End to End Spoken Language Understanding Using Partial Disentangled Slot Embedding

Tan Liu,Wu Guo*
* University of Science and Technology of China
National Engineering Laboratory for Speech and Language Information Processing,Hefei, China
E-mail: liutan@mail.ustc.edu.cn,guowu@ustc.edu.cn

***Abstract*—Spoken language understanding (SLU) has switched from pipeline approaches to end-to-end (E2E) ones recently. For most E2E approaches, neural networks are adopted to extract embeddings from the audio signals directly for final intents prediction. In this paper, we explore this method for intent classification on Fluent Speech Commands (FSC) dataset, where intents are formed as combinations of three slots (action, object, and location). The information of different slots will be entangled with each other in the extracted embeddings, which sometimes brings about errors in the prediction of the current slot. To address this problem, we propose partial disentangled slot embedding (PDSE) method through adversarial training. Results show that the proposed method can achieve an error rate of 0.53%, which outperforms the baseline with over 35.3% error rate reduction.**

***Index Terms*—end to end, spoken language understanding, disentangled embedding.**

I. Introduction

Spoken language understanding (SLU) systems aim to infer the intents of spoken utterances,which have attracted very much attention in recent years. Most state-of-the-art SLU systems involve two major sub-systems [1,2]. The first system is an automatic speech recognizer (ASR), whose responsibility is to transcribe the spoken utterances to texts. The second one is an intent detection system based on the transcribed texts. Since speech recognizers will inevitably make some mistakes during decoding, it is difficult for an intent detection module to yield correct intent from erroneous ASR outputs [3,4,5]. With the widespread of end-to-end (E2E) methods in pattern recognition, the end-to-end SLU recently receives attention as a promising research direction for better intents prediction.

A natural approach to deal with E2E SLU is to use deep neural networks (DNN) to encode the variable-length acoustic signal (or human-designed features) into fixed-dimension embeddings. Lugosch et al [6] uses a stack of multiple recurrent neural network (RNN) layers to encode the variable-length utterances into fixed-dimension embeddings which are fed to an intent classifier. Transformer [7] which adopts non-recurrent self-attention is also used as the encoder to deal with E2E SLU [8]. Chen et al [9] feed the softmax probabilities over graphemes produced by a pre-trained acoustic model (AM) component to the subsequent SLU component to predict the intents of the utterances. In this paper, we employ the Conformer [10] as the backbone network, which can be looked on as an update version of Transformer with combination of self-attention and convolution.

To further improve the performances, pre-training strategies have been applied in SLU systems. Lugosch et al [6] pre-trains the lower RNN layers with ASR targets (words and phonemes) to provide better feature to the subsequent RNN layers. Following [6], Wang et al [11] first pre-trains the AM component with ASR targets, specifically phonemes, then applies a BERT-like approach to pre-train the subsequent SLU component which takes the phoneme posteriors output by the AM component as input.

For most SLU tasks, the intent of a spoken utterance is considered as a tuple of slots [12]. The SLU systems are expected to predict the intent based on the value of each slot, so SLU can also be regarded as a multi-label classification task [13]. E2E SLU systems usually squeeze the related information into an embedding for each slot [12], which is fed to the corresponding classifier to predict the value. As for each slot embedding, there exists the information of other slots, sometimes the information of other slot helps to predict the value of the current slot [12], and sometimes the information of other slot covers the information of the current slot, which results in errors for the prediction of the current slot. It is a straightforward idea to decouple the information of different slots, and disentangled embedding has been explored in various domains, such as speaker verification [14,15], face recognition [16,17] and so on.

In this paper, we propose a partial disentangled slot embedding (PDSE) method which leverages disentangled embedding on the slot embedding. Different from conventional methods that use whole slot embedding to conduct disentanglement, we only use a continuous portion of the slot embedding to conduct disentanglement. While in the inference stage, the whole slot embedding is used to predict the slot value. We evaluate the proposed approach on the public Fluent Speech Commands (FSC) dataset. The proposed PDSE can obtain an error rate of 0.53%, which outperforms the baseline system with over 35.3% error rate reduction.

