A Bi-model based RNN Semantic Frame Parsing Model for Intent Detection and Slot Filling

Yu Wang **Yilin Shen** **Hongxia Jin**
Samsung Research America
{yu.wang1, yilin.shen, hongxia.jin}@samsung.com

Abstract

Intent detection and slot filling are two main tasks for building a spoken language understanding(SLU) system. Multiple deep learning based models have demonstrated good results on these tasks . The most effective algorithms are based on the structures of sequence to sequence models (or "encoder-decoder" models), and generate the intents and semantic tags either using separate models((Yao et al., 2014; Mesnil et al., 2015; Peng and Yao, 2015; Kurata et al., 2016; Hahn et al., 2011)) or a joint model ((Liu and Lane, 2016a; Hakkani-Tür et al., 2016; Guo et al., 2014)). Most of the previous studies, however, either treat the intent detection and slot filling as two separate parallel tasks, or use a sequence to sequence model to generate both semantic tags and intent. Most of these approaches use one (joint) NN based model (including encoder-decoder structure) to model two tasks, hence may not fully take advantage of the cross-impact between them. In this paper, new Bi-model based RNN semantic frame parsing network structures are designed to perform the intent detection and slot filling tasks jointly, by considering their cross-impact to each other using two correlated bidirectional LSTMs (BLSTM). Our Bi-model structure with a decoder achieves state-of-the-art result on the benchmark ATIS data (Hemphill et al., 1990; Tur et al., 2010), with about 0.5% intent accuracy improvement and 0.9 % slot filling improvement.

1 Introduction

The research on spoken language understanding (SLU) system has progressed extremely fast during the past decades. Two important tasks in an SLU system are intent detection and slot filling. These two tasks are normally considered as parallel tasks but may have cross-impact on each other. The intent detection is treated as an utterance classification problem, which can be modeled using conventional classifiers including regression, support vector machines (SVMs) or even deep neural networks (Haffner et al., 2003; Sarikaya et al., 2011). The slot filling task can be formulated as a sequence labeling problem, and the most popular approaches with good performances are using conditional random fields (CRFs) and recurrent neural networks (RNN) as recent works (Xu and Sarikaya, 2013).

Some works also suggested using one joint RNN model for generating results of the two tasks together, by taking advantage of the sequence to sequence(Sutskever et al., 2014) (or encoder-decoder) model, which also gives decent results as in literature(Liu and Lane, 2016a).

In this paper, Bi-model based RNN structures are proposed to take the cross-impact between two tasks into account, hence can further improve the performance of modeling an SLU system. These models can generate the intent and semantic tags concurrently for each utterance. In our Bi-model structures, two task-networks are built for the purpose of intent detection and slot filling. Each task-network includes one BLSTM with or without a LSTM decoder (Hochreiter and Schmidhuber, 1997; Graves and Schmidhuber, 2005).

The paper is organized as following: In section 2, a brief overview of existing deep learning approaches for intent detection and slot fillings are given. The new proposed Bi-model based RNN approach will be illustrated in detail in section 3. In section 4, two experiments on different datasets will be given. One is performed on the ATIS benchmark dataset, in order to demonstrate a state-of-the-art result for both semantic parsing tasks. The other experiment is tested on our internal multi-domain dataset by comparing our new algorithm with the current best performed RNN based joint model in literature for intent detection and slot filling.

2 Background

In this section, a brief background overview on using deep learning and RNN based approaches to perform intent detection and slot filling tasks is given. The joint model algorithm is also discussed for further comparison purpose.

2.1 Deep neural network for intent detection

Using deep neural networks for intent detection is similar to a standard classification problem, the only difference is that this classifier is trained under a specific domain. For example, all data in ATIS dataset is under the flight reservation domain with 18 different intent labels. There are mainly two types of models that can be used: one is a feed-forward model by taking the average of all words' vectors in an utterance as its input, the other way is by using the recurrent neural network which can take each word in an utterance as a vector one by one (Xu and Sarikaya, 2014).

