ROBUST SPOKEN LANGUAGE UNDERSTANDING WITH UNSUPERVISED ASR-ERROR ADAPTATION

Su Zhu, Ouyu Lan and Kai Yu

Key Laboratory of Shanghai Education Commission for Intelligent Interaction and Cognitive Engineering
SpeechLab, Department of Computer Science and Engineering
Brain Science and Technology Research Center
Shanghai Jiao Tong University, Shanghai, China
{paul2204,blue-0-0-,kai.yu}@sjtu.edu.cn

ABSTRACT

Robustness to errors produced by automatic speech recognition (ASR) is essential for Spoken Language Understanding (SLU). Traditional robust SLU typically needs ASR hypotheses with semantic annotations for training. However, semantic annotation is very expensive, and the corresponding ASR system may change frequently. Here, we propose a novel unsupervised ASR-error adaptation method, obviating the need of annotated ASR hypotheses. It only requires semantically annotated transcripts for the slot-tagging task and the transcripts paired with hypotheses for an input sentence reconstruction task. In this method, feature encoders which share part of the parameters are exploited to enforce the tasks in a similar feature space. Therefore, the transcript side slot-tagging model can be transferred to ASR hypotheses side easily. Experiments show that the proposed approach can yield significant improvement over strong baselines, and achieve performance very close to the oracle system.

Index Terms— Spoken Language understanding, ASR-error robustness, adversarial adaptation

1. INTRODUCTION

The spoken language understanding (SLU) module is a key component of spoken dialogue system (SDS), parsing user's utterances into corresponding semantic concepts. For example, the utterance *"Show me flights from Boston to New York"* can be parsed into *(fromloc.city_name=Boston, toloc.city_name=New York)* [1]. Typically, the SLU problem is regarded as a slot tagging task. We focus on slot tagging in this paper as well. With sufficient in-domain data and deep learning models (e.g. recurrent neural networks, bidirectional long-short memory network), statistical methods have achieved high performance in the slot tagging task recently [2, 3, 4].

Most of the previous work about SLU only focuses on utterance transcripts by ignoring Automatic Speech Recognition (ASR) errors. The SLU system trained on transcripts would get a significant decrease in performance when used on ASR hypotheses [5]. To improve system robustness, the traditional method requires sufficient data of labelled ASR hypotheses for training. However, SLU annotation on hypotheses is a labor-intensive and time-consuming task [6]. Moreover, the slot-tagging annotation on the hypotheses might be renewed when ASR system changes, since the hypotheses may change as well. Tur et al. [7] investigated slot tagging on ASR hypotheses with semantically annotated bins of word confusion networks. This method automatically creates a annotation on hypotheses by an ASR alignment trick for training data, while the automatic alignment may create wrong data samples.

Decreasing SLU performance on ASR hypotheses stems from a mismatch of semantic distribution between training and evaluation. We propose an unsupervised adaptation method to transfer slot-tagging model trained on the transcripts to hypotheses. In this method, semantically labelled transcripts are exploited for the slot-tagging task. Utterance transcripts and hypotheses are used for an unsupervised task (e.g. language modelling). The slot-tagging task shares part of parameters with the unsupervised task so that it could switch from transcribed sentences to ASR hypotheses. Moreover, an adversarial training trick [8] is used to force the shared parameters task-invariant.

We are the first to investigate unsupervised ASR-error adaptation problem for slot tagging without annotation on ASR hypotheses. It would potentially be useful for the deployment of commercial dialogue systems. We propose an adversarial adaptation method for ASR-error adaptation problem in SLU, exploiting pairs of the utterance transcript and ASR hypotheses. The experimental results show that our method outperforms the strong baselines significantly.

The corresponding author is Kai Yu. This work has been supported by the National Key Research and Development Program of China under Grant No.2017YFB1002102, and the China NSFC projects (No. 61573241). Experiments have been carried out on the PI supercomputer at Shanghai Jiao Tong University.

978-1-5386-4658-8/18/$31.00 ©2018 IEEE

The rest of the paper is organized as follows. The next section describes the framework of unsupervised ASR-error adaptation method. Experiments are presented in section 3, followed by relations to prior works and conclusions.

