, are the training set; 40 files, from `chtb_041.fid` to `chtb_080.fid`, are the development set; 72 files, from `chtb_001.fid` to `chtb_040.fid`, and `chtb_900.fid` to `chtb_931.fid`, are used as the test set. We also divide shuffled CSB corpus into three sets with similar partition ratios: 8900 samples are used as training set, 500 samples as development set and the rest 965 samples as test set. We use Stanford Parser[3] for PoS tagging.

5.2 Results

Performance on Chinese SemBank Table 2 gives the results of Experiment 1. We see that precision on CPB with automatic PoS tagging is about 0.9 percentage point higher than that on CSB, while recall is about 0.4 percentage point lower, and the gap between F1 scores on CPB and CSB is not significant, which is only about 0.3 percentage point, although the size of CSB is smaller. We can explain this by two reasons. First, CSB

[1] https://code.google.com/p/word2vec/
[2] https://catalog.ldc.upenn.edu/LDC2005T23
[3] http://nlp.stanford.edu/software/lex-parser.shtml

Corpus	Pr.(%)	Rec.(%)	F1(%)
1. CSB	75.80	73.45	74.61
2. CPB	76.75	73.03	74.84

Table 2: Results of Chinese SRL tested on CPB and CSB with automatic PoS tagging, using standard LSTM RNN model (Experiment 1).

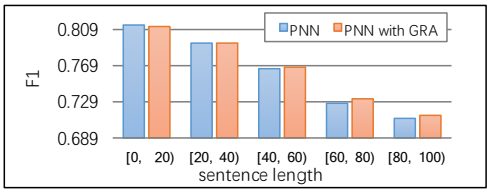


Figure 4: Performance of PNN models with and without GRAs over sentence length. For sentences shorter than 40 words, there is no big difference. But for longer sentences (≥40 words), PNN with GRA model performs significantly better.

does not have predicate-specific roles which may leads to inconsistency, as we explained in Section 3. Thus, it might be easier to learn by machine. Second, there are underlying similarities between them: both of them annotate predicate-argument structures. So when there are sufficient training data, difference between scores on testing sets is not very likely to be huge.

Overall, the results indicated that the new annotated corpus CSB is not a bad choice for training semantic parser even when this does not involve larger training sets.

Compare to Methods without Using Heterogeneous Data Table 3 summarizes the SRL performance of previous benchmark methods and our experiments described above. Collobert and Weston only conducted their experiments on English corpus, but we notice that their approach has been implemented and tested on CPB by Wang et al. (2015), so we also put their result here for comparison. We can make several observations from these results. Our approach significantly outperforms Sha et al. (2016) by a large margin (Wilcoxon Signed Rank Test, $p < 0.05$), even without using GRA. This result can prove the ability of our model to capture underlying similarities between heterogeneous SRL resources.

Compare to Methods Using Heterogeneous Resources The results of methods using external language resources are also presented in Table 3. Not surprisingly, we see that the overall best F1 score, 79.67%, is achieved by the progressive nets with the GRAs. Furthermore, as shown in Fig. 4, PNN with GRA performs better on longer sentences, which is consistent with our expectation. Without GRA, the F1 drops 0.37% percentage point to 79.30, confirming that gated recurrent adapter structure is more suitable for our task because it can remember what has been transferred in previous time steps.

Compared to progressive learning methods, finetuning method does not perform well even with the same network structure (Two-column finetuning), but it is still better than simply pretraining word embeddings (bi-LSTM+CSB embedding). This confirms the effectiveness of multi-column learning structure which add capacity to the model by adding new columns. Therefore, as can be seen, our PNN model achieves 79.30% F1 score, outperforming finetuning by 0.88% percentage point, and pretraining embeddings by even larger margin.

To sum up, not only network structures but also learning methods (finetuning/multitask/progressive) can influence the performance of knowledge transfer. According to the results, our PNN approach is more effective than others because it is immune to forgetting and robust to harmful features, and GRA is more suitable for our task than simple adapters.

6 Related Work

6.1 Chinese Semantic Role Labeling

The concept of Semantic Role Labeling is first proposed by Gildea and Jurafsky(2002). Previous work on Chinese SRL mainly focused on how to improve SRL on single corpus. Approaches falls into two categories: feature-based machine learning approaches and neural-network-based approaches. Using feature-based method, Sun and Jurafsky (2004) did the preliminary work and achieved promising results without using any large annotated corpus. After CPB was built by Xue and Palmer (2003), more complete and systematic research on Chinese SRL were done (Xue and Palmer, 2005; Chen et al., 2006; Ding and Chang, 2009; Yang et al., 2014).

Neural network methods not rely on hand-

Method	F1(%)
Xue (2008) ME	71.90
Collobert and Weston (2008) MTL	74.05
Ding and Chang (2009) CRF	72.64
Yang et al. (2014) Multi-Predicate	75.31
Wang et al. (2015) bi-LSTM	77.09 (+0.00)
Sha et al. (2016) bi-LSTM+QOM	**77.69**
With external language resources	
Wang et al. (2015) +Gigaword embedding	77.21
Wang et al. (2015) +NetBank embedding	**77.59**
Guo et al. (2016) +Relataion Classification	75.46
With CSB corpus	
bi-LSTM+CSB embedding	77.68 (+0.59)
Two-column finetuning	78.42 (+1.33)
Tow-column progressive(ours)	79.30 (+2.21)
Tow-column Progressive+GRA(ours)	**79.67 (+2.58)**

Table 3: Result comparison on CPB dataset. Compared to learning with single corpus using bi-LSTM model (77.09%), learning with CSB can improve the performance by at list 0.59%. And the best score (79.67%) was achieved by the PNN GRA model.

crafted features. For Chinese SRL, Wang et al. (2015) proposed bidirectional a LSTM RNN model. And based on their work, Sha (2016) proposed quadratic optimization method as a post-processing module and further improved the result.