2.2 Recurrent Neural network for slot filling

The slot filling task is a bit different from intent detection as there are multiple outputs for the task, hence only RNN model is a feasible approach for this scenario. The most straight-forward way is using single RNN model generating multiple semanctic tags sequentially by reading in each word one by one (Liu and Lane, 2015; Mesnil et al., 2015; Peng and Yao, 2015). This approach has a constrain that the number of slot tags generated should be the same as that of words in an utterance. Another way to overcome this limitation is by using an encoder-decoder model containing two RNN models as an encoder for input and a decoder for output (Liu and Lane, 2016a). The advantage of doing this is that it gives the system capability of matching an input utterance and output slot tags with different lengths without the need of alignment. Besides using RNN, It is also possible to use the convolutional neural network (CNN) together with a conditional random field (CRF) to achieve slot filling task (Xu and Sarikaya, 2013).

2.3 Joint model for two tasks

It is also possible to use one joint model for intent detection and slot filling (Guo et al., 2014; Liu and Lane, 2016a,b; Zhang and Wang, 2016; Hakkani-Tür et al., 2016). One way is by using one encoder with two decoders, the first decoder will generate sequential semantic tags and the second decoder generates the intent. Another approach is by consolidating the hidden states information from an RNN slot filling model, then generates its intent using an attention model (Liu and Lane, 2016a). Both of the two approaches demonstrates very good results on ATIS dataset.

3 Bi-model RNN structures for joint semantic frame parsing

Despite the success of RNN based sequence to sequence (or encoder-decoder) model on both tasks, most of the approaches in literature still use one single RNN model for each task or both tasks. They treat the intent detection and slot filling as two separate tasks. In this section, two new Bi-model structures are proposed to take their cross-impact into account, hence further improve their performance. One structure takes the advantage of a decoder structure and the other doesn't. An asynchronous training approach based on two models' cost functions is designed to adapt to these new structures.

3.1 Bi-model RNN Structures

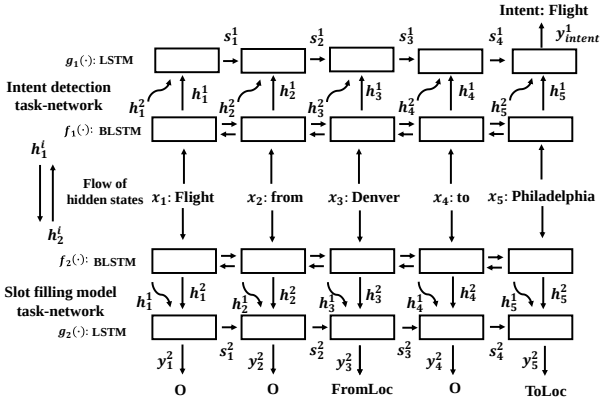
A graphical illustration of two Bi-model structures with and without a decoder is shown in Figure 1. The two structures are quite similar to each other except that Figure 1a contains a LSTM based decoder, hence there is an extra decoder state s_t to be cascaded besides the encoder state h_t.

Remarks:

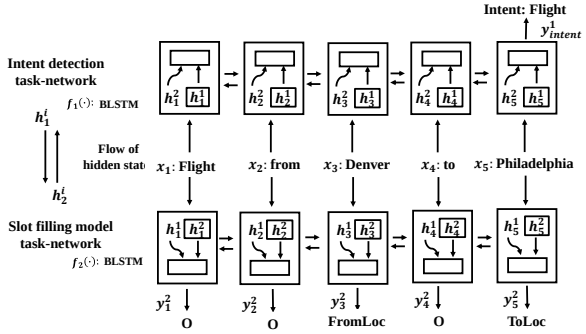
The concept of using information from multiple-model/multi-modal to achieve better performance has been widely used in deep learning (Dean et al., 2012; Wang, 2017; Ngiam et al., 2011; Srivastava and Salakhutdinov, 2012), system identification (Murray-Smith and Johansen, 1997; Narendra et al., 2014, 2015) and also reinforcement learning field recently (Narendra et al., 2016; Wang and Jin, 2018). Instead of using collective information, in this paper, our work introduces a totally new approach of training multiple neural networks asynchronously by sharing their internal state information.

