Joint Model-Based Attention for Spoken Language Understanding Task

Xin Liu, Dalian University of Foreign Languages, China

RuiHua Qi, Dalian University of Foreign Languages, China

Lin Shao, Dalian University of Foreign Languages, China

ABSTRACT

Intent determination (ID) and slot filling (SF) are two critical steps in the spoken language understanding (SLU) task. Conventionally, most previous work has been done for each subtask respectively. To exploit the dependencies between intent label and slot sequence, as well as deal with both tasks simultaneously, this paper proposes a joint model (ABLCJ), which is trained by a united loss function. In order to utilize both past and future input features efficiently, a joint model based Bi-LSTM with contextual information is employed to learn the representation of each step, which are shared by two tasks and the model. This paper also uses sentence-level tag information learned from a CRF layer to predict the tag of each slot. Meanwhile, a submodule-based attention is employed to capture global features of a sentence for intent classification. The experimental results demonstrate that ABLCJ achieves competitive performance in the Shared Task 4 of NLPCC 2018.

KEYWORDS

Bi-RNN, CRF, Joint Models, Self-Attention, Semantic Utterance Classification, Sequence Tagging, SLU

INTRODUCTION AND BACKGROUND

According to Singh et al. (2000), “*Systems in which human users speak to a computer in order to achieve a goal are called spoken dialogue systems (SDS)*”. In recent years, task-oriented spoken dialogue system (SDS) that helps users finish tasks more efficiently via spoken interaction is being applied on various devices (Lison & Kennington 2016). There are many famous technology companies involved in this type of system, such as Apple Siri, Microsoft Cortana, Baidu Duer and so on (Hoy, 2018). As a critical component of SDS, spoken language understanding (SLU) aims to parse users’ queries and convert them to structured representations that machines can handle. The result of SLU is passed to SDS to update dialogue state and take the next proper action. Therefore, the performance of SLU is critical to SDS (Tur & De Mori, 2011).

As SLU has become a focus in research communities, the Shared Task 4 in NLPCC 2018 named “*Spoken Language Understanding in Task-Oriented Dialogue Systems*” tries to provide a platform for evaluation. It aims to parse users’ multiple rounds of queries in a session and convert them into some structure that machines can handle. To understand users’ queries expressed in spoken language, the task

DOI: 10.4018/IJDCF.2020100103

This article, originally published under IGI Global’s copyright on October 1, 2020 will proceed with publication as an Open Access article starting on January 27, 2021 in the gold Open Access journal, International Journal of Digital Crime and Forensics (converted to gold Open Access January 1, 2021), and will be distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

contains two subtasks, namely intent detection (ID) and slot filling (SF), which need to automatically recognize the intent of the queries and extract associated arguments or slots towards achieving a goal.

There are a large number of literatures on ID and SF, and many of them process the subtasks in a pipeline framework; firstly the intent is classified and secondly the semantic slots are extracted. ID is usually framed as a semantic utterance classification (SUC) problem. Many popular classifiers like support vector machines (SVMs) (Fan et al., 2008), maximum entropy (Chelba et al., 2003) and RNN models (Ravuri & Stolcke, 2015) have already been employed before. Similarly, SF can be treated as a sequence tagging problem, which is customarily solved by some traditional approaches, such as Hidden Markov Models (HMMs) (Pieraccini et al., 1992), Conditional Random Fields (CRF) (Raymond & Riccardi, 2007) and various RNN models (Mesnil et al., 2015; Vu et al., 2016; Huang et al., 2015). However, using pipeline systems not only takes more time to process tasks, but also cannot model the interaction between multiple subtasks.

In order to simplify the SLU system and use the shared information provided by ID and SF to promote the results of the two subtasks, more and more joint models for multiple tasks have also been proposed in recent years (Liu & Lane, 2016; Zhang & Wang, 2016; Wen et al., 2017). The ability to feature the correlations between subtasks helps them achieve competitive performances in ATIS.

