WAIS: Word Attention for Joint Intent Detection and Slot Filling

Sixuan Chen,[1]* **Shuai Yu** [2]*
[1]Department of Computer Science and Engineering,
Hong Kong University of Science and Technology
[2]School of Computer Science, Fudan University
E-mail: schenbd@connect.ust.hk, Tel: +852 95175593

Abstract

Attention-based recurrent neural network models for joint intent detection and slot filling have achieved a state-of-the-art performance. Most previous works exploited semantic level information to calculate the attention weights. However, few works have taken the importance of word level information into consideration. In this paper, we propose *WAIS*, word attention for joint intent detection and slot filling. Considering that intent detection and slot filling have a strong relationship, we further propose a fusion gate that integrates the word level information and semantic level information together for jointly training the two tasks. Extensive experiments show that the proposed model has robust superiority over its competitors and sets the state-of-the-art.

Introduction

Spoken language understanding (SLU) is a critical component in spoken dialogue systems. SLU aims to form a semantic frame that captures the semantics of user utterances or queries. It typically involves two tasks: intent detection and slot filling (Tur and De Mori 2011). These two tasks focus on predicting speaker's intent and extracting semantic concepts as constraints for natural language.

Considering that pipelined approaches usually suffer from error propagation due to their independent models, the joint model has been proposed to improve sentence-level semantics via mutual enhancement between two tasks (Guo et al. 2014; Hakkani-Tür et al. 2016). In addition, the attention mechanism (Bahdanau, Cho, and Bengio 2014) was introduced and leveraged into the model in order to provide a precise focus, which allows the network to learn where to pay attention to in the input sequence for each output label (Liu and Lane 2015; 2016). The attentional model proposed by Liu and Lane (Liu and Lane 2016) achieved the state-of-the-art performance. However, the prior works ignored the importance of word level information, instead, only semantic information was exploited to calculate the attention weights for the two tasks. Inspired by the success (Wu et al. 2018; Wang et al. 2018) on NLP tasks, we introduce word-level attention for this task.

[2]Shuai Yu is the corresponding author. * denotes that the two authors contributed equally.

The contribution of this work is threefold: (1) We propose a word attention technique to utilize word level information for joint training the two tasks. To the best of our knowledge, we are the first to introduce the word level information for the two tasks. (2) Further, we propose a fusion gate that integrates the context vectors from word level information and semantic level information. (3) The conducted experiments demonstrate the effectiveness of our proposed approach that sets the state-of-the-art.

Methodology

The bidirectional long short-term memory (BLSTM) model takes a word sequence $\mathbf{x} = (x_1, ..., x_T)$ as input, and then generates the forward hidden state $\overrightarrow{h_i}$ and the backward hidden state $\overleftarrow{h_i}$. The final hidden state h_i at time step i is a concatenation of $\overrightarrow{h_i}$ and $\overleftarrow{h_i}$, i.e. $h_i = [\overrightarrow{h_i}, \overleftarrow{h_i}]$.

Slot Filling. For slot filling, $\mathbf{x}$ is mapped to its corresponding slot label $\mathbf{y} = (y_1^s, ..., y_T^s)$. For each hidden state h_i, we compute the slot context vector c_i^s as the weighted sum of the hidden states by the learned attention weights:

$$c_i^s = \sum_{j=1}^{T} \alpha_{i,j}^s h_j \tag{1}$$

where the slot attention can be calculated as below:

$$\alpha_{i,j}^s = \frac{exp(e_{i,j})}{\sum_{k=1}^{T} exp(e_{i,k})} \tag{2}$$

$$e_{i,k} = \sigma(W_h^s h_k) \tag{3}$$

where σ is activation function and W_h^s is the weight matrix of a feed-forward neural network. Then the hidden state and the slot context vector are utilized for slot filling:

$$y_i^s = Softmax(W_{hy}^s(h_i + c_i^s)) \tag{4}$$

where W_{hy}^s is the weight matrix and y_i^s is the slot label of the i-th word in the input.

Intent Detection. For intent detection, the intent context vector c^I can also be computed in the same manner as c_i^s, but the intent detection part only takes the last hidden of BLSTM h_T. The intent prediction can be modeled as below:

$$y^I = Softmax(W_{hy}^I(h_T + c^I)) \tag{5}$$

Model	ATIS		Snips	
	Slot (F1)	Intent(Acc)	Slot (F1)	Intent (Acc)
RecNN+Viterbi (Guo et al. 2014)	93.96	95.40	88.3	97.26
Attention-Based (Liu and Lane 2016)	95.78	98.1	90.96	98.0
WAIS	**95.87**	**98.6**	**91.09**	**98.43**

Table 1: SLU performance on ATIS and Snips datasets (%).

