A BERT BASED JOINT LEARNING MODEL WITH FEATURE GATED MECHANISM FOR SPOKEN LANGUAGE UNDERSTANDING

Wang Zhang[1,2] *Lei Jiang*[1*] *Shaokang Zhang*[1,2] *Shuo Wang*[1,2] *Jianlong Tan*[1]

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
[2] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China

ABSTRACT

Intent detection (ID) and slot filling (SF) are two major tasks for spoken language understanding (SLU). Recent joint learning approaches consider the relationship between intent detection and slot filling, which leverage the shared knowledge across two tasks to benefit each other. However, most existing methods do not make full use of the BERT model and gate mechanisms to improve the semantic correlation between slot filling and intent detection tasks. In this paper, we propose a joint learning model based on BERT, which introduce dual encoder structure and utilizes semantic information by performing feature gate mechanisms in predicting intents and slots. Experimental results demonstrate that our proposed method provides very competitive results on CAIS and DDoST datasets.

***Index Terms*—** Spoken Language Understanding, Slot Filling, Intent Detection, Pre-trained Language Model

1. INTRODUCTION

Spoken language understanding, which aims to form a semantic-frame structure from user utterances or queries, is a core component in intelligent human-machine dialogue systems. It typically contains two tasks: intent detection and slot filling tasks [1].

Traditional approaches treat ID and SF as two independent tasks, which ignore semantic correlations across the two tasks. Intuitively, ID and SF tasks are semantically related. Therefore, recent approaches try to exploit the relationship between ID and SF tasks simultaneously in a joint learning method, such as Attention BiRNN [2], Slot-Gated [3], Stack-Propagation [4] and Joint BERT [5]. Although these methods have achieved good performance, there are still some issues. First, most approaches do not consider using a pre-trained language model to extract contextual representations. Pre-trained language models, such as OpenAI GPT [6] and BERT [7], can provide rich semantic knowledge and achieve excellent results in various NLP tasks [8]. And they can also improve the problem of poor generalization ability due to the lack of labeled data in the SLU tasks [9]. Second, these methods do not make full use of different kinds of semantic information in predicting intent and slots.

In this paper, we propose a BERT based joint learning model for intent detection and slot filling tasks. First, through the dual encoder, BiRNN encoder and BERT encoder, we obtain sequential and global context representations of the utterance. Second, intent decoder uses sequential and global context representations to predict the intent label. Then, slot decoder uses the above representations and the predicted intent label to predict the slot label. We also use feature gates to balance the importance of the each kind of information in the above decoders. The experimental results indicate the superiority of our proposed method. Our main contributions are listed as follows:

- We propose a new joint model that uses dual encoders to simultaneously learn sequential information and global context information.
- We designed a feature gate mechanism to automatically assign the importance of different kinds of information.
- The proposed model is evaluated on two Chinese SLU datasets and experimental results show that our method provides very competitive results on the intent detection and slot filling tasks.

2. PROPOSED METHOD

In this section, we first describe our approach on BERT encoder and bidirectional RNN (BiRNN) encoder. Following that, we introduce our proposed method on intent decoder and slot decoder for intent detection and slot filling. The model architecture is illustrated in Fig. 1.

2.1. Encoder

The model architecture of BERT contains multi-layer bidirectional Transformer encoders [10]. In our method, we use BERT_{base} model to compute the contextual semantic representations for every token in the sequence. Given an input

*Corresponding Author: Lei Jiang, jianglei@iie.ac.cn

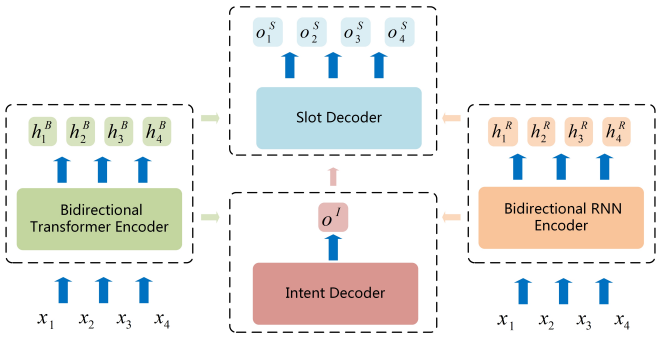

Fig. 1. The overview architecture of our model.

