ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-6654-0540-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICASSP43922.2022.9747843

JOINT MULTIPLE INTENT DETECTION AND SLOT FILLING VIA SELF-DISTILLATION

Lisong Chen *Peilin Zhou* *Yuexian Zou**

ADSPLAB, School of ECE, Peking University, Shenzhen, China

ABSTRACT

Intent detection and slot filling are two main tasks in natural language understanding (NLU). These two tasks are highly related and often trained jointly. However, most previous works assume an utterance only corresponds to one intent, ignoring that it can include multiple intents. In this paper, we propose a novel Self-Distillation Joint NLU model (SDJN) for multi-intent NLU. Specifically, we adopt three orderly connected decoders and a self-distillation approach to form an auxiliary loop that establishes interrelated connections between multiple intents and slots. The output of each decoder serves as auxiliary information for the next decoder, and the auxiliary loop completes via the self-distillation. Furthermore, we formulate multiple intent detection as a weakly supervised task and handle it with multiple instance learning (MIL), which exploits token-level intent information to predict multiple intents and guide slot decoder. Experimental results indicate that our model achieves competitive performance compared to others.

***Index Terms*—** Multiple intent detection, slot filling, multiple instance learning, self-distillation.

1. INTRODUCTION

Natural language understanding (NLU) plays a pivotal role in task-oriented dialogue systems. It aims to understand user's current goal by constructing semantic frames and typically consists of two sub-tasks, intent detection (ID) and slot filling (SF) [1]. As shown in Figure 1, intent detection is often regarded as a classification task [2–4] while slot filling is regarded as a sequence tagging task [5, 6].

Taking a deeper look at the example shown in Figure 1, intent "*BookRestaurant*" is highly related to slot "*B−restaurant_type*". This observation inspires many works [7–12] to jointly model the ID and SF. Despite these works have made remarkable success, they assume that each utterance only contains one intent, ignoring the fact that users may express multiple intents in an utterance to communicate more efficiently. Therefore, it is inappropriate for directly applying aforementioned single intent NLU models due to their incapability to 1) correctly identify multiple intents from a single utterance, and 2) effectively build the interactions between multiple intents and slot labels.

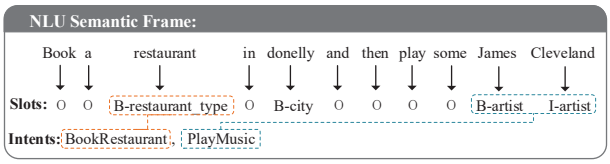

Fig. 1. NLU semantic frame.

Unlike previous single intent NLU methods, [13] first explored the multi-task framework to jointly model the multiple intent detection and slot filling. An intent-slot graph interaction layer was proposed in [14] to capture the interaction between multiple intents and each token. Though achieving promising performance, their models still suffer some issues. First, [13] adopts utterance context vectors to detect multiple intents. Such utterance-level representations may miss out on fine-grained information that could be crucial to distinguishing intents. Second, they only consider the unidirectional interaction, namely using intent to guide slot prediction, while slot can also offer important information for intent prediction.

In this paper, we propose a Self-distillation Joint NLU model (SDJN) to address the above issues. For the first issue, we argue that it is necessary to discover multiple intent signals by preserving fine-grained token-level information. However, it is hard to assign precise intent label to each token. To alleviate this problem, we reformulate multiple intent detection as a weakly supervised task and handle it with multiple instance learning (MIL) [15, 16]. In our case, we consider the tokens in the utterance as instances in MIL and the whole utterance is regarded as a bag. An aggregation layer is used to combine instance predictions and assign the overall intent labels. It allows the model to utilize token-level representation to predict intent and offer slot decoder the token-aligned intent information. For the second issue, we argue that it is essential to form an auxiliary information transmitting loop to better achieve the synergy effect between multiple intents and slots. To achieve this, we extend the basic idea of self-distillation network [17] into multi-task setting. Specifically,

This work was partially supported by Shenzhen Science & Technology Fundamental Research Program (NO: GXWD20201231165807007-20200814115301001) and National Natural Science Foundation of China (NSFC 62176008). Special acknowledgements are given to AOTO-PKUSZ JointResearch Center for Artificial Intelligence on Scene Cognition Technology Innovation for its support. * Corresponding author.