Motivated by the inherent ability of bidirectional RNN in capturing the past and future features of sequence, this paper proposes a joint model, which uses Bi-LSTM to learn the representation of each word in Chinese query text and then share them with ID task and SF task. With a joint loss function, the two tasks can interact and promote each other through the shared representations. Experimental results demonstrate that the joint model outperforms separate models for each task.

The main contributions of this paper are:

- Adaptation of a joint model based attention mechanism, Bi-LSTM and CRF for intent determination and slot filling;
- An analysis of how intent determination and slot filling can benefit from the contextual information of the Chinese queries within a session.

BACKBONE ALGORITHM AND MODEL

This paper proposes an Attention-based Bi-LSTM-CRF Joint Model (ABLCJ) to deal with both tasks simultaneously. The backbone algorithm used in this paper and the structure of ABLCJ model is shown in Figure 1. As the Figure shows, the model is composed of three sub-modules with gray background colors. The Bi-LSTM module below is responsible for feature extraction, the module at the upper left corner can detect intents, and the module at the upper right corner is used for slot filling.

Bi-LSTM Module for Feature Extraction

Recurrent neural networks (RNN) which can maintain a memory based on historical information have been employed to produce promising results on a variety of NLP tasks. Unfortunately, as the gap widens between the relevant information and the point where it is needed grows, RNNs become unable to learn to connect the information. As a special kind of RNN, Long Short Term Memory networks (LSTMs) are explicitly designed to avoid the long-term dependency problem. The hidden layers of LSTMs are replaced by purpose-built memory cells to exploit long range dependencies in the data. To efficiently make use of the sequence's past features and future features, the researchers apply a bidirectional Long Short-Term Memory (Bi-LSTM) to learn the representation of each word in the query text. Bi-LSTM exploits both the previous and future context by processing the sequence in two directions and generates two independent sequences of LSTM output vectors. It usually learns faster than one-directional approach and is good at dealing with sequence tagging task (Reimers & Gurevych, 2017).

Figure 1. The structure of ABLCJ model

The structure of Bi-LSTM submodule for feature extraction is shown in Figure 2. The input is a sequence of words $w_1, w_2, \dots, w_n$ (for example: ["我", "想", "听", "神话"]), and n is the length of the query. After the words have been converted to word embedding, this neural network will process the whole sequence forward and backward simultaneously.

Figure 2. The structure of Bi-LSTM submodule for feature extraction

Most of the previous work processes each utterance individually, and seldom take full advantage of dialogue context. However, dialogue history is important to understand the current utterance in task-oriented dialogue systems. The intent and slot value of the current round is usually dependent on the preceding rounds. As is shown in Table 1, the 2th round and 4th round in Dialogue are both “

Table 1. Some sample dialogues

Round	Dialogue	Intent	Round	Dialogue	Slot Value
1	我想听神话		5	我想听首歌	
2	取消	music.pause	6	成都	song
3	打电话给王刚		7	启动导航,寻找目的地	
4	取消	phone_call.cancel	8	成都	destination

取消”。When people only look at this round, it is impossible to identify whether the intent is “*music.pause*” or “*phone_call.cancel*”. The same problem exists in slot filling. When people only look at the 6th round and 8th round in Dialogue, it is hard to tell whether “成都” is a song or a city. Fortunately, with the preceding rounds, it is easy to make a correct judgment. Therefore, how to model and use dialogue history for SLU is worth of researching.

Because the intent of query and slot value are usually related to the preceding part of the same session, the researchers assign the last hidden states h_n of the current query to the first hidden states h_0 of the next query, and reset h_0 to zero at the beginning of each session. The dialogue history is memorized by the hidden states h_0 , which is available for both intent detection and slot filling and contributes to the two tasks. The assignment formula is as follows:

$$h_0 = \begin{cases} 0 & (\text{first query in the session}) \\ h_n \text{ of the previous query} & (\text{other queries in the session}) \end{cases}$$

Module Based Attention for ID

The structure of module based Attention for ID is shown in Figure 3. (Note: “ B ” refers to the size of the batch, “ N ” refers to the length of the longest sentence, “ D ” refers to the dimension of hidden state, “ I ” refers to the kinds of intents.) Before passing batch of inputs of variable length to Bi-LSTM, the researchers perform zero padding to make short sentences have the same length as long sentences. In order to avoid errors in the final result, the useless information in the hidden states from Bi-LSTM has been filtered out.