token sequence $\mathrm{X} = (x_1, ..., x_T)$, the BERT encoder outputs are denoted by following equation:

$$\mathrm{H}^B = \mathrm{BERT}(x_1, ..., x_T) \tag{1}$$

where $\mathrm{H}^B = (h_1^B, ..., h_T^B)$, h_i^B is the semantic representation of token x_i.

At the same time, we also use bidirectional RNN encoder to get the sequential semantic representation of the utterance, which has been successfully applied in spoken language understanding [11, 12] and can consider both past and future information at the same time. We use GRU [13] as the basic unit in our work, because it can better capture long-term dependencies than simple RNN and has less parameters comparing to LSTM. The bidirectional RNN encoder reads the token sequence forward and backward. Forward RNN encoder reads the sequence of tokens in its original order and generates a hidden state $\overrightarrow{h_t}$ at each time step t. And backward RNN generates a hidden state $\overleftarrow{h_t}$ in reverse order. We define bidirectional hidden state as follows:

$$\overrightarrow{h_t} = \overrightarrow{\mathrm{GRU}}(x_t, \overrightarrow{h}_{t-1}) \tag{2}$$

$$\overleftarrow{h_t} = \overleftarrow{\mathrm{GRU}}(x_t, \overleftarrow{h}_{t+1}) \tag{3}$$

$$h_t^R = [\overrightarrow{h_t}, \overleftarrow{h_t}] \tag{4}$$

where h_t^R is the bidirectional hidden state at time step t, $[,]$ is the concatenation operation. Given an input token sequence $\mathrm{X} = (x_1, ..., x_T)$, the output of BiRNN encoder is $\mathrm{H}^R = (h_1^R, ..., h_T^R)$.

2.2. Intent Decoder

In intent decoder, we use the information encoded by the BERT model and the information encoded by the BiRNN model to obtain the representation of the whole sentence. First, we apply an attention module to calculate the weighted sum of all hidden states from H^R:

$$c^{RI} = \sum_{i=1}^{T} \alpha_i h_i^R, \alpha_i = \frac{\exp(e_i)}{\sum_{k=1}^{T} \exp(e_k)}, e_k = f(h_k^R, h_T^R) \tag{5}$$

where f is a single layer neural network, and c^{RI} is the context representation of BiRNN encoder. The last state of BiRNN encoder carries information of the entire source sequence. In this way, we can capture the complex pairwise relations between the last state and all other states from its assigned scores. On the other hand, Reimers *et al*. [14] have previously demonstrated that the average context embedding is better than the [CLS] token embedding. Thus, we average the output representations of the BERT encoder as the representation of the whole sentence. The context representation of BERT encoder is defined as:

$$c^{BI} = \frac{1}{T} \sum_{i=1}^{T} h_i^B \tag{6}$$

In order to adaptively balance the different information, we design a feature gate to automatically assign the importance to each kind of the information. To be specific, the global representation is defined as:

$$c_{global} = [\tilde{c}_1, ..., \tilde{c}_K], \tilde{c}_i = g_i \odot c_i \tag{7}$$

$$g_i = \sigma\big(f_i^1(c_i) + f_i^2([c_1, ..., c_{i-1}, c_{i+1}, ..., c_K])\big) \tag{8}$$

where f_i^1 and f_i^2 are single layer neural networks, σ is the sigmoid function ($\sigma(x) = \frac{1}{1+e^{-x}}$), and $\odot$ is the Hadamard product. g_i can be regarded as the weight feature of the representation vector c_i and others. Thus, we use g_i as a "valve" that control the information flow of representations to get the final representation and predict the label:

$$c^I = [\tilde{c}^{BI}, \tilde{c}^{RI}, \tilde{h}_T^R] \tag{9}$$

$$P(\hat{y}^I | c^I) = \mathrm{softmax}(\mathbf{W}^I c^I + \mathbf{b}^I) \tag{10}$$

$$\hat{o}^I = \underset{\hat{y}^I \in S^I}{\arg\max}\, P(\hat{y}^I | c^I) \tag{11}$$

where $\hat{o}^I$ is the predicted intent label of the entire utterance, S^I is the intent label set, and $\mathbf{W}^I$ and $\mathbf{b}^I$ are the trainable parameters.

2.3. Slot Decoder

Slot filling can be treated as a sequence labeling problem. Thus, in slot filling, we map a word sequence $\mathrm{X} = (x_1, ..., x_T)$ to its corresponding slot label sequence $\mathrm{O}^S = (o_1^S, ..., o_T^S)$. At each time step t, we first use a unidirectional RNN to decode the encoded information of BiRNN:

$$h_i = \mathrm{GRU}\big(h_{i-1}, E(\hat{o}_{i-1}^S)\big) \tag{12}$$

where h_{i-1} is the previous decoder state, $\hat{o}_{i-1}^S$ is the previous emitted slot label, and $E(x)$ is a simple embedding function. Then, we also calculate the context vector c_i^{RS} from H^R:

$$c_i^{RS} = \sum_{j=1}^{T} \alpha_{i,j}^R h_j^R, \alpha_{i,j}^R = \frac{\exp(e_{i,j}^R)}{\sum_{k=1}^{T} \exp(e_{i,k}^R)}, e_{i,k}^R = f_1(h_i, h_k^R) \tag{13}$$

where f_1 is a single layer neural network. Here, we use the attention mechanism to obtain the relationship between the current hidden state and the encoded states, and get the additional global support information. This information may not be fully captured by the current hidden state. Similarly, we use the same method to obtain the context vector c_i^{BS} from H^B:

$$c_i^{BS} = \sum_{j=1}^{T} \alpha_{i,j}^B h_j^B, \alpha_{i,j}^B = \frac{\exp(e_{i,j}^B)}{\sum_{k=1}^{T} \exp(e_{i,k}^B)}, e_{i,k}^B = f_2(h_i^B, h_k^B) \tag{14}$$

where f_2 is a single layer neural network.

We use the representations of BiRNN encoder, the representations of BERT encoder and the intent label predicted by the intent decoder to predict the current slot label. We also apply a feature gate to automatically assign the importance to each kind of information in the slot decoder.

$$\tilde{e}^I = \text{Feature-Gate}(E(\hat{o}^I)) \tag{15}$$

$$c_i^S = [\tilde{c}_i^{RS}, \tilde{c}_i^{BS}, \tilde{h}_i^R, \tilde{e}^I, \tilde{h}_i] \tag{16}$$

$$P(\hat{y}_i^S | c_i^S) = \text{softmax}(\mathbf{W}^S c_i^S + \mathbf{b}^S) \tag{17}$$

$$\hat{o}_i^S = \arg\max_{\hat{y}_i^S \in S^S} P(\hat{y}_i^S | c_i^S) \tag{18}$$

where $\mathbf{W}^S$ and $\mathbf{b}^S$ are the trainable parameters, $\tilde{h}_i^R$ is the aligned BiRNN encoder hidden state h_i^R with the feature gate, $\hat{o}_i^S$ is the predicted slot label of the current token, and S^S is the slot label set.

