Slot-Gated Modeling for Joint Slot Filling and Intent Prediction

Chih-Wen Goo[†] **Guang Gao**[†] **Yun-Kai Hsu**[⋆] **Chih-Li Huo**[⋆]
Tsung-Chieh Chen[⋆] **Keng-Wei Hsu**[⋆] **Yun-Nung Chen**[†]
[†]National Taiwan University
[⋆]Institute for Information Industry
r05944049@ntu.edu.tw y.v.chen@ieee.org

Abstract

Attention-based recurrent neural network models for joint intent detection and slot filling have achieved the state-of-the-art performance, while they have independent attention weights. Considering that slot and intent have the strong relationship, this paper proposes a slot gate that focuses on learning the relationship between intent and slot attention vectors in order to obtain better semantic frame results by the global optimization. The experiments show that our proposed model significantly improves sentence-level semantic frame accuracy with 4.2% and 1.9% relative improvement compared to the attentional model on benchmark ATIS and Snips datasets respectively[1].

1 Introduction

Spoken language understanding (SLU) is a critical component in spoken dialogue systems. SLU is aiming 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 speakers intent and extracting semantic concepts as constraints for the natural language. Take a movie-related utterance as an example, *"find comedies by James Cameron"*, as shown in Figure 1. There are different slot labels for each word in the utterance, and a specific intent for the whole utterance.

Slot filling can be treated as a sequence labeling task that maps an input word sequence $\mathbf{x} = (x_1, \cdots, x_T)$ to the corresponding slot label sequence $\mathbf{y}^S = (y_1^S, \cdots, y_T^S)$, and intent detection can be seen as a classification problem to decide the intent label y^I. Popular approaches for slot filling include conditional random fields (CRF) (Raymond and Riccardi, 2007) and recurrent neural network (RNN) (Yao et al., 2014), and different classification methods, such as support vector machine (SVM) and RNN, have been applied to intent prediction.

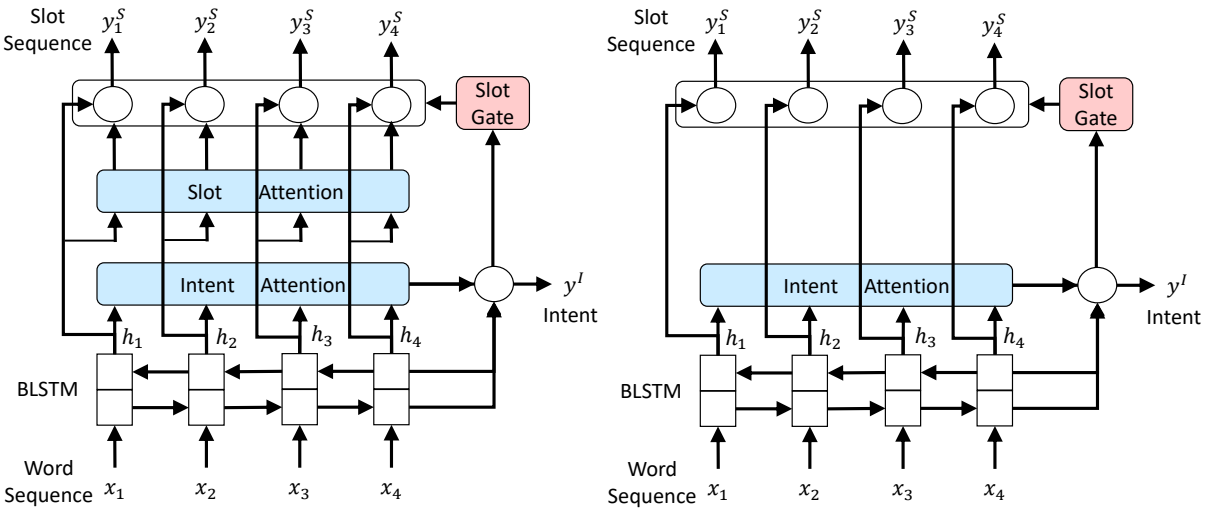
W	find	comedies	by	james	cameron
	↓	↓	↓	↓	↓
S	O	B-genre	O	B-dir	I-dir
I	find_movie				

Figure 1: An example utterance with annotations of semantic slots in IOB format (S) and intent (I), B-dir and I-dir denote the director name.