The rest of this paper is organized as follows. Section II introduces our baseline E2E SLU system. Section III provides detailed description of the proposed PDSE method. Section IV presents the experimental setup and results. Finally, the discussion and conclusion are presented in Section V.

II. Baseline E2E SLU System

We use a Conformer-based SLU system as our baseline, as depicted in Fig.2.

A. Conformer

Conformer is proposed for ASR [10], which combines self-attention with convolution in a cascading way. The architecture of a Conformer block is shown in Fig.1. As the figure shows, the Conformer block has a macaron-like architecture, where the multi-head self-attention module and the convolution module are sandwiched by two feed-forward modules with half-step residual connections.

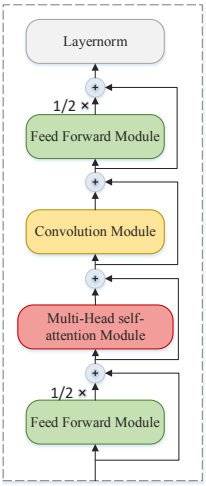

Fig. 1: The architecture of a Conformer block.

B. Baseline SLU system

We carry out SLU on the Fluent Speech Commands (FSC) dataset, whose intents are formed as combinations of three slots (action, object, and location). The baseline system is mainly composed of four components, an encoder, an action decoder, an object decoder and a location decoder, as depicted in Fig.2. Most of the components are composed of the conformer blocks.

The encoder starts with a Convolution Subsamping module which is followed by a linear layer and a dropout layer. Then a stack of multiple Conformer blocks is deployed after the dropout layer. We set a decoder for each slot to predict the corresponding slot value. As for the action and location decoders, we first employ Conformer blocks to learn action- and location-specific information from the output of the encoder respectively. Then a max-pooling layer is applied to squeeze the information into an embedding which is fed to the subsequent linear layers to predict the values. The object decoder has same architecture with the other two decoders except that the Conformer block is replaced with a CNN block for better performace. The CNN block has same architecture with the Conformer block except that the multi-head self-attention module is removed. The system aims to accurately

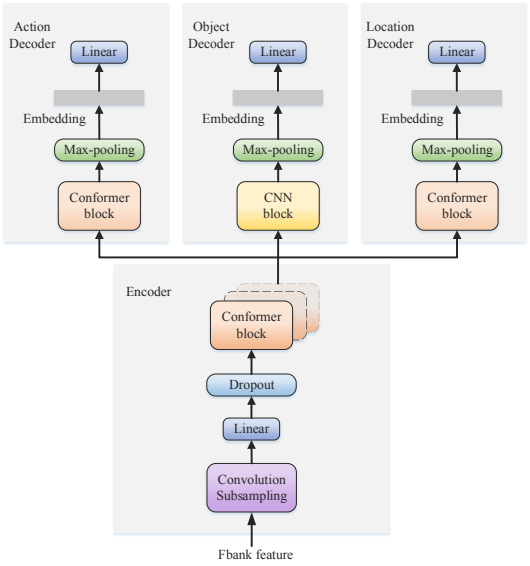

Fig. 2: The architecture of the baseline system.

predict the value of all slots, therefore the overall loss is the summation of the three slot-specific Cross Entropy losses, which is represented as:

$$\boldsymbol{L_{CE}} = L_a + L_o + L_l \tag{1}$$

Where L_a , L_o and L_l are the Cross Entropy losses for action, object and location classification respectively,$\boldsymbol{L_{CE}}$ represents the summation of all these losses.

III. Proposed Approach

A. Partial Disentangled Slot Embedding (PDSE)

The baseline system in Fig.2 can extract discriminative embedding for each slot on the FSC dataset. However, the information of action and object are entangled with each other in the extracted slot embeddings. We employ partial disentangled slot embedding (PDSE) method on the action decoder and the object decoder of the baseline system to obtain refined embeddings. Taking the object decoder for example, the architecture of the object decoder with the proposed PDSE is shown in Fig.3. We select the first quarter of the object embedding as the PDSE, denoted as e_D, which is the red part of the embedding shown in Fig.3, For the object decoder with PDSE, two extra classifiers are deployed in the object decoder, the slot-specific classifier C_{sp} and the adversarial classifier C_{adv}, both of which take e_D as input. C_{sp} is to predict the value of object based on e_D,we adopt cross entropy loss for C_{sp}, which can be written as:

$$L_{sp} = CE\left(y_o, \mathrm{softmax}\left(C_{sp}\left(e_D\right)\right)\right) \tag{2}$$

Where y_o is the true label of object and $CE()$ represents cross entropy function.