2.4. Optimization

To jointly model intent classification and slot filling tasks, we add up the cross entropy losses of the above tasks. The losses of two tasks are defined as follows:

$$\mathcal{L}_I = -\sum_{i=1}^{n_I} y_i^I \log(P(\hat{y}^I | c^I)) \tag{19}$$

$$\mathcal{L}_S = -\sum_{t=1}^{T} \sum_{j=1}^{n_S} y_{t,j}^S \log(P(\hat{y}_t^S | c^S)) \tag{20}$$

where y_i^I and $y_{t,j}^S$ are golden labels, n^I is the size of intent label set, and n_S is the size of slot label set. On the other hand, we also construct some negative samples to improve the representation quality of BiRNN encoder. This is similar to using the data augmentation technology in nlp, which can boost performance on nlp tasks [15]. Here, we use the random shuffling method to generate negative samples, and get their representations H_{neg}^R through the BiRNN encoder. The negative sample loss is defined as follows:

$$\mathcal{L}_{neg} = -\frac{1}{N_{neg}} \log\sigma(z^\top z_{neg}) \tag{21}$$

where z is the sentence representation, and z_{neg} is the sentence representation of the negative sample. We adopt a joint training scheme for optimization and the final joint objective function is defined as follows:

$$\mathcal{L} = \mathcal{L}_I + \mathcal{L}_S + \eta \mathcal{L}_{neg} + \nu \mathcal{L}_{reg} \tag{22}$$

where η and ν are hyperparameters, and $\mathcal{L}_{reg}$ is an L2-norm regularizer.

3. EVALUATION

3.1. Experimental Setup

To evaluate the efficiency of our method, we conduct experiments on two Chinese datasets, DDoST and CAIS. DDoST is a DDoS service chat data set, which includes 1379 training, 232 validation and 699 test utterances. There are 71 slot labels and 16 intent types. Chinese Artificial Intelligence Speakers (CAIS) corpus is widely used for the Chinese SLU research. The training, development and test sets in CAIS consists of 7995, 994 and 1024 utterances, respectively. There are 75 slot labels and 11 intent types.

We use three evaluation metrics to evaluate our model. For the intent detection task, the accuracy is adopted. The F1-score is used in the slot filling task. Furthermore, we use sentence-level semantic frame accuracy (sentence accuracy) to indicate the overall accuracy of the two tasks.

We use Chinese BERT-wwm-ext base model[1] in the BERT endcoder. Adam [16] is used for optimization. The learning rate of the BERT encoder is 3e-5. And the learning rate of other components in the model is 0.001. The batch size is 32. Hyperparameter η is 1.0 for DDoST and 0.1 for CAIS. Hyperparameter ν is set to 1e-5.

3.2. Experimental Results

Table 1 shows the main results of our model and compared baselines on the CAIS and DDoST datasets. (The results of Slot-Gated, SF-ID and Stack-Propagation on CAIS dataset are cited from [17], and the results of CM-Net on the CAIS dataset are cited from [18].) On the CAIS dataset, our method achieves 0.77% improvement on the slot filling task, 0.19% improvement on the intent detection task and 1.58% improvement on the sentence-level semantic frame accuracy when compared with all baselines. On the DDoST dataset, our method achieves 0.86% on the intent detection task and 2.43% improvement on the sentence-level semantic frame accuracy. The performance of our method is slightly lower than ITD-Net on the slot filling task. However, compared with ITD-Net, our method achieves significant improvements (6.16% and 2.43%) on the intent accuracy and sentence accuracy.

[1] https://github.com/ymcui/Chinese-BERT-wwm

Table 1. Intent detection and slot filling results on CAIS and DDoST datasets.

Models	CAIS			DDoST		
	Slot (F1)	Intent (Acc)	Sen (Acc)	Slot (F1)	Intent (Acc)	Sen (Acc)
Slot-Gated	81.13	94.37	80.83	-	-	-
SF-ID	84.85	94.27	82.41	-	-	-
Stack-Propagation	87.64	94.37	84.68	71.30	85.98	63.09
CM-Net	86.16	94.56	-	-	-	-
ITD-Net	-	-	-	**71.64**	82.40	63.38
Joint BERT	76.98	95.16	76.19	63.45	87.70	57.51
Our method	**88.41**	**95.35**	**86.26**	71.39	**88.56**	**65.81**

Table 2. Ablation experiments on CAIS and DDoST datasets.