Considering that pipelined approaches usually suffer from error propagation due to their independent models, the joint model for slot filling and intent detection has been proposed to improve sentence-level semantics via mutual enhancement between two tasks (Guo et al., 2014; Hakkani-Tür et al., 2016; Chen et al., 2016). In addition, the attention mechanism (Bahdanau et al., 2014) was introduced and leveraged into the model in order to provide the precise focus, which allows the network to learn where to pay attention in the input sequence for each output label (Liu and Lane, 2015, 2016). The attentional model proposed by Liu and Lane (2016) achieved the state-of-the-art performance for joint slot filling and intent prediction, where the parameters for slot filling and intent prediction are learned in a single model with a shared objective. However, the prior work did not "explicitly" model the relationships between the intent and slots; instead, it applied a joint loss function to "implicitly" consider both cues. Because the slots often highly depend on the intent, this work focuses on how to model the explicit relationships between slots and intent vectors by introducing a slot-gated mechanism. The contributions are three-fold: 1) the proposed slot-

[1]The code is available at: https://github.com/MiuLab/SlotGated-SLU.

Proceedings of NAACL-HLT 2018, pages 753–757
New Orleans, Louisiana, June 1 - 6, 2018.



(a) Slot-Gated Model with Full Attention

(b) Slot-Gated Model with Intent Attention

Figure 2: The architecture of the proposed slot-gated models.

gated approach achieves better performance than the attention-based models; 2) the experiments on two SLU datasets show the generalization and the effectiveness of the proposed slot gate; 3) the gating results help us analyze the slot-intent relations.

2 Proposed Approach

This section first explains our attention-based RNN model and then introduces the proposed slot gate mechanism for joint slot filling and intent prediction. The model architecture is illustrated in Figure 2, where there are two different model. (a) is one with both slot attention and intent attention and (b) is another with only intent attention.

2.1 Attention-Based RNN Model

The bidirectional long short-term memory (BLSTM) model (Mesnil et al., 2015) takes a word sequence $\mathbf{x} = (x_1, \ldots, x_T)$ as input, and then generates forward hidden state $\overrightarrow{h_i}$ and 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 mapping to its corresponding slot label sequence $\mathbf{y} = (y_1^S, \ldots, y_T^S)$. For each hidden state h_i, we compute the slot context vector c_i^S as the weighted sum of LSTM's hidden states, $h_1, ..., h_T$, by the learned attention weights $\alpha_{i,j}^S$:

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

where the slot attention weights are computed as below.

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

$$e_{i,k} = \sigma(W_{he}^S h_k), \tag{3}$$

where σ is the activation function, and W_{he}^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:

$$y_i^S = \texttt{softmax}(W_{hy}^S(h_i + c_i^S)), \tag{4}$$

where y_i^S is the slot label of the i-th word in the input, and W_{hy}^S is the weight matrix. The slot attention is shown as the blue component in Figure 2(a).

Intent Prediction The intent context vector c^I can also be computed in the same manner as c^S, but the intent detection part only takes the last hidden state of BLSTM. The intent prediction is modeled similarly:

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

2.2 Slot-Gated Mechanism

This section describes the proposed slot-gated mechanism illustrated in the red part of Figure 2. The proposed slot-gated model introduces an additional gate that leverages intent context vector for modeling slot-intent relationships in order to improve slot filling performance. First, slot context vector c_i^S and intent context vector c^I are combined (c^I broadcasts in time dimension to have the

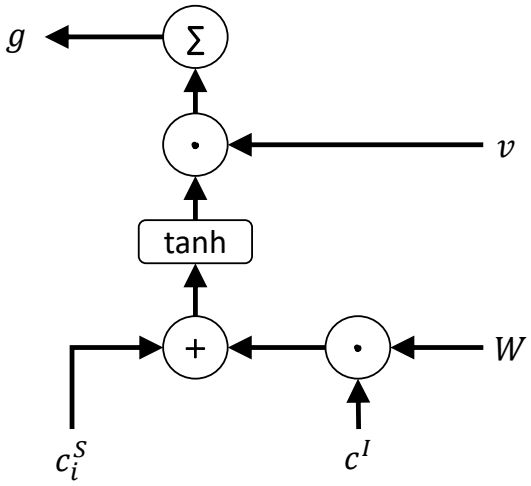


Figure 3: Illustration of the slot gate.