3.1.1 Bi-model structure with a decoder

The Bi-model structure with a decoder is shown as in Figure 1a. There are two inter-connected bidirectional LSTMs (BLSTMs) in the structure, one is for intent detection and the other is for slot filling. Each BLSTM reads in the input utterance sequences $(x_1, x_2, \cdots, x_n)$ forward and backward, and generates two sequences of hidden states hf_t and hb_t. A concatenation of hf_t and hb_t forms a final BLSTM state $h_t = [hf_t, hb_t]$ at time step t. Hence, Our bidirectional LSTM $f_i(\cdot)$ generates a



(a) Bi-model structure with a decoder



(b) Bi-model structure without a decoder

Figure 1: Bi-model structure

sequence of hidden states $(h_1^i, h_2^i, \cdots, h_n^i)$, where $i = 1$ corresponds the network for intent detection task and $i = 2$ is for the slot filling task.

In order to detect intent, hidden state h_t^1 is combined together with h_t^2 from the other bidirectional LSTM $f_2(\cdot)$ in slot filling task-network to generate the state of $g_1(\cdot)$, s_t^1, at time step t:

$$\begin{aligned} s_t^1 &= \phi(s_{t-1}^1, h_{n-1}^1, h_{n-1}^2) \\ y_{intent}^1 &= \arg\max_{\hat{y}_n^1} P(\hat{y}_n^1 | s_{n-1}^1, h_{n-1}^1, h_{n-1}^2) \end{aligned} \quad (1)$$

where $\hat{y}_n^1$ contains the predicted probabilities for all intent labels at the last time step n.

For the slot filling task, a similar network structure is constructed with a BLSTM $f_2(\cdot)$ and a LSTM $g_2(\cdot)$. $f_2(\cdot)$ is the same as $f_1(\cdot)$, by reading in the a word sequence as its input. The difference is that there will be an output y_t^2 at each time step t for $g_2(\cdot)$, as it is a sequence labeling problem. At each step t:

$$\begin{aligned} s_t^2 &= \psi(h_{t-1}^2, h_{t-1}^1, s_{t-1}^2, y_{t-1}^2) \\ y_t^2 &= \arg\max_{\hat{y}_t^2} P(\hat{y}_t^2 | h_{t-1}^1, h_{t-1}^2, s_{t-1}^2, y_{t-1}^2) \end{aligned} \quad (2)$$

where y_t^2 is the predicted semantic tags at time step t.

3.1.2 Bi-Model structure without a decoder

The Bi-model structure without a decoder is shown as in Figure 1b. In this model, there is no LSTM decoder as in the previous model.

For the intent task, only one predicted output label y_{intent}^1 is generated from BLSTM $f_1(\cdot)$ at the last time step n, where n is the length of the utterance. Similarly, the state value h_t^1 and output intent label are generated as:

$$\begin{aligned} h_t^1 &= \phi(h_{t-1}^1, h_{t-1}^2) \\ y_{intent}^1 &= \arg\max_{\hat{y}_n^1} P(\hat{y}_n^1 | h_{n-1}^1, h_{n-1}^2) \end{aligned} \quad (3)$$

For the slot filling task, the basic structure of BLSTM $f_2(\cdot)$ is similar to that for the intent detection task $f_1(\cdot)$, except that there is one slot tag label y_t^2 generated at each time step t. It also takes the hidden state from two BLSTMs $f_1(\cdot)$ and $f_2(\cdot)$, *i.e.* h_{t-1}^1 and h_{t-1}^2, plus the output tag y_{t-1}^2 together to generate its next state value h_t^2 and also the slot tag y_t^2. To represent this as a function mathematically:

$$\begin{aligned} h_t^2 &= \psi(h_{t-1}^2, h_{t-1}^1, y_{t-1}^2) \\ y_t^2 &= \arg\max_{\hat{y}_t^2} P(\hat{y}_t^2 | h_{t-1}^1, h_{t-1}^2, y_{t-1}^2) \end{aligned} \quad (4)$$