Models	CAIS			DDoST		
	Slot (F1)	Intent (Acc)	Sen (Acc)	Slot (F1)	Intent (Acc)	Sen (Acc)
Our method	**88.41**	**95.35**	**86.26**	**71.39**	**88.56**	**65.81**
w/o feature gate (slot)	88.02	95.26	85.87	70.24	87.84	64.52
w/o feature gate (intent)	87.54	95.15	84.09	70.20	87.70	63.95
w/o intent information	88.28	94.96	85.87	69.31	86.41	61.66
w/o negative samples	87.00	94.86	84.68	70.55	87.27	64.23

In addition, compared with Joint BERT, our method has great advantages on the slot filling task, because token-level representations in Joint BERT are not appropriate for the slot filling tasks of the Chinese SLU. This shows that BiRNN encoders can effectively promote learning token-level semantics information. In general, these results indicate that dual encoders with feature gate mechanism can effectively learn token-level and utterance-level semantics information, and improve the performance of intent detection and slot filling tasks.

3.3. Ablation Study

In this section, we conduct ablation experiments to explore the effectiveness of each part of our proposed model. The results of the ablation experiments are shown in Table 2. First, from the results of removing feature gate mechanism in the slot decoder or intent decoder, we observe that the results of three metrics on both datasets decline. This demonstrates that feature gate mechanism can balance the different kinds of information and promote intent detection and slot filling tasks. We also observe that the results of three metrics on both datasets decline when removing predicted intent label in the slot decoder. The drops show that the interaction of intent information is beneficial to the joint model. Besides, we can see that using negative samples for BiRNN encoder produces better results. We believe that the reason is that negative samples allow the BiRNN encoder to produce more robust results and improve the quality of the representation.

4. RELATED WORK

According to the way of using semantic knowledge, the current joint models are mainly divided into two categories: implicit joint model and explicit joint model [19] [20]. And we briefly introduce the representative studies here.

For implicit joint methods, these models only use a shared encoder to capture shared features for both tasks without any explicit interaction. Liu and Lane [2] propose an attention-based RNN model, in which the authors explore how to effectively introduce alignment information. Chen *et al*. [5] propose a joint ID and SF model based on BERT and explore BERT model for SLU.

For explicit joint methods, the models explicitly leverage the interaction information between ID and SF tasks to improve performance. Li *et al*. [21] also propose a self-attentive model, which introduces intent-augmented gate mechanism to model the interaction between ID and SF. Qin *et al*. [4] propose a stack-propagation framework for SLU to incorporate the token-level intent information to guide the slot filling and alleviate the error propagation. Peng *et al*. [22] propose an interactive two-pass decoding network, which leverage token-level interactions between the ID and SF.

5. CONCLUSION

In this paper, we present a new joint learning method for intent detection and slot filling. Our model first uses the dual encoder structure to fully learn sequential information and global context information. Then, our method introduces feature gate mechanism to calculate the weight of different kinds of information. Our method also introduces noise samples to enhance the learning representations. We evaluate our method in intent detection and slot filling tasks. The experimental results demonstrate that our proposed method provides very competitive results compared with baselines on two Chinese SLU datasets.

6. ACKNOWLEDGEMENT

This paper is supported by Pilot Projects of Chinese Academy of Sciences (No.Y9W0013401).

7. REFERENCES

[1] Gokhan Tur and Renato De Mori, *Spoken language understanding: Systems for extracting semantic information from speech*, John Wiley & Sons, 2011.