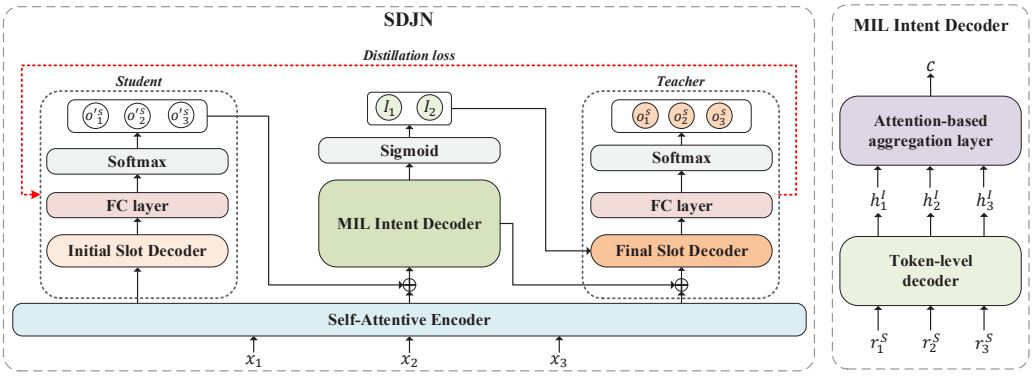

Fig. 2. The architecture of SDJN model and the MIL Intent Decoder

in our proposed SDJN, three decoders are developed and connected in series, including Initial Slot Decoder, MIL Intent Decoder, and Final Slot Decoder. The output of each decoder will serve as auxiliary information for the next one. With the intent information provided by MIL Intent Decoder, Final Slot Decoder tends to generate better slot hidden states compared to Initial Slot Decoder. Thus, we consider Final Slot Decoder as the teacher model and impart its knowledge back to Initial Slot Decoder, leading to a complete information transmitting loop. Such workflow could further establish the interrelated connections between multiple intents and slot information.

To summarize, the contributions of this paper are: (1) We formulate multiple intent detection as a weakly supervised problem and approach it with MIL where token-level information is utilized. (2) A self-distillation approach is proposed for improving joint modeling, allowing the model to exploit the interrelated connection between multiple intents and slot information in depth. (3) We evaluate our approach on two public multi-intent datasets (i.e., MixATIS and MixSNIPS [14]). The experimental results demonstrate the effectiveness of our approach, which outperforms all comparison methods.

2. APPROACH

In this section, we introduce our SDJN model in detail. The architecture of the model is illustrated in Figure 2. SDJN model consists of a shared encoder, three decoders, and a self-distillation process.

2.1. Self-attentive Encoder

Following the self-attentive encoder in [10,14], we use a bidirectional LSTM (BiLSTM) [18] with self-attention mechanism [19] as encoder to model temporal and contextual information from the utterance. The BiLSTM generates a series of context-sensitive hidden states H. Self-attention is expressive for both local and long-range dependencies, it outputs a context-aware feature A. The final encoding representation E is the concatenation of H and A, which is given by the outputs of BiLSTM and the self-attention mechanism:

$$E = H \oplus A \tag{1}$$

2.2. Initial Slot Decoder

For Initial Slot Decoder, it aims to decode initial slot fillings that will be used for guiding intents. We use a unidirectional GRU [20] for Initial Slot Decoder. The input feature of Initial Slot Decoder is $E = \{e_1, ..., e_n\}$. At every decoding step t, the decoder state $h_t'^S$ can be formalized as:

$$h_t'^S = f(h_{t-1}'^S, y_{t-1}'^S, e_t) \tag{2}$$

where $h_{t-1}'^S$ is the previous decoder state, $y_{t-1}'^S$ is the previous emitted slot prediction and e_t is the aligned encoder hidden state. The decoder state $h_t'^S$ will further be utilized to generate initial slot filling of the the utterance $O'^S = \{o_1'^S, ..., o_n'^s\}$ with softmax.