same shape with c_i^S) to pass through a slot gate illustrated in Figure 3:

$$g = \sum v \cdot \tanh(c_i^S + W \cdot c^I) \quad (6)$$

where v and W are trainable vector and matrix respectively. The summation is done over elements in one time step. g can be seen as a weighted feature of the joint context vector (c_i^S and c^I). We use g to weight between h_i and c_i^S to derive y_i^S and replace (4) as below:

$$y_i^S = \texttt{softmax}(W_{hy}^S(h_i + c_i^S \cdot g)). \quad (7)$$

A larger g indicates that the slot context vector and the intent context vector pay attention to the same part of the input sequence, which also infers that the correlation between the slot and the intent is stronger and the context vector is more "reliable" for contributing the prediction results.

To compare the power of the slot gate with attention mechanism, we also propose a slot-gated model with only intent attention in which (6) and (7) are reformed as (8) and (9) respectively (shown in Figure 2(b)):

$$g = \sum v \cdot \tanh(h_i + W \cdot c^I) \quad (8)$$

$$y_i^S = \texttt{softmax}(W_{hy}^S(h_i + h_i \cdot g)) \quad (9)$$

This version allows the slots and intent to share the attention mechanism.

2.3 Joint Optimization

To obtain both slot filling and intent prediction jointly, the objective is formulated as

$$p(y^S, y^I \mid \mathbf{x}) \quad (10)$$

	ATIS	Snips
Vocabulary Size	722	11,241
#Slots	120	72
#Intents	21	7
Training Set Size	4,478	13,084
Development Set Size	500	700
Testing Set Size	893	700

Table 1: Statistics of ATIS and Snips datasets.

$$= p(y^I \mid \mathbf{x}) \prod_{t=1}^{T} p(y_t^S \mid \mathbf{x})$$

$$= p(y^I \mid x_1, \cdots, x_T) \prod_{t=1}^{T} p(y_t^S \mid x_1, \cdots, x_T),$$

where $p(y^S, y^I \mid \mathbf{x})$ is the conditional probability of the understanding result (slot filling and intent prediction) given the input word sequence and is maximized for SLU.

3 Experiment

To evaluate the proposed model, we conduct experiments on the benchmark datasets, ATIS (Airline Travel Information System) and Snips. The statistics are shown in Table 1.

3.1 Setup

The ATIS (Airline Travel Information Systems) dataset (Tur et al., 2010) is widely used in SLU research. The dataset contains audio recordings of people making flight reservations. The training set contains 4,478 utterances and the test set contains 893 utterances. We use another 500 utterances for development set. There are 120 slot labels and 21 intent types in the training set.

To justify the generalization of the proposed model, we use another NLU dataset custom-intent-engines[2] collected by Snips for model evaluation. This dataset is collected from the Snips personal voice assistant, where the number of samples for each intent is approximately the same. The training set contains 13,084 utterances and the test set contains 700 utterances. We use another 700 utterances as the development set. There are 72 slot labels and 7 intent types.

Compared to single-domain ATIS dataset, Snips is more complicated mainly due to the intent diver-

[2] https://github.com/snipsco/nlu-benchmark/tree/master/2017-06-custom-intent-engines

Intent	Utterance Example
SearchCreativeWork	Find me the I, Robot television show
GetWeather	Is it windy in Boston, MA right now?
BookRestaurant	I want to book a highly rated restaurant tomorrow night
PlayMusic	Play the last track from Beyonc off Spotify
AddToPlaylist	Add Diamonds to my roadtrip playlist
RateBook	Give 6 stars to Of Mice and Men
SearchScreeningEvent	Check the showtimes for Wonder Woman in Paris

Table 2: Intents and examples in Snips dataset.