[2] Bing Liu and Ian Lane, "Attention-based recurrent neural network models for joint intent detection and slot filling," in *Interspeech 2016*, 2016, pp. 685–689.

[3] Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen, "Slot-gated modeling for joint slot filling and intent prediction," in *NAACL 2018*, 2018, pp. 753–757.

[4] Libo Qin, Wanxiang Che, Yangming Li, Haoyang Wen, and Ting Liu, "A stack-propagation framework with token-level intent detection for spoken language understanding," in *EMNLP 2019*, 2019, pp. 2078–2087.

[5] Qian Chen, Zhu Zhuo, and Wen Wang, "Bert for joint intent classification and slot filling," *arXiv preprint arXiv:1902.10909*, 2019.

[6] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al., "Language models are few-shot learners," *arXiv preprint arXiv:2005.14165*, 2020.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *NAACL 2019*, 2019, pp. 4171–4186.

[8] Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang, "Pre-trained models for natural language processing: A survey," *Science China Technological Sciences*, pp. 1–26, 2020.

[9] Zhichang Zhang, Zhenwen Zhang, Haoyuan Chen, and Zhiman Zhang, "A joint learning framework with bert for spoken language understanding," *IEEE Access*, vol. 7, pp. 168849–168858, 2019.

[10] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.

[11] Grégoire Mesnil, Yann Dauphin, Kaisheng Yao, Yoshua Bengio, Li Deng, Dilek Hakkani-Tur, Xiaodong He, Larry Heck, Gokhan Tur, Dong Yu, et al., "Using recurrent neural networks for slot filling in spoken language understanding," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 3, pp. 530–539, 2014.

[12] Xiaodong Zhang and Houfeng Wang, "A joint model of intent determination and slot filling for spoken language understanding," in *IJCAI 2016*, 2016, pp. 2993–2999.

[13] Kyunghyun Cho, B van Merrienboer, Caglar Gulcehre, F Bougares, H Schwenk, and Yoshua Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," in *EMNLP 2014*, 2014, pp. 1724–1734.

[14] Nils Reimers, Iryna Gurevych, Nils Reimers, Iryna Gurevych, Nandan Thakur, Nils Reimers, Johannes Daxenberger, and Iryna Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," in *EMNLP 2019*, 2019, pp. 3980–3990.

[15] Jason Wei and Kai Zou, "Eda: Easy data augmentation techniques for boosting performance on text classification tasks," in *EMNLP-IJCNLP 2019*, 2019, pp. 6383–6389.

[16] Diederik P Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[17] Dechuang Teng, Libo Qin, Wanxiang Che, Sendong Zhao, and Ting Liu, "Injecting word information with multi-level word adapter for chinese spoken language understanding," *arXiv preprint arXiv:2010.03903*, 2020.

[18] Yijin Liu, Fandong Meng, Jinchao Zhang, Jie Zhou, Yufeng Chen, and Jinan Xu, "Cm-net: A novel collaborative memory network for spoken language understanding," in *EMNLP-IJCNLP 2019*, 2019, pp. 1050–1059.

[19] Libo Qin, Tianbao Xie, Wanxiang Che, and Ting Liu, "A survey on spoken language understanding: Recent advances and new frontiers," *arXiv preprint arXiv:2103.03095*, 2021.

[20] Samuel Louvan and Bernardo Magnini, "Recent neural methods on slot filling and intent classification for task-oriented dialogue systems: A survey," in *Proceedings of the 28th International Conference on Computational Linguistics*, 2020, pp. 480–496.

[21] Changliang Li, Liang Li, and Ji Qi, "A self-attentive model with gate mechanism for spoken language understanding," in *EMNLP 2018*, 2018, pp. 3824–3833.

[22] Huailiang Peng, Mengjun Shen, Lei Jiang, Qiong Dai, and Jianlong Tan, "An interactive two-pass decoding network for joint intent detection and slot filling," in *NLPCC 2020*, 2020, pp. 69–81.