Self-attention mechanism that has been used successfully in a variety of NLP tasks (Vaswani et al., 2017), is also incorporated into Bi-LSTM Model. Instead of using the mean of all hidden states by calculated component-wise, or only using the last hidden state, the intend representation h^* is generated by a sum of the word annotation vectors weighted by their corresponding different weights, where the weight a_i assigned to each value is computed by a compatibility function. Due to the limited amount of data available for training, the researchers use a linear projection instead of non-linear regression methods to transform each hidden state from D dimension to I dimension, and then apply a softmax function to obtain the weights on the values. All calculation formulas of this module are as follows:

$$s_i = (W^T h_i + b); \quad a_i = \text{softmax}(s_i); \quad h^* = \sum_{i=1}^n a_i * h_i$$

Bi-LSTM CRF Module for SF

As mentioned above, SF is usually treated as a sequence tagging problem. Considering the correlations and syntactical constrains between the current tag and neighboring tags, the researchers try not to

Figure 3. The Structure of Submodule Based Attention for ID

simply use the hidden states directly to predict the tags. Instead, they use Conditional Random Fields (CRF) models that can produce higher tagging accuracy in general (Lafferty et al., 2001). It has been shown that CRF is the most popular way to control the structure prediction and that basic idea is to use a series of potential functions to approximate the conditional probability of the output tag sequence given the input word sequence (Lin et al., 2017; Lample et al., 2016).

ABLCJ combines Bi-LSTM network and a CRF layer to form a Bi-LSTM CRF model, which is shown in Figure 4 (Note: T refers to the kinds of tags). As there is a certain semantic relationship between the intent and the slot values of a sentence, the researchers combine the intermediate results h^* generated by the previous sub-modules with each hidden state h_i in Bi-LSTM to utilize the information of one task in the other task efficiently and make a joint prediction.

Finally, instead of modeling tagging decisions independently, the pre-calculated sequence $h' = (h'_1, h'_2, \dots, h'_n)$ is taken as input to the CRF layer, and its output is final prediction tag sequence $t = (t_1, t_2, \dots, t_n)$, where t_i is in the set of all possible tags. The calculation formula of the CRF layer is as follows:

$$s(h', t) = \sum_{i=1}^n P_{i, t_i} + \sum_{i=0}^n A_{t_i, t_{i+1}}$$

As the matrix of scores that is output by the Bi-LSTM, the size of P is $n \times k$, where k is the number of distinct tags, and P_{i, t_i} refers to the probability that the i^{th} word in a sentence is marked as tag t_i . A is a matrix of transition scores, and $A_{t_i, t_{i+1}}$ represents the score of a transition from the tag t_i to tag t_{i+1} . Since the start and end tags of a sentence are added to the set of possible tags, the size of A is $(k+2) \times (k+2)$. The scoring function $s(h', t)$ makes up for the shortcomings of the traditional Bi-LSTM. When the score of a prediction sequence is high, in addition to the need for the softmax output of each location as the maximum probability value, it is also considered to add the transition probability between the tags. For example, if the subsequence ('B-Singer', 'I-Song') is included in the prediction sequence, its score will be reduced because tag 'I-Song' will never follow tag 'B-Singer'.

Figure 4. The structure of Bi-LSTM CRF submodule for SF

Experiments and Results Analysis

Data and Experimental Parameters

The dataset used for NLPCC 2018 Shared Task 4 is a sample of the real query log from a commercial task-oriented dialog system. The training data contains 21352 queries ordered by time stamp, which can be further split into 4705 sessions according to the gaps of time stamps. The test data contains 5350 queries which can be split into 1177 sessions. The data provided by the task is all in Chinese. An example session with annotations is shown in Table 2. To understand the intention of user’s queries more accurately, the model needs to understand the contexts within a session. For example, in order to judge the intent of the 4th query(“神话”), it is necessary to consider the contents of the previous query(“我想听什话”). Meanwhile, the corrected values of slot are included in annotations as well if the slot value contains ASR errors. For example, for the slot annotation “我想听<song>什话||神话</song>”, the string “神话” is the correction of “什话”.