Model		ATIS Dataset			Snips Dataset		
		Slot (F1)	**Intent (Acc)**	**Sentence (Acc)**	**Slot (F1)**	**Intent (Acc)**	**Sentence (Acc)**
Joint Seq. (Hakkani-Tür et al., 2016)		94.3	92.6	80.7	87.3	96.9	73.2
Atten.-Based (Liu and Lane, 2016)		94.2	91.1	78.9	87.8	96.7	74.1
Proposed	Slot-Gated (Full Atten.)	94.8†	93.6†	82.2†	**88.8**†	**97.0**	**75.5**†
	Slot-Gated (Intent Atten.)	**95.2**†	**94.1**†	**82.6**†	88.3	96.8	74.6

Table 3: SLU performance on ATIS and Snips datasets (%). † indicates the significant improvement over all baselines ($p < 0.05$).

sity and large vocabulary. Table 2 shows the intents and associated utterance examples. Regarding the intent diversity, for example, *GetWeather* and *BookRestaurant* in Snips are from different topics, resulting larger vocabulary. In the other hand, intents in ATIS are all about flight information with similar vocabularies across them. Moreover, intents in ATIS are highly unbalanced, where *atis_flight* accounts for about 74% of training data while *atis_cheapest* appears only once. The comparison between two datasets can be found in Table 1.

In all experiments, we set the size of hidden vectors to 64, the optimizer is adam, the reported numbers are averaged over 20 runs, and the maximum epoch is set to 10 and 20 on ATIS and Snips respectively with an early-stop strategy.

3.2 Results and Analysis

We evaluate the SLU performance about slot filling using F1 score, intent prediction using accuracy, and sentence-level semantic frame parsing using whole frame accuracy. The experimental results are shown in Table 3, where the compared baselines for joint slot filling and intent prediction include the state-of-the-art sequence-based joint model using bidirectional LSTM (Hakkani-Tür et al., 2016) and attention-based model (Liu and Lane, 2016). We validate the performance improvement with statistical significance test for all experiments, where single-tailed t-test is performed to measure whether the results from the proposed model are significant better than ones from baselines. The numbers with star markers indicate that the improvement is significant with $p < 0.05$.

Table 3 shows that the proposed slot-gated mechanism with full attention significantly outperforms the baselines for both datasets, where almost all tasks (slot filling, intent prediction, and semantic frame) obtain the improvement, demonstrating that explicitly modeling strong relationships between slots and intent can benefit SLU effectively. In ATIS dataset, the proposed slot-gated model with only intent attention achieves slightly better performance with fewer parameters (from 284K to 251K). However, it does not achieve better results in Snips dataset. Considering different complexity of these datasets, the probable reason is that a simpler SLU task, such as ATIS, does not require additional slot attention to achieve good results, and the slot gate is capable of providing enough cues for slot filling. On the other hand, Snips is more complex, so that the slot attention is needed in order to model slot filling better (as well as the semantic frame results).

It is obvious that our proposed model performs better especially on sentence-level semantic frame results, where the relative improvement is around 4.1% and 1.9% for ATIS and Snips respectively.

It may credit to the proposed slot gate that learns the slot-intent relations to provide helpful information for global optimization of the joint model. In sum, for joint slot filling and intent prediction, the experiments show that leveraging explicit slot-intent relations controlled by the slot-gated mechanism can effectively achieve better sentence-level semantic frame performance due to global consideration.

4 Conclusion

This paper focuses on learning the explicit slot-intent relations by introducing a slot-gated mechanism into the state-of-the-art attention model, which allows the slot filling can be conditioned on the learned intent result in order to achieve better SLU (joint slot filling and intent detection). The experiments show that the proposed approach outperforms the baselines and can be generalized to different datasets. Also, the slot-gated model is more useful for a simple understanding task, because the slot-intent relations are stronger and easily modeled, and this paper provides the guidance of model design for future SLU work.

Acknowledgements

We would like to thank reviewers for their insightful comments on the paper. The authors are supported by the Institute for Information Industry, Ministry of Science and Technology of Taiwan, Google Research, Microsoft Research, and MediaTek Inc..

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Yun-Nung Chen, Dilek Hakanni-Tür, Gokhan Tur, Asli Celikyilmaz, Jianfeng Guo, and Li Deng. 2016. Syntax or semantics? knowledge-guided joint semantic frame parsing. In *Proceedings of 2016 IEEE Spoken Language Technology Workshop*, pages 348–355. IEEE.

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

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 *Proceedings of INTERSPEECH*, pages 715–719.

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. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. In *Proceedings of INTERSPEECH*.

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.

Christian Raymond and Giuseppe Riccardi. 2007. Generative and discriminative algorithms for spoken language understanding. In *Eighth Annual Conference of the International Speech Communication Association*.

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

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

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 *Proceedings of 2014 IEEE Spoken Language Technology Workshop*, pages 189–194. IEEE.