2.3. MIL Intent Decoder

In this study, we approach multiple intent detection with MIL. Under MIL, the input utterance $X = \{x_1, ..., x_n\}$ is regarded as a bag and the goal is to map each instance which is token x_t to intent label. The overall intents of the utterance will be the combination of token intents.

As shown in Figure 2, we concatenate encoding representation E and the initial slot information O'^S to form the slot reinforce representation $R^S = \{r_1^S, ..., r_n^S\}$ as the input of MIL Intent Decoder. The MIL Intent Decoder consists of two components, as shown in Figure 2: a GRU-based token-level decoder and an aggregation layer. With the decoder state h_t^I from token-level decoder, we use an attention-based prediction weighting module as aggregation layer:

$$w_t = softmax(w_d h_t^I + b) \tag{3}$$

$$c = \sum_t w_t h_t^I \tag{4}$$

The w_d is the trainable parameters and w_t is the weight for each token. The aggregation layer rewards the tokens that provide meaningful intent information. The overall intent distribution $o^I = \{o_1^I, ..., o_{N_I}^I\}$ is calculated by c, the weighted sum of hidden representation $H^I = \{h_1^I, ..., h_n^I\}$, with a sigmoid activation. Since it is a multiple intent detection task, we apply a threshold $0 < t_I < 1.0$ to obtain intents $I = \{I_1, ..., I_m\}$.

2.4. Final Slot Decoder

Final Slot Decoder is composed of two modules. A vanilla slot decoder that is identical to Initial Slot Decoder. And following [14], we incorporate the adaptive intent-slot graph interaction layer. First, we concatenate the encoding representation E with hidden representation H^I from MIL Intent Decoder to form intent reinforce representation $R^I = \{r_1^I, ..., r_n^I\}$. The vanilla slot decoder adopts r_t^I to generate decoder state h_t^S which goes through the graph interaction layer. The graph interaction layer adopts the graph attention network (GAT) [21] to model the interrelation of intents and slots at the token level. Specifically, the slot hidden state h_t^S from vanilla slot decoder and predicted multiple intents $I = \{I_1, ..., I_m\}$ are used as the initialized representation at t time step $\tilde{H}^{[0,t]} = \{h_t^S, \phi^{emb}(I_1), ..., \phi^{emb}(I_m)\}$ where $\phi^{emb}(\cdot)$ represents the embedding matrix of intents. Within the graph, the slot node representation in the l-th layer is calculated as:

$$\tilde{h}_i^{[l,t]} = \sigma(\sum_{j \in \mathcal{N}_i} \alpha_{ij}^{[l,t]} W_h^{[l]} \tilde{h}_j^{[l-1,t]}) \tag{5}$$

$\tilde{h}_i^{[l,t]}$ can be understood as node i in the l-th layer of the graph. $\mathcal{N}_i$ is the first-order neighbors of node i, W_h is the trainable weight matrix, α_{ij} is the normalized attention weight and σ represents the nonlinearity activation function. Through L-layer of adaptive intent-slot graph interaction, we adopt the final slot hidden state representation $\tilde{h}_0^{[L,t]}$ at t time step for slot filling:

$$y_t^S = softmax(W_s \tilde{h}_0^{[L,t]}) \tag{6}$$

$$o_t^S = argmax(y_t^S) \tag{7}$$

where o_t^S is the final predicted slot label of the t-th word in the utterance.