C_{adv} is to predict the value of action based on e_D , while the encoder and the object decoder are trained to enable e_D

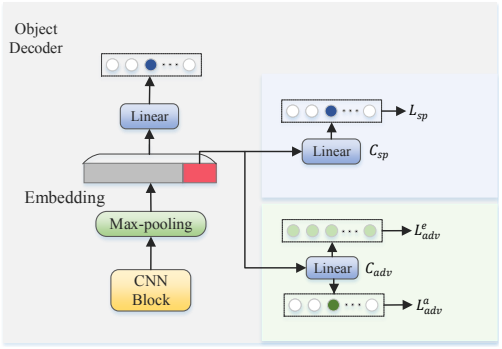

Fig. 3: The architecture of the object decoder with PDSE.

fool the C_{adv} through adversarial learning [18], so that C_{adv} outputs the same probability over each action class. Hence, the information of action in e_D is removed. Similarly, we adopt cross entropy loss to train C_{adv} by:

$$P_{adv} = \text{softmax}\left(C_{adv}\left(e_D\right)\right) \tag{3}$$

$$L^a_{adv} = CE\left(y_a, P_{adv}\right) \tag{4}$$

Where P_{adv} is the softmax probability over action predicted by C_{adv}, y_a is the true label of actions, L^a_{adv} is the cross entropy loss for action classification. Note that the gradient of L^a_{adv} only propagates back to C_{adv} in model training.

Since the encoder and the object decoder are trained to enable e_D fool the C_{adv}, the target distribution should be uniform over all action categories, which equal to $\frac{1}{N_a}$, and N_a is the total number of all action categories. We represent the target distribution as $Q_{avg} = \left\{\frac{1}{N_a}, \frac{1}{N_a}, \cdots \frac{1}{N_a}\right\}$, and adopt Kullback-Leibler (KL) divergence as the loss function which is calculated by:

$$L^e_{adv} = \sum Q_{avg}(i) \log\left(\frac{Q_{\text{avg}}\left(i\right)}{P_{adv}(i)}\right) \tag{5}$$

Note that the gradient of L^e_{adv} doesn't update any parameters of C_{adv}. By combining L^a_{adv} and L^e_{adv},the information of action in e_D is eliminated.

During the inference stage, both C_{sp} and C_{adv} are abandoned, the whole object embedding is fed into the original object classifier to predict the value of object, which utilizes the information of action meanwhile preserves the information of object from being covered by the information of action.

As for the action decoder, we select the first half dimension of the action embedding as the partial disentangled action embedding. Similar with the object decoder, there are two extra classifiers added in the action decoder, and the PDSE in action decoder is not expected to contain the information of object.

B. Total Loss

In summary, the system with the proposed PDSE involves multiple losses that consist of the essential intent prediction loss $\boldsymbol{L_{CE}}$, the loss $\boldsymbol{L_{sp}}$, the adversarial losses $\boldsymbol{L^a_{adv}}$ and $\boldsymbol{L^e_{adv}}$. Therefore, the overall loss with a weighted combination of them is as below:

$$\boldsymbol{L_{total}} = \boldsymbol{L_{CE}} + \alpha \boldsymbol{L_{sp}} + \beta\left(\boldsymbol{L^a_{adv}} + \boldsymbol{L^e_{adv}}\right) \tag{6}$$

Where α, β are the weight parameters and set as $\alpha = 0.1$, $\beta = 0.1$ in our experiments. $\boldsymbol{L_{sp}}$ is the summation of the L_{sp} in the object decoder and the action decoder, $\boldsymbol{L^a_{adv}}$ is the summation of the L^a_{adv} in the object decoder and the action decoder, $\boldsymbol{L^e_{adv}}$ is the summation of the L^e_{adv} in the object decoder and the action decoder. The overall goal is to minimize $\boldsymbol{L_{total}}$. As mentioned above, the PDSE system is trained in a way of adversarial learning, $\boldsymbol{L^a_{adv}}$ is minimized alone while all other losses are minimized simultaneously.