3.1.3 Asynchronous training

One of the major differences in the Bi-model structure is its asynchronous training, which trains two task-networks based on their own cost functions in an asynchronous manner. The loss function for intent detection task-network is $\mathcal{L}_1$, and for slot filling is $\mathcal{L}_2$. $\mathcal{L}_1$ and $\mathcal{L}_2$ are defined using cross entropy as:

$$\mathcal{L}_1 \triangleq -\sum_{i=1}^{k} \hat{y}_{intent}^{1,i} \log(y_{intent}^{1,i}) \quad (5)$$

and

$$\mathcal{L}_2 \triangleq -\sum_{j=1}^{n} \sum_{i=1}^{m} \hat{y}_j^{2,i} \log(y_j^{2,i}) \quad (6)$$

where k is the number of intent label types, m is the number of semantic tag types and n is the number of words in a word sequence. In each training iteration, both intent detection and slot filling networks will generate a groups of hidden states h^1 and h^2 from the models in previous iteration. The intent detection task-network reads in a batch of input data x_i and hidden states h^2, and generates the estimated intent labels $\hat{y}_{intent}^1$. The intent detection task-network computes its cost based on

function $\mathcal{L}_1$ and trained on that. Then the same batch of data x_i will be fed into the slot filling task-network together with the hidden state h^1 from intent task-network, and further generates a batch of outputs y_i^2 for each time step. Its cost value is then computed based on cost function $\mathcal{L}_2$, and further trained on that.

The reason of using asynchronous training approach is because of the importance of keeping two separate cost functions for different tasks. Doing this has two main advantages:

1. It filters the negative impact between two tasks in comparison to using only one joint model, by capturing more useful information and overcoming the structural limitation of one model.
2. The cross-impact between two tasks can only be learned by sharing hidden states of two models, which are trained using two cost functions separately.

4 Experiments

In this section, our new proposed Bi-model structures are trained and tested on two datasets, one is the public ATIS dataset (Hemphill et al., 1990) containing audio recordings of flight reservations, and the other is our self-collected datset in three different domains: Food, Home and Movie. The ATIS dataset used in this paper follows the same format as in (Liu and Lane, 2015; Mesnil et al., 2015; Xu and Sarikaya, 2013; Liu and Lane, 2016a). The training set contains 4978 utterance and the test set contains 893 utterance, with a total of 18 intent classes and 127 slot labels. The number of data for our self-collected dataset will be given in the corresponding experiment sections with a more detailed explanation. The performance is evaluated based on the classification accuracy for intent detection task and F1-score for slot filling task.

4.1 Training Setup

The layer sizes for both the LSTM and BLSTM networks in our model are chosen as 200. Based on the size of our dataset, the number of hidden layers is chosen as 2 and Adam optimization is used as in (Kingma and Ba, 2014). The size of word embedding is 300, which are initialized randomly at the beginning of experiment.

4.2 Performance on the ATIS dataset

Our first experiment is conducted on the ATIS benchmark dataset, and compared with the current existing approaches, by evaluating their intent detection accuracy and slot filling F1 scores. A detailed comparison is given in Table 1. Some of the models are designed for single slot filling task, hence only F1 scores are given. It can be observed that the new proposed Bi-model structures outperform the current state-of-the-art results on both intent detection and slot filling tasks, and the Bi-model with a decoder also outperform that without a decoder on our ATIS dataset. The current Bi-model with a decoder shows the state-of-the-art performance on ATIS benchmark dataset with 0.9% improvement on F1 score and 0.5% improvement on intent accuracy.

Table 1: Performance of Different Models on ATIS Dataset

Model	F1 Score	Intent Accuracy
Recursive NN (Guo et al., 2014)	93.96%	95.4%
Joint model with recurrent intent and slot label context (Liu and Lane, 2016b)	94.47%	98.43%
Joint model with recurrent slot label context (Liu and Lane, 2016b)	94.64%	98.21%
RNN with Label Sampling (Liu and Lane, 2015)	94.89%	NA
Hybrid RNN (Mesnil et al., 2015)	95.06%	NA
RNN-EM (Peng and Yao, 2015)	95.25%	NA
CNN CRF (Xu and Sarikaya, 2013)	95.35%	NA
Encoder-labeler Deep LSTM (Kurata et al., 2016)	95.66%	NA
Joint GRU Model (W) (Zhang and Wang, 2016)	95.49%	98.10%
Attention Encoder-Decoder NN (Liu and Lane, 2016a)	95.87%	98.43%
Attention BiRNN (Liu and Lane, 2016a)	95.98%	98.21%
Bi-model without a decoder	**96.65%**	**98.76%**
Bi-model with a decoder	**96.89%**	**98.99%**

Remarks:

1. It is worth noticing that the complexities of encoder-decoder based models are normally higher than the models without using encoder-decoder structures, since two networks are used and more parameters need to be updated. This is another reason why we use two models with/without using encoder-decoder structures to demonstrate the new bi-model structure design. It can also be observed that the model with a decoder gives a better result due to its higher complexity.