The evaluation contains 10 intents in 3 domains (music, navigation and phone call). Within the dataset, an additional domain label ‘OTHERS’ is used to annotate the data that is not covered by the three domains. The evaluation also contains 15 types of slot. To correct ASR errors in the slot values, 12 of these types have corresponding dictionaries that include all the values occurring in the evaluation dataset.

Table 2. An example session with annotations

Session ID	User’s Query	Intent Detection	Slot Annotation
1	打电话	phone_call.make_a_phone_call	打电话
1	我想听美观	music.play	我想听<song>美观</song>
1	我想听什话	music.play	我想听<song>什话 神话</song>
1	神话	music.play	<song>神话</song>

The researchers use Jieba (Sun, 2014) as the Chinese word segmentation tool and added the dictionaries provided by task to custom dictionary. In order to increase the speed of data processing, ABL CJ model uses batch processing which enables multiple sentences to be processed at the same time. 120G+ corpus is used to train 64-dimensional word embedding as initializations and the word embedding is fine-tuned during mini-batch training with batch size of 16. Bi-LSTM cell is used as the basic RNN unit in the experiments. Given the size the data sets, the number of units in single directional LSTM cell is set to 300. Dropout rate 0.5 and L2 is applied to the non-recurrent connections (Zaremba et al., 2014) during model training for regularization. ABL CJ model uses Adam optimization method following the suggested parameter setup in (Kingma & Ba, 2014).

Evaluation Metrics

The evaluation metric of Sub-task 1 is $F1_{macro}$ value of all intents (excluding OTHERS), which is calculated as the following equations:

$$P_{macro} = \frac{1}{N} \sum_{i=1}^N \frac{\# \text{ of queries correctly predicted as intent } c_i}{\# \text{ of queries predicted as intent } c_i}$$

$$R_{macro} = \frac{1}{N} \sum_{i=1}^N \frac{\# \text{ of queries correctly predicted as intent } c_i}{\# \text{ of queries labelled as intent } c_i}$$

$$F1_{macro} = \frac{2}{\frac{1}{P_{macro}} + \frac{1}{R_{macro}}}$$

The evaluation metric of Sub-task 2 is Precision that is as given by the following equation, where “# of queries” denotes the number of queries in the test set (including the queries with intent annotated as ‘OTHERS’). “# of queries correctly parsed” denotes the number of queries for which the predicted intent and the predicted slot values (including the corrected values if correction is needed) are both exactly the same as the annotations:

$$Precision = \frac{\# \text{ of queries correctly parsed}}{\# \text{ of queries}}$$

The researchers only use the dataset and the slot dictionaries provided by the task for the model training/tuning of intent detection and slot filling.

Baselines and Final Model

The researchers improve the model step by step through experiments and analysis, and used historical model versions as baseline to compare with ABL CJ model:

- **LSTM:** This model simply feeds the hidden states from LSTM independently to a linear network layer and a softmax layer to predict the intent and slot values. In addition, this model has considered the contextual relationship of adjacent query and initialized h_0 with the last hidden state of the previous query. It uses dialogue history not only for intent detection but also for slot filling;
- **Bi-LSTM:** On the basis of the previous one, this model uses a bidirectional LSTM instead of unidirectional LSTM to extract features for user’s query sequences;

- **Attention Based Bi-LSTM (ATT-Bi-LSTM):** Instead of using the mean of all hidden states by calculated component-wise, this model incorporates self-attention mechanism into Bi-LSTM Model to predict the intent;
- **Attention Based Bi-LSTM CRF (ABLCJ):** On the basis of the previous one, this model combines Bi-LSTM network and a CRF network to complete the slot filling task.

This paper also compares the final model (ABLCJ) with the following baselines:

- **SVM:** Fan, Rong-En, et al (2018) used Support Vector Machine with linear kernel classifiers for intent detection;
- **CRF:** Raymond, C. and Riccardi, G. (2015) used a model based on conditional random fields for slot filling;
- **Attention Bi-RNN Joint Model (ABRJ):** Bing Liu and Ian Lane (2016) proposed an attention-based bidirectional RNN model for joint intent detection and slot filling.