IV. Experiments

A. Dateset

We use the publicly available Fluent Speech Commands (FSC) dataset to evaluate our proposed approach. The dataset consists of a number of 16KHZ single-channel spoken utterances stored in the format of *.wav*, each of which is a speech command that might be used for smart home or virtual assistant applications. The dataset contains about 19 hours of speech with a total of 30043 spoken utterances from 97 different speakers. Each utterance is labeled with three slots: action, object, and location, and there are total 6, 14 and 4 unique values for action, object and location respectively. The combination of three slots is used as the intent of the utterance. The dataset is split into train, valid and test set, the detailed information is shown in Tabel I.

TABLE I: The detailed information about FSC.

Set	Speakers	Utterances	Hours
Train	77	23132	14.7
Valid	10	3118	1.9
Test	10	3793	2.4

B. Implementation Details

We adopt 108-dimensional filterbank features (36 filterbank features, delta coefficients, and delta-delta coefficients) without mean and variance normalization as the input of the system. The system configurations are as follows. As for the baseline system in Fig. 2, the encoder uses a stack of 2 conformer blocks. All Conformer blocks used in the encoder have identical configuration. The multi-head self-attention module of each Conformer block has 8 attention heads and sets the attention dim to 552. The convolution module of each conformer block sets the kernel size to 11 and the expansion factor to 2. We apply dropout with rate of 0.1 in each residual unit of each Conformer block. The Convolution module of the CNN blocks in the object decoder contains two pointwise convolution layers and a depthwise convolution layer [19] with kernel size of 31. The conformer blocks of the action and location are same as those of the encoder.

C. Experimental Results

We use the error rate to evaluate the performance and the results are listed in Table II. Compared with the contrastive systems [11], our baseline can achieve a decent performance with error rate of 0.82%. As shown in Table II, the PDSE_A_O system with the proposed PDSE used on the object and action decoders achieves the best performance with the error rate of 0.53%, which illustrates the effectiveness of the PDSE. We also build a PDSE_O system where the PDSE is only applied on the object decoder. The result of the PDSE_O is listed in the last row of Table II. The PDSE_A_O system is only slightly superior over the PDSE_O.

TABLE II: Comparison of error rate between different approaches on FSC.

Model	Error Rate(%)
Radfar et al [8]	2.4
Lugosch et al [6]	1.2
Wang et al [11]	0.8
Baseline	0.82
PDSE_A_O	**0.53**
PDSE_O	0.58

D. Ablation Study

As introduced in Section III, the losses involved in the PDSE consist of $\boldsymbol{L_{CE}}$ (in Eq1), L_{sp} (in Eq2), L^a_{adv} (in Eq4), L^e_{adv} (in Eq7). In the following experiments, we provide an ablation study to evaluate the contribution of each loss. For simplicity, we employ the PDSE_O to conduct all ablation experiments. We build three ablation systems trained with different combinations of the above losses and the results are listed in Table III. As shown in Table III, PDSE_O outperforms all ablation systems, which illustrates that the system can only achieve the best performance when all the above losses are used together.

TABLE III: the performances of different ablation systems.

Model	Error Rate(%)
Baseline	0.82
Baseline+L_{sp}	0.74
Baseline+L^a_{adv}+L^e_{adv}	0.87
Baseline+L_{sp}+L^e_{adv}	0.81
PDSE_O	**0.58**

In addition, the dimension of PDSE has a great impact on the performance. For simplicity, we use different dimensions of the e_D in the PDSE_O to explore how the dimension of PDSE affect the performance. The results are listed in Table IV. As shown in Table IV, the best performance is obtained with $e_D = 138$, that is the quarter of the whole object embedding. As mentioned above, the information of action in e_D is eliminated, when the dimension of e_D increases (as listed in the first two lines in Table IV), the performance drops obviously, the reason is that the whole object embedding contains little information of action which helps to predict the object value. Finally, we select a quarter of the whole object embedding as e_D in the object decoder.