6.2 Learning with Heterogeneous Data

In this paper, we mainly focus on learning with heterogeneous semantic resource for Chinese SRL. Wang et al. (2015) introduced heterogeneous data by using pretrained embeddings at initialization and achieved promising results. Guo et al. (2016) proposed a multitask learning method with a unified neural network model to learn SRL and relation classification task together and also achieved improvement.

Different from previous work, we proposed a progressive neural network model with gated recurrent adapters to leverage knowledge from heterogeneous semantic data. Compare with previous methods, this approach is more constructive, rather than destructive, because it uses lateral connections to access previously learned features which are fixed when learning new tasks. And by introducing gated recurrent adapters, we further enhance our model to deal with long sentences and achieve state-of-the-art performance on Chinese PropBank.

7 Conclusion and Future Work

In this paper, we proposed a progressive neural network model with gated recurrent adapters to leverage heterogeneous corpus for Chinese SRL. Unlike previous methods like finetuning, ours leverage prior knowledge via lateral connections. Experiments have shown that our model yields better performance on CPB than all baseline models. Moreover, we proposed novel gated recurrent adapter to handle transfer on long sentences, The experiment has proved the effectiveness of the new adapter structure.

We believe that progressive learning with heterogeneous data is a promising avenue to pursue. So in the future, we might try to combine more heterogeneous semantic data for other tasks like event extraction and relation classification, etc.

We also release the new corpus Chinese SemBank[4], for Chinese SRL in the hope that it will be helpful in providing common benchmarks for future work on Chinese SRL tasks.

References

Wilker Aziz, Miguel Rios, and Lucia Specia. 2016. Shallow semantic trees for smt. In *Proc. of the 6th Workshop on Statistical Machine Translation*. Edinburgh, Scotland, pages 316–322.

[4]We will provide the address on publication.

Emanuele Bastianelli, Giuseppe Castellucci, Danilo Croce, and Roberto Basili. 2013. Textual inference and meaning representation in human robot interaction. In *In Proceedings of the Joint Symposium on Semantic Processing. Textual Inference and Structures in Corpora*. pages 65–69.

Daniel M Bikel. 2004. *On the parameter space of generative lexicalized statistical parsing models*. Ph.D. thesis, Citeseer.

Wenliang Chen, Yujie Zhang, and Hitoshi Isahara. 2006. An empirical study of chinese chunking. In *Proceedings of the COLING/ACL on Main conference poster sessions*. Association for Computational Linguistics, pages 97–104.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*. ACM, pages 160–167.

Weiwei Ding and Baobao Chang. 2009. Word based chinese semantic role labeling with semantic chunking. *International Journal of Computer Processing Of Languages* 22(02n03):133–154.

Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Computational linguistics* 28(3):245–288.

Klaus Greff, Rupesh Kumar Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber. 2015. Lstm: A search space odyssey. *arXiv preprint arXiv:1503.04069* .

Jiang Guo, Wanxiang Che, Haifeng Wang, Ting Liu, and Jun Xu. 2016. A unified architecture for semantic role labeling and relation classification. In *Proc. of the 26th International Conference on Computational Linguistics (COLING)*.

Tianshi Li, Qi Li, and BaoBao Chang. 2016. Improving chinese semantic role labeling with english proposition bank. In *China National Conference on Chinese Computational Linguistics*. Springer, pages 3–11.

Andrei A. Rusu, Neil C. Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. 2016. Progressive neural networks. *CoRR* abs/1606.04671.

Lei Sha, Tingsong Jiang, Sujian Li, Baobao Chang, and Zhifang Sui. 2016. Capturing argument relationships for chinese semantic role labeling .

Honglin Sun and Daniel Jurafsky. 2004. Shallow semantic parsing of chinese. In *Proceedings of NAACL 2004*. pages 249–256.

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

Zhen Wang, Tingsong Jiang, Baobao Chang, and Zhifang Sui. 2015. Chinese semantic role labeling with bidirectional recurrent neural networks. In *Proc. of the 2015 Conference on Empirical Methods in Natural Language Processing*. pages 1626–1631.

Deyi Xiong, Min Zhang, and Haizhou Li. 2012. Modeling the translation of predicate-argument structure for smt. In *In Proc. of the 50th Annual Meeting of the Association for Computational Linguistics*. pages 902–911.

Nianwen Xue. 2006. Annotating the predicate-argument structure of chinese nominalizations. In *Proceedings of the fifth international conference on Language Resources and Evaluation*. pages 1382–1387.

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

Nianwen Xue and Martha Palmer. 2003. Annotating the propositions in the penn chinese treebank. In *Proceedings of the second SIGHAN workshop on Chinese language processing-Volume 17*. Association for Computational Linguistics, pages 47–54.

Nianwen Xue and Martha Palmer. 2005. Automatic semantic role labeling for chinese verbs. In *In Proceedings of the 19th International Joint Conference on Artificial Intelligence*. pages 1160–1165.

Haitong Yang, Chengqing Zong, et al. 2014. Multi-predicate semantic role labeling. In *EMNLP*. pages 363–373.

Yuan Yulin. 2007. The fineness hierarchy of semantic roles and its application in nlp. *Journal of Chinese Information Processing* 21(4):10–20.

Qiang Zhou, Wei Zhang, and Shiwen Yu. 1997. Building a chinese treebank. *Journal of Chinese Information Processing* 4.