All of the joint models above deal with ID and SF tasks together and use the same 5-fold cross validation.

Results and Analysis

The results of the two subtasks are demonstrated in Table 3. The first column lists the models introduced in previous section. The second column lists the features used by each method. “W” refers to the fact that researchers only feed the 64-dim pre-training word embedding mentioned above to the Bi-LSTM that is used to extract features. “W+D” means that the researchers add 11-dim feature vectors that are generated according to dictionaries for 12 types of slot provided by the task on the basis of the 64-dim pre-training word embedding.

Table 3. Comparison with historical model versions and previous approaches

Model	Features	Sub-Task 1(F1 _{macro})	Sub-Task 2(Precision)
LSTM	W	87.17	84.64
LSTM	W+D	90.72	86.31
Bi-LSTM	W+D	91.46	87.73
ATT-Bi-LSTM	W+D	92.98	88.52
ABLCJ	W+D	93.84	90.08
SVM	W+D	91.25	----
CRF	W+D	----	88.06
ABRJ	W+D	93.76	89.73

The results of Bi-LSTM are better than LSTM, which means that Bi-LSTM can make use of the past features and future features of the sequence efficiently. Furthermore, the model that incorporated self-attention mechanism into Bi-LSTM has made significant progress in the ID task. To model label transfer and acquire the global optimum of the whole sequence, ABLCJ Model has combined Bi-LSTM and CRF that is suitable for the sequence tagging task, and has achieved the best performance on both tasks. Meanwhile, the experimental results show that adding features generated by slot dictionaries can significantly improve the performance.

Compared with the traditional single task models used by the predecessors, ABLCJ Model outperforms SVM method for ID, making an improvement of the F1-score of 2.51%. As to SF, ABLCJ model outperforms CRF method by 2.02%. Although the absolute improvement may not be very big, ABLCJ Model outperforms the previous joint work ABRJ, thanks to the powerful ability of CRF for the sequence-level optimization.

Joint Model vs. Separate Model

In this section, the researchers compare the performance of the separate models and the joint model. The joint model is ABLCJ in Figure 1. The separate models are similar to ABLCJ except that they only process one task. For ID Only, there are only the shared layers and ID specific layers without SF specific layers. It is in the same way for SF Only.

A pipeline method has also been implemented, which is better than the separate model but not so good as ABLCJ for SF. The results are listed in Table 4.

Table 4. Comparison of joint model and separate model

Model	F1macro of ID	Precision of SF
ID Only	91.63	----
SF Only	----	87.05
Pipeline	91.63	88.34
Joint Model	93.84	90.08

For quantitative analysis, the programs of the joint model and separate models are executed respectively with the same parameter settings. The results of this experiment indicate that the joint model outperforms the separate models for both tasks, which suffices to show that the joint training is effective.

Contribution of Dialogue History

In order to analyze how intent determination and slot filling can benefit from the contextual information in a session, this section adds a set of experiments, and the relevant models are as follows:

- **LSTM Without Contextual Information (LSTM-WCI):** This model is the same as LSTM except that the interaction between adjacent queries is not considered and the hidden state h_0 of each sequence is initialized with zero in each round;
- **LSTM-Intent:** This model is the same as LSTM except that it uses dialogue history for intent prediction only;
- **Attention Based Bi-LSTM CRF Without Contextual Information (ABLCJ-WCI):** This model is the same as ABLCJ except that it does not make use of the information in dialogue context;
- **ABLCJ-Intent:** This model is the same as ABLCJ except that it uses dialogue history for intent prediction only.