2. It is also shown in the table that the joint model in (Liu and Lane, 2015, 2016a) achieves better performance on intent detection task with slight degradation on slot filling, so a joint model is not necessary always better for both tasks. The bi-model approach overcomes this issue by generating two tasks' results separately.

3. Despite the absolute improvement of intent

accuracy and F1 scores are only 0.5% and 0.9% on ATIS dataset, the relative improvement is not small. For intent accuracy, the number of wrongly classified utterances in test dataset reduced from 14 to 9, which gives us the 35.7% relative improvement on intent accuracy. Similarly, the relative improvement on F1 score is 22.63%.

4.3 Performance on multi-domain data

In this experiment, the Bi-model structures are further tested on an internal collected dataset from our users in three domains: food, home and movie. There are 3 intents for each domain, 15 semantic tags in food domain, 16 semantic tags in home domain, 14 semantic tags in movie domain. The data size of each domain is listed as in Table 2, and the split is 70% for training, 10% for validation and 20% for test.

Due to the space limitation, only the best performed semantic frame parsing model on ATIS dataset in literature,i.e. attention based BiRNN (Liu and Lane, 2016a) is used for comparison with our Bi-model structures. Table 2 shows a performance comparison in three domains of data. The Bi-model structure with a decoder gives the best performance in all cases based on its intent accuracy and slot filling F1 score. The intent accuracy has at least 0.5% improvement, the F1 score improvement is around 1% to 3% for different domains.

Table 2: Performance Comparison between Bi-model Structures and Attention BiRNN

Domain	SLU model	Size	F1 Score	Accuracy
Movie	Attention BiRNN	979	92.1%	92.86%
	Bi-model without a decoder	979	93.3%	94.89%
	Bi-model with a decoder	979	**93.8%**	**95.91%**
Food	Attention BiRNN	983	92.3%	98.48%
	Bi-model without a decoder	983	93.6%	98.98%
	Bi-model with a decoder	983	**95.8%**	**99.49%**
Home	Attention BiRNN	689	96.5%	97.83%
	Bi-model without a decoder	689	97.8%	98.55%
	Bi-model with a decoder	689	**98.2%**	**99.27%**

5 Conclusion

In this paper, a novel Bi-model based RNN semantic frame parsing model for intent detection and slot filling is proposed and tested. Two substructures are discussed with the help of a decoder or not. The Bi-model structures achieve state-of-the-art performance for both intent detection and slot filling on ATIS benchmark data, and also surpass the previous best SLU model on the multi-domain data. The Bi-model based RNN structure with a decoder also outperforms the Bi-model structure without a decoder on both ATIS and multi-domain data.

References

Jeffrey Dean, Greg Corrado, Rajat Monga, Kai Chen, Matthieu Devin, Mark Mao, Andrew Senior, Paul Tucker, Ke Yang, Quoc V Le, et al. 2012. Large scale distributed deep networks. In *Advances in neural information processing systems*. pages 1223–1231.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks* 18(5):602–610.

Daniel Guo, Gokhan Tur, Wen-tau Yih, and Geoffrey Zweig. 2014. Joint semantic utterance classification and slot filling with recursive neural networks. In *Spoken Language Technology Workshop (SLT), 2014 IEEE*. IEEE, pages 554–559.

Patrick Haffner, Gokhan Tur, and Jerry H Wright. 2003. Optimizing svms for complex call classification. In *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*. IEEE, volume 1, pages I–I.

Stefan Hahn, Marco Dinarelli, Christian Raymond, Fabrice Lefevre, Patrick Lehnen, Renato De Mori, Alessandro Moschitti, Hermann Ney, and Giuseppe Riccardi. 2011. Comparing stochastic approaches to spoken language understanding in multiple languages. *IEEE Transactions on Audio, Speech, and Language Processing* 19(6):1569–1583.

Dilek Hakkani-Tür, Gökhan Tür, Asli Celikyilmaz, Yun-Nung Chen, Jianfeng Gao, Li Deng, and Ye-Yi Wang. 2016. Multi-domain joint semantic frame parsing using bi-directional rnn-lstm. In *INTERSPEECH*. pages 715–719.