2.5. Self Distillation and Joint Training

In SDJN model, we propose a knowledge distillation approach within a joint training model by taking advantage of multi-task. The teacher model is Final Slot Decoder while the student model is Initial Slot Decoder. We select Final Slot Decoder as the teacher model for the following reasons. On the input wise, Final Slot Decoder incorporates the token-level intent information to form intent reinforce representation for better decoding. On the structure-wise, Final Slot Decoder has an adaptive intent-slot graph interaction layer to correlate intent information with slots explicitly. Therefore, Final Slot Decoder is able to generate better output. To perform the distillation method, as illustrated in Figure 2, Final Slot Decoder provides the hint for Initial Slot Decoder. A hint is defined as the output of the hidden layers from the teacher model, whose aim is to guide the student model [23]. Specifically, we leverage the hidden state from Initial Slot Decoder and Final Slot Decoder to calculate the representative distance. The relation is obtained through the computation of the MSE loss. The implicit knowledge in Final Slot Decoder imparts to Initial Slot Decoder, which induces $h_t'^S$ to fit $\tilde{h}_0^{[L,t]}$:

$$\mathcal{L}_{MSE} \triangleq -\frac{1}{n} \sum_{t=1}^{n} (h_t'^S - \tilde{h}_0^{[L,t]})^2 \tag{8}$$

The parameters of the models are optimized jointly. We use the NLLloss and BCEWithLogitsLoss for slot filling and multiple intent detection respectively. The total loss is:

$$\mathcal{L}_{total} = \alpha \cdot \mathcal{L}_{MSE} + \beta \cdot \mathcal{L}_{NLL} + \lambda \cdot \mathcal{L}_{BCE} \tag{9}$$

with three hyper-parameters α, β, and λ to balance them.

3. EXPERIMENTS

3.1. Datasets

We conduct our experiments on two public multi-intent NLU datasets. They are the cleaned version of MixATIS [14] and MixSNIPS [14]. MixATIS dataset is collected from ATIS dataset [24] and MixSNIPS dataset is from SNIPS dataset [25]. MixATIS and MixSNIPS datasets have 13162, 759, 828 utterances and 39776, 2198, 2199 utterances for training, validation, and testing respectively.

3.2. Experimental Setup

We set the self-attentive encoder hidden units as 256, dropout rate as 0.4, threshold t_i as 0.5 empirically with Adam optimizer for both datasets. For batch size, we set 16 and 64 for

Table 1. Slot filling and multiple intent detection results on two multi-intent datasets. *: the improvement of SDJN model over all baselines is statistically significant with $p < 0.05$ under t-test.

Model	MixATIS			MixSNIPS		
	Slot (F1)	Intent (Acc)	Overall (Acc)	Slot (F1)	Intent (Acc)	Overall (Acc)
Slot-Gated [8]	87.7	63.9	35.5	87.9	94.6	55.4
Bi-Model [22]	83.9	70.3	34.4	90.7	95.6	63.4
SF-ID [9]	87.4	66.2	34.9	90.6	95.0	59.9
Stack-Propogation [10]	87.8	72.1	40.1	94.2	96.0	72.9
Joint Multiple ID-SF [13]	84.6	73.4	36.1	90.6	95.1	62.9
AGIF [14]	86.7	74.4	40.8	94.2	95.1	74.2
SDJN	**88.2***	77.1*	44.6*	94.4	96.5*	75.7*
SDJN + BERT	87.5	**78.0**	**46.3**	**95.4**	**96.7**	**79.3**

Table 2. Ablation study on MixATIS. We try two types of distilled knowledge sources. (1) "Soft": soft targets with the temperature setting "$Temp$" under. (2) "Hint": the output of the hidden layers from the teacher model. "Implicit" represents that decoders only share the same encoder, while "Explicit" means the output of one decoder will serve as auxiliary information for the next Decoder.