The results are demonstrated in Table 5, which shows that the LSTM-WCI preforms worst on both tasks. LSTM-Intent and ABLCJ-Intent use dialogue history for intent detection, which improves the score of intent significantly. LSTM out performs LSTM-WCI and LSTM-Intent on both tasks and ABLCJ uses dialogue history not only for intent detection but also for slot filling and obtains the best performance on both tasks. The experimental results indicate that it is effective to consider

Table 5. Contribution of the dialogue contextual information

Model	Features	Sub-Task 1(F1 _{macro})	Sub-Task 2(Precision)
LSTM-WCI	W+D	86.17	83.64
LSTM-Intent	W+D	90.21	84.52
LSTM	W+D	90.72	86.31
ABLCJ-WCI	W+D	91.63	87.97
ABLCJ-Intent	W+D	93.39	88.26
ABLCJ	W+D	93.84	90.08

context information in the same session when analyzing multiple rounds of query, which supports our argument that the dialogue history is important for the two tasks in SLU.

CONCLUSION

Joint Model can deal with both tasks in SLU simultaneously and more effectively. Meanwhile, dialogue history provides import information for SLU in dialogues. In this paper, the researchers have introduced attention-based recurrent neural networks with contextual information for joint intent determination and slot filling, which are two major tasks in SLU. The submodule based on Bi-LSTM is used to learn the sequence representations shared with two tasks. The shared representations are fed to a submodule that incorporates self-attention mechanism to predict intent values. Meanwhile, the tags of slots are also predicted by the shared representations. The procedure uses the submodule that combines Bi-LSTM and CRF. Through a united loss function and shared representations, the correlations of the two tasks are learned so as to promote each other. The researchers conduct experiments on the datasets provided by the Shared Task 4 in NLPCC 2018. The joint model demonstrates advantages over separate models and achieves competitive performance on both tasks.

In future works, the researchers plan to improve ABLCJ model by introducing CNN into the sub-module for feature extraction. ABLCJ model only uses one layer of Bi-LSTM in the proposed models, and deeper models by stacking the Bi-LSTM layers are to be explored in future work. The researchers will also test ABLCJ on English datasets to investigate the generalization of this model.

ACKNOWLEDGMENT

This paper would like to gratefully acknowledge the organizers of NLPCC 2018 as well as Tencent Dingdang for making the datasets available. The authors were supported by National Social Science Fund General Project “Mining research on foreign reader’s online comments on translation of Chinese classics into English” [grant number 15BYY028]; National Natural Science Foundation Youth Science Fund Project “Research on Multilingual Text Affective Analysis Method for Social Media” [grant number 61806038]; Ministry of Education Humanities and Social Sciences Research Project “Research on multi lingual text emotion recognition oriented to ‘one belt and one road”” [grant number 18YJCZH208]; Liaoning Natural Science Foundation “Research on Search Engine-Oriented Multi-Source Knowledge Map Fusion” [grant number 20170540232]; Dalian University of Foreign Languages Research Innovation Team “Computational Linguistics and Artificial Intelligence Innovation Team” [grant number 2016CXTD06]; and Dalian University of Foreign Languages Research Project “Research on Information Extraction in Weak Supervisory Domain Based on Deep Learning” [grant number 2016XJJS56].