Word Attention. To add word level information, at each decoding step, we leverage source word embedding x_j together to compute *word attention weight* β_{ij}, and by weighted sum the word embeddings using β_{ij}, we can obtain another context vector c_i^{ω}, which we refer to as the word context vector.

Specifically, we first calculate the attention weight β_{ij} as the softmax of energy function e_{ij}^{β}:

$$\beta_{ij} = \frac{exp(e_{ij}^{\beta})}{\sum_{k=1}^{T} exp(e_{ik}^{\beta})} \tag{6}$$

where the energy function e_{ij}^{β} is computed by the j-th source word embedding x_j:

$$e_{ij}^{\beta} = V^T \tanh(W_{hy}^{w} x_j) \tag{7}$$

where $V \in \mathbb{R}^m$, $W_{hy}^{w} \in \mathbb{R}^{m \times m}$ are the weight matrices, with the dimension of word embedding denoted as m.

After getting the attention weight β_{ij} for all source words, the word context vector c_i^{ω} can be calculated as:

$$c_i^{\omega} = \sum_{j=1}^{T} \beta_{ij} x_j \tag{8}$$

The word context vector is then provided as an additional input to derive intent detection and slot filling.

Fusion Gate. To achieve a better balance at both the word level and semantic level information via adaptively controlling the weights of the two context vectors. We devise a fusion gate to automatically assign the importance of the two-part information. To be specific, the global context vector c_g can be computed:

$$c_g = \lambda \odot c_i^s + (1 - \lambda) \odot c_i^{\omega} \tag{9}$$

$$\lambda = V_1^T \tanh(W_1 c_i^s + W_2 c_i^{\omega}) \tag{10}$$

where $V_1 \in \mathbb{R}^m$, $W_1 \in \mathbb{R}^{m \times n}$, $W_2 \in \mathbb{R}^{m \times m}$ are the weight matrices, m and n denote the dimensions of hidden units from BLSTM layer and word embeddings respectively.

Then the global context vector can be provided as additional information for the two tasks. The predictions of intent detection can be computed as below:

$$y_i^s = Softmax(W_{hy}^s(h_i + c_g)) \tag{11}$$

$$y^I = Softmax(W_{hy}^I(h_T + c^I + c_g)) \tag{12}$$

Since the two tasks have a strong relationship (Liu and Lane 2016), we also introduce the global context vector c_g for intent detection.

Experiments

Dataset. In this paper, we conduct experiments on two widely used datasets: the ATIS dataset and Snips. The ATIS dataset contains 4,978 utterances and the test set contains 893 utterances. The Snips dataset contains 13,784 utterances and the test set contains 700 utterances.

Training Procedure. Given the size of the dataset, we set the number units in the LSTM cell at 256. The default forget gate is set to 1. We use only one layer of LSTM in our model. The dropout rate is set to 0.5 during the training for regularization. We use the Adam optimization method to optimize our approach.

Experiment Results. We compare our approach with RecNN+Viterbi (Guo et al. 2014) and Attention-Based (Liu and Lane 2016) methods. The *WAIS* outperforms the two baseline methods. On the ATIS dataset, we achieve 95.87% on slot filling and 98.6% on intent detection. For the Snips dataset, we achieve 91.09% on slot filling and 98.43% on intent detection.

References

Bahdanau, D.; Cho, K.; and Bengio, Y. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Guo, D.; Tur, G.; Yih, W.-t.; and Zweig, G. 2014. Joint semantic utterance classification and slot filling with recursive neural networks. In *SLT*, 554–559. IEEE.

Hakkani-Tür, D.; Tür, G.; Celikyilmaz, A.; Chen, Y.-N.; Gao, J.; Deng, L.; and Wang, Y.-Y. 2016. Multi-domain joint semantic frame parsing using bi-directional rnn-lstm. In *Interspeech*, 715–719.

Liu, B., and Lane, I. 2015. Recurrent neural network structured output prediction for spoken language understanding. In *NIPS Workshop*.

Liu, B., and Lane, I. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. *arXiv preprint arXiv:1609.01454*.

Tur, G., and De Mori, R. 2011. *Spoken language understanding: Systems for extracting semantic information from speech*. John Wiley & Sons.

Wang, J.; Li, J.; Li, S.; Kang, Y.; Zhang, M.; Si, L.; and Zhou, G. 2018. Aspect sentiment classification with both word-level and clause-level attention networks. In *IJCAI*.

Wu, L.; Tian, F.; Zhao, L.; Lai, J.; and Liu, T.-Y. 2018. Word attention for sequence to sequence text understanding. In *AAAI*.