	SDJN Components								MixATIS		
	Initial Slot Decoder	MIL Intent Decoder	Final Slot Decoder	Soft $Temp$=2	Soft $Temp$=4	Hint	Implicit	Explicit	Slot (F1)	Intent (Acc)	Overall (Acc)
(a)	✔	✔					✔		86.5	73.1	36.7
(b)	✔	✔						✔	88.0	75.9	42.0
(c)		✔	✔					✔	86.8	74.2	40.9
(d)	✔	✔	✔					✔	88.1	76.1	43.0
(e)	✔	✔	✔	✔				✔	87.5	**77.4**	43.5
(f)	✔	✔	✔		✔			✔	**88.3**	76.3	43.1
(g)	✔	✔	✔			✔		✔	88.2	77.1	**44.6**

MixATIS and MixSNIPS. The hyper-parameters of loss are empirically set as α: β: λ= 1: 0.7: 0.6 for MixATIS and α: β: λ= 1.25: 1: 1 for MixSNIPS. We evaluate the performance of slot filling with F1 score, intent detection with accuracy, and the NLU semantic frame parsing with overall accuracy that represents all metrics are correct in the utterance.

3.3. Main Results

The main results from the experiments are shown in Table 1. As we can see, our model outperforms all baselines on both datasets. For Slot (F1) score, our model outperforms the best baseline, AGIF, 1.5% and 0.2% on MixATIS and MixSNIPS, showing the advantages of adopting fine-grained multiple intent information for slot filling. For Intent (Acc), our model outperforms the top score baseline 2.7% and 0.5% on MixATIS and MixSNIPS respectively, showing the effectiveness of leveraging MIL and using slot information to guide intent prediction. For overall (Acc), the improvements are 3.8% and 1.5% on MixATIS and MixSNIPS respectively. It indicates that SDJN model can better correlate the relation between multiple intents and slots and further improve the whole NLU semantic frame parsing. The experiment results imply that our model benefits the NLU performance from the auxiliary loop and distillation method. We also investigate the effect of the pre-trained model by substituting the Self-attentive encoder into BERT [26]. The SDJN+BERT shows significant improvement, suggesting the effectiveness of a strong pre-trained model in multi-intent NLU tasks.

3.4. Ablation Study

Effect of each Decoder. We analyze how three decoders work with an ablation study on the MixATIS dataset as illustrated in Table 2 with rows (a)(b)(c)(d). The experiments are conducted by gradually adding each decoder and whether to adopt explicit interactions between slots and intents. As shown in Table 2, with the slot information from the Initial Slot decoder, row (b) outperforms row (a) on every metric significantly. The increment of 5.3% on overall accuracy shows that the MIL Intent Decoder benefits a lot from the aligned token-level slot information. Comparing row (b) and row (d), with the Final Slot decoder adding on to row (d), the results again show improvements. This ablation study suggests that each decoder contributes improvements and considering the cross-impact between slots and intents brings better results.

Effect of Distillation Approach. To further examine the effectiveness of our distillation approach, we show the ablation study on rows (d)(e)(f)(g) in Table 2. As shown, all rows with distillation approach (rows (e)(f)(g)) outperform row (d) from 0.1% up to 1.6% in overall accuracy. It suggests the gain of the proposed self-distillation approach. We find it interesting that while both using soft targets as a knowledge source, row (e) with $Temp$=2 shows better performance in multiple intent detection and overall accuracy while row (f) with $Temp$=4 shows better performance in slot filling. Comparing the distilled knowledge source, row (g) which uses hint as knowledge source outperforms rows (e)(f) that both use the soft target as a knowledge source 1.1% and 1.5% respectively in overall accuracy. We argue that the soft targets mainly rely on the output of the last layer of the decoder and fail to address the intermediate-level supervision which is important for representation learning. On the other hand, hint offers intermediate-level supervision and enables the model to learn better representation. Therefore, row (g) has a more balanced performance in slot filling and multiple intent detection, which further leads to a better result in overall accuracy.

4. CONCLUSIONS

In this work, we propose a Self-distillation Joint NLU model by taking advantage of multi-task information. In addition, we approach multiple intent detection as a weakly supervised task with MIL. Experiments on two public multi-intent datasets show that SDJN achieves performance gains over strong baselines.

5. REFERENCES

[1] G. Tur, *Spoken Language Understanding: Systems for Extracting Semantic Information from Speech*, Ph.D.

thesis, Morgan & Claypool, 2011.