REFERENCES

- Chelba, C., Mahajan, M., & Acero, A. (2003). *Speech utterance classification*. Paper presented at the ICASSP Conference, Hong Kong, China.
- Fan, R. E., Chang, K. W., Hsieh, C. J., Wang, X. R., & Lin, C. J. (2008). LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, 9(Aug), 1871–1874.
- Hoy, M. B. (2018). Alexa, siri, cortana, and more: An introduction to voice assistants. *Medical Reference Services Quarterly*, 37(1), 81–88. doi:10.1080/02763869.2018.1404391 PMID:29327988
- Huang, Z., Xu, W., & Yu, K. (2015). *Bidirectional LSTM-CRF models for sequence tagging*. arXiv preprint arXiv:1508.01991
- Kingma, D. P., & Ba, J. (2014). *Adam: A method for stochastic optimization*. Paper presented at the ICLR Conference, Banff, Canada.
- Lafferty, J., McCallum, A., & Pereira, F. C. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning* (pp. 282–289). San Francisco, CA: Morgan Kaufmann Publishers Inc.
- Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., & Dyer, C. (2016). Neural architectures for named entity recognition. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 260–270). San Diego, CA: Association for Computational Linguistics.
- Lin, B. Y., Xu, F., Luo, Z., & Zhu, K. (2017). Multi-channel bilstm-crf model for emerging named entity recognition in social media. In *Proceedings of the 3rd Workshop on Noisy, User-generated Text (W-NUT) at EMNLP* (pp. 160-165). Copenhagen, Denmark: Academic Press.
- Lison, P., & Kennington, C. (2016). OpenDial: A toolkit for developing spoken dialogue systems with probabilistic rules. *Proceedings of ACL-2016 System Demonstrations*, 67-72. doi:10.18653/v1/P16-4012
- Liu, B., & Lane, I. (2016). *Attention-based recurrent neural network models for joint intent detection and slot filling*. Paper presented at Annual Conference of the International Speech Communication Association, San Francisco, CA. doi:10.21437/Interspeech.2016-1352
- Mesnil, G., Dauphin, Y., Yao, K., Bengio, Y., Deng, L., Hakkani-Tur, D., & Zweig, G. et al. (2015). Using recurrent neural networks for slot filling in spoken language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(3), 530–539.
- Pieraccini, R., Tzoukermann, E., Gorelov, Z., Gauvain, J. L., Levin, E., Lee, C. H., & Wilpon, J. G. (1992). *A speech understanding system based on statistical representation of semantics*. Paper presented at the ICASSP Conference, San Francisco, CA. doi:10.1109/ICASSP.1992.225939
- Ravuri, S., & Stolcke, A. (2015). *Recurrent neural network and lstm models for lexical utterance classification*. Paper presented at Annual Conference of the International Speech Communication Association, Dresden, Germany.
- Raymond, C., & Riccardi, G. (2007). *Generative and discriminative algorithms for spoken language understanding*. Paper presented at Annual Conference of the International Speech Communication Association, Antwerp, Belgium.
- Reimers, N., & Gurevych, I. (2017). *Optimal hyperparameters for deep lstm-networks for sequence labeling tasks*. Paper presented at the EMNLP Conference, Copenhagen, Denmark.
- Singh, S. P., Kearns, M. J., Litman, D. J., & Walker, M. A. (2000). Reinforcement learning for spoken dialogue systems. *Advances in Neural Information Processing Systems*, 956–962.
- Sun, J. Y. (2014). *Jieba Chinese text segmentation*. Retrieved from <https://github.com/fxsjy/jieba>.
- Tur, G., & De Mori, R. (2011). *Spoken language understanding: Systems for extracting semantic information from speech*. Hoboken, NJ: John Wiley & Sons. doi:10.1002/9781119992691

- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., & Polosukhin, I. et al. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 5998–6008.
- Vu, N. T., Gupta, P., Adel, H., & Schütze, H. (2016). *Bi-directional recurrent neural network with ranking loss for spoken language understanding*. Paper presented at the ICASSP Conference, Shanghai, China. doi:10.1109/ICASSP.2016.7472841
- Wen, L., Wang, X., Dong, Z., & Chen, H. (2017). *Jointly Modeling Intent detection and Slot Filling with Contextual and Hierarchical Information*. Paper presented at National CCF Conference on Natural Language Processing and Chinese Computing, Dalian, China.
- Zaremba, W., Sutskever, I., & Vinyals, O. (2014). *Recurrent neural network regularization*. arXiv preprint arXiv:1409.2329
- Zhang, X., & Wang, H. (2016). *A Joint Model of Intent Determination and Slot Filling for Spoken Language Understanding*. Paper presented at the IJCAI Conference, Macao, China.

Liu Xin's education Experience: July 2004, graduated with a Bachelor's Degree in Computer Science from Harbin Institute of Technology July 2007, graduated with a Post-Graduate Degree in Computer Science from Harbin Institute of Technology. Work Experience: From September 2007 to the present, Teacher of Dalian University of Foreign Languages.

RuiHua Qi (PhD) was born in 1974. She is a professor of Dalian University of Foreign Languages. Her major research interests are in Natural Language Process and Text Mining.

Lin Shao is an English teacher at Dalian University of Foreign Languages, China. They earned their BA degree 2003-2005 Duquesne University, PA, USA. They received their MA degree 2006.