TABLE IV: The performance of the PDSE_O with different dimension of e_D.

Dimension of e_D	Error Rate(%)
552	0.84
276	0.92
138	**0.58**
69	0.69

V. Conclusion

In this paper, we propose partial disentangled slot embedding (PDSE) and apply it on an end to end SLU system. We selects a continuous portion of the original slot embedding as PDSE which is trained to preserve the information of the current slot from being covered by the information of other slots through adversarial training. We evaluate the proposed approach on the public FSC dataset and the PDSE shows significant improvement on the performance of the baseline. In additional, we conduct a number of ablation experiments to explore the contribution of each loss involved and the influence of the dimension of PDSE.

VI. Acknowledge

This work was partially funded by the National Natural Science Foundation of China (Grant No. U1836219).

References

[1] Coucke, Alice, et al, "Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces," *arXiv preprint arXiv:1805.10190,* 2018.

[2] Mesnil, Grégoire, et al, "Using recurrent neural networks for slot filling in spoken language understanding," *IEEE/ACM Transactions on Audio, Speech, and Language Processing 23.3 (2014): 530-539,* 2014.

[3] Serdyuk, Dmitriy, et al, "Towards end-to-end spoken language understanding," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE,* 2018.

[4] Simonnet, Edwin, et al, "Simulating ASR errors for training SLU systems," *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018),* 2018.

[5] Huang, Chao-Wei, and Yun-Nung Chen, "Learning asr-robust contextualized embeddings for spoken language understanding," *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE,* 2020.

[6] Lugosch, Loren, et al, "Speech model pre-training for end-to-end spoken language understanding," *arXiv preprint arXiv:1904.03670,* 2019.

[7] Vaswani, Ashish, et al, "Attention is all you need," *Advances in neural information processing systems,* 2017.

[8] Radfar, Martin, Athanasios Mouchtaris, and Siegfried Kunzmann, "End-to-end neural transformer based spoken language understanding," *arXiv preprint arXiv:2008.10984 (2020). in Proc. Interspeech, 2020, pp. 5036–5040,* 2020.

[9] Chen, Yuan-Ping, Ryan Price, and Srinivas Bangalore, "Spoken language understanding without speech recognition," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE,* 2018.

[10] Gulati, Anmol, et al, "Conformer: Convolution-augmented transformer for speech recognition," *arXiv preprint arXiv:2005.08100* 2020.

[11] Wang, Pengwei, et al, "Large-scale unsupervised pre-training for end-to-end spoken language understanding," *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE,* 2020.

[12] Palogiannidi, Elisavet, et al, "End-to-end architectures for ASR-free spoken language understanding," *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE,* 2020.
[13] Bhosale, Swapnil, et al, "End-to-End Spoken Language Understanding: Bootstrapping in Low Resource Scenarios," *Interspeech,* 2019.
[14] Tai, Jianwei, et al, "SEEF-ALDR: A Speaker Embedding Enhancement Framework via Adversarial Learning based Disentangled Representation," *Annual Computer Security Applications Conference,* 2020.
[15] Peri, Raghuveer, et al, "Robust speaker recognition using unsupervised adversarial invariance," *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE,* 2020.
[16] Tran, Luan, Xi Yin, and Xiaoming Liu, "Disentangled representation learning gan for pose-invariant face recognition," *Proceedings of the IEEE conference on computer vision and pattern recognition,* 2017.
[17] Jiang, Zi-Hang, et al, "Disentangled representation learning for 3D face shape," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition,* 2019.
[18] Lowd, Daniel, and Christopher Meek, "Adversarial learning," *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining,* 2005.
[19] Chollet, François, "Xception: Deep learning with depthwise separable convolutions," *Proceedings of the IEEE conference on computer vision and pattern recognition,* 2017.