[2] Corinna Cortes and Vladimir Vapnik, "Support-vector networks," *Machine learning*, 1995.

[3] Christopher M Bishop, *Pattern recognition and machine learning*, springer, 2006.

[4] Ruhi Sarikaya, Geoffrey E Hinton, and Bhuvana Ramabhadran, "Deep belief nets for natural language call-routing," in *ICASSP*. IEEE, 2011.

[5] John Lafferty, Andrew McCallum, and Fernando CN Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," 2001.

[6] Kaisheng Yao, Baolin Peng, Yu Zhang, Dong Yu, Geoffrey Zweig, and Yangyang Shi, "Spoken language understanding using long short-term memory neural networks," in *2014 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2014.

[7] Bing Liu and I. Lane, "Attention-based recurrent neural network models for joint intent detection and slot filling," in *INTERSPEECH*, 2016.

[8] Chih-Wen Goo, Guang-Lai 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.

[9] E. Haihong, Peiqing Niu, Zhongfu Chen, and Meina Song, "A novel bi-directional interrelated model for joint intent detection and slot filling," 2019.

[10] Libo Qin, W. Che, Yangming Li, Haoyang Wen, and T. Liu, "A stack-propagation framework with token-level intent detection for spoken language understanding," in *EMNLP/IJCNLP*, 2019.

[11] Peilin Zhou, Zhiqi Huang, Fenglin Liu, and Yuexian Zou, "Pin: A novel parallel interactive network for spoken language understanding," *ArXiv*, 2020.

[12] Zhiqi Huang, Fenglin Liu, Peilin Zhou, and Yuexian Zou, "Sentiment injected iteratively co-interactive network for spoken language understanding," in *ICASSP*, 2021.

[13] Rashmi Gangadharaiah and Balakrishnan Narayanaswamy, "Joint multiple intent detection and slot labeling for goal-oriented dialog," in *NAACL*, 2019.

[14] Libo Qin, Xiao Xu, Wanxiang Che, and Ting Liu, "AGIF: An adaptive graph-interactive framework for joint multiple intent detection and slot filling," in *Findings of the Association for Computational Linguistics: EMNLP 2020*. Association for Computational Linguistics.

[15] J. Keeler and D. Rumelhart, "A self-organizing integrated segmentation and recognition neural net," in *NIPS*, 1991.

[16] Yunjie Ji, H. Liu, Bolei He, X. Xiao, Hua Wu, and Yanhua Yu, "Diversified multiple instance learning for document-level multi-aspect sentiment classification," in *EMNLP*, 2020.

[17] Linfeng Zhang, Jiebo Song, Anni Gao, J. Chen, Chenglong Bao, and Kaisheng Ma, "Be your own teacher: Improve the performance of convolutional neural networks via self distillation," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[18] S Hochreiter and J Schmidhuber, "Long short-term memory," *Neural Computation*, 1997.

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

[20] Kyunghyun Cho, Bart Van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *Computer Science*, 2014.

[21] Petar Velickovic, Guillem Cucurull, A. Casanova, A. Romero, P. Liò, and Yoshua Bengio, "Graph attention networks," *ArXiv*, 2018.

[22] Yu Wang, Yilin Shen, and Hongxia Jin, "A bi-model based rnn semantic frame parsing model for intent detection and slot filling," *ArXiv*, 2018.

[23] A. Romero, Nicolas Ballas, S. Kahou, Antoine Chassang, C. Gatta, and Yoshua Bengio, "Fitnets: Hints for thin deep nets," *CoRR*, 2015.

[24] C. T. Hemphill, J. Godfrey, and G. Doddington, "The atis spoken language systems pilot corpus," in *HLT*, 1990.

[25] A. Coucke, A. Saade, Adrien Ball, Théodore Bluche, A. Caulier, D. Leroy, Clément Doumouro, Thibault Gisselbrecht, F. Caltagirone, Thibaut Lavril, Maël Primet, and J. Dureau, "Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces," *ArXiv*, 2018.

[26] J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *NAACL*, 2019.