Charles T Hemphill, John J Godfrey, George R Doddington, et al. 1990. The atis spoken language systems pilot corpus. In *Proceedings of the DARPA speech and natural language workshop*. pages 96–101.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9(8):1735–1780.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* .

Gakuto Kurata, Bing Xiang, Bowen Zhou, and Mo Yu. 2016. Leveraging sentence-level information with encoder lstm for semantic slot filling. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. pages 2077–2083.

Bing Liu and Ian Lane. 2015. Recurrent neural network structured output prediction for spoken language understanding. In *Proc. NIPS Workshop on Machine Learning for Spoken Language Understanding and Interactions*.

Bing Liu and Ian Lane. 2016a. Attention-based recurrent neural network models for joint intent detection and slot filling. *Interspeech 2016* pages 685–689.

Bing Liu and Ian Lane. 2016b. Joint online spoken language understanding and language modeling with recurrent neural networks. In *17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. page 22.

Grégoire Mesnil, Yann Dauphin, Kaisheng Yao, Yoshua Bengio, Li Deng, Dilek Hakkani-Tur, Xiaodong He, Larry Heck, Gokhan Tur, Dong Yu, et al. 2015. Using recurrent neural networks for slot filling in spoken language understanding. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)* 23(3):530–539.

Roderick Murray-Smith and T Johansen. 1997. *Multiple model approaches to nonlinear modelling and control*. CRC press.

Kumpati S Narendra, Yu Wang, and Wei Chen. 2014. Stability, robustness, and performance issues in second level adaptation. In *American Control Conference (ACC), 2014*. IEEE, pages 2377–2382.

Kumpati S Narendra, Yu Wang, and Wei Chen. 2015. Extension of second level adaptation using multiple models to siso systems. In *American Control Conference (ACC), 2015*. IEEE, pages 171–176.

Kumpati S Narendra, Yu Wang, and Snehasis Mukhopadhay. 2016. Fast reinforcement learning using multiple models. In *Decision and Control (CDC), 2016 IEEE 55th Conference on*. IEEE, pages 7183–7188.

Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y Ng. 2011. Multimodal deep learning. In *Proceedings of the 28th international conference on machine learning (ICML-11)*. pages 689–696.

Baolin Peng and Kaisheng Yao. 2015. Recurrent neural networks with external memory for language understanding. *arXiv preprint arXiv:1506.00195* .

Ruhi Sarikaya, Geoffrey E Hinton, and Bhuvana Ramabhadran. 2011. Deep belief nets for natural language call-routing. In *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, pages 5680–5683.

Nitish Srivastava and Ruslan R Salakhutdinov. 2012. Multimodal learning with deep boltzmann machines. In *Advances in neural information processing systems*. pages 2222–2230.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*. pages 3104–3112.

Gokhan Tur, Dilek Hakkani-Tür, and Larry Heck. 2010. What is left to be understood in atis? In *Spoken Language Technology Workshop (SLT), 2010 IEEE*. IEEE, pages 19–24.

Yu Wang. 2017. A new concept using lstm neural networks for dynamic system identification. In *American Control Conference (ACC), 2017*. IEEE, pages 5324–5329.

Yu Wang and Hongxia Jin. 2018. A boosting-based deep neural networks algorithm for reinforcement learning. In *American Control Conference (ACC), 2018*. IEEE.

Puyang Xu and Ruhi Sarikaya. 2013. Convolutional neural network based triangular crf for joint intent detection and slot filling. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*. IEEE, pages 78–83.

Puyang Xu and Ruhi Sarikaya. 2014. Contextual domain classification in spoken language understanding systems using recurrent neural network. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*. IEEE, pages 136–140.

Kaisheng Yao, Baolin Peng, Yu Zhang, Dong Yu, Geoffrey Zweig, and Yangyang Shi. 2014. Spoken language understanding using long short-term memory neural networks. In *Spoken Language Technology Workshop (SLT), 2014 IEEE*. IEEE, pages 189–194.

Xiaodong Zhang and Houfeng Wang. 2016. A joint model of intent determination and slot filling for spoken language understanding. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*. AAAI Press, pages 2993–2999.