2. UNSUPERVISED ASR-ERROR ADAPTATION

In this section, the details of unsupervised ASR-error adaptation are given. This method only requires semantically annotated transcripts for slot tagging and raw transcripts paired with ASR hypotheses for the ASR-error adaptation, obviating the annotation on the hypotheses. The corresponding data sources used in this method are described as below:

- tag: utterance transcripts with the annotations of slot-tag sequence.
- $tscp$: utterance transcripts.
- asr: utterance hypotheses given by ASR system.

2.1. BLSTM Encoder

We use a bidirectional LSTM (BLSTM) [9, 10] model as the encoder of input. Let e_w denote the word embedding for each word w, and $\oplus$ denote the vector concatenation operation. The encoder reads the input sentence $\mathbf{w} = (w_1, w_2, ..., w_T)$ and generates T hidden states of BLSTM:

$$h_i = \overleftarrow{h_i} \oplus \overrightarrow{h_i}; \quad \overleftarrow{h_i} = f_l(\overleftarrow{h_{i+1}}, e_{w_i}); \quad \overrightarrow{h_i} = f_r(\overrightarrow{h_{i-1}}, e_{w_i})$$

where $\overleftarrow{h_i}$ is the hidden vector of the backward pass in BLSTM and $\overrightarrow{h_i}$ is the hidden vector of the forward pass in BLSTM at time i, f_l and f_r are LSTM units [11] of the backward and forward passes respectively.

Following the notation in [8], we write the entire operation as a mapping BLSTM_Θ (Θ refers to the parameters):

$$(h_1...h_T) = \text{BLSTM}_\Theta(w_1...w_T)$$

2.2. Unsupervised Adaptive Learning

In the unsupervised ASR-error adaptation, we exploit SLU annotation on utterance transcripts instead of ASR hypotheses. Our approach closely follows the previous work on unsupervised neural domain adaptation [12, 13, 8]. The major difference is that we make the encoders of slot tagging and unsupervised reconstruction tasks different. Therefore we have four BLSTM encoders, as shown in Figure 1:

- Θ^{tag}: produces features which are specific for slot tagging task.
- Θ^{tscp}: produces features which are specific for the transcripts side unsupervised task.
- Θ^{asr}: produces features which are specific for the ASR hypotheses side unsupervised task.
- Θ^{shd}: produces task-invariant features.

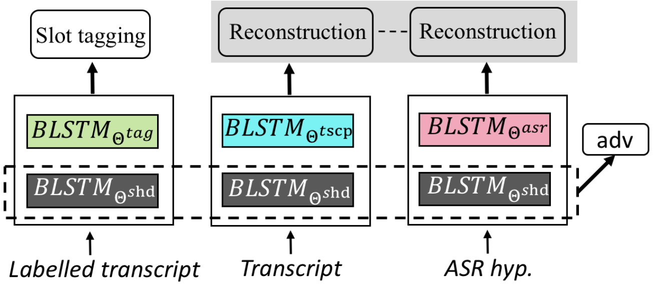

Fig. 1. Architecture of the proposed method which includes three tasks: transcript side slot-tagging, transcript side and ASR-hypotheses side reconstructions. The framework contains four BLSTM encoder for feature learning, one of which is shared by three tasks and the others are private for each task.

The architecture of our method is illustrated in Figure 1. The word embeddings are shared in theses encoders. Now we define three loss functions for the ASR-error adaptation: (1) slot tagging, (2) reconstruction (unsupervised), (3) adversarial domain classification.

2.2.1. *Transcript side tagging loss*

The most important objective is to minimize the slot tagging error on labelled transcripts. Let $\mathbf{w} = (w_1...w_T)$ be an utterance transcript labelled with labels $\mathbf{y} = (y_1...y_T)$. We produce

$$(h_1^{tag}...h_T^{tag}) \leftarrow \text{BLSTM}_{\Theta^{tag}}(\mathbf{w})$$
$$(h_1^{shd}...h_T^{shd}) \leftarrow \text{BLSTM}_{\Theta^{shd}}(\mathbf{w})$$

Then we define the probability of slot tag y for the i-th word as

$$z_i = W_{tag}\overline{h}_i + b_{tag}; \quad p(y|\overline{h}_i) \propto exp([z_i]_y)$$

where $\overline{h}_i = h_i^{tag} \oplus h_i^{shd}$, W_{tag} and b_{tag} are the weighted matrix and bias vector of output layer respectively. Let Ω^{tag} denote $\{W_{tag}, b_{tag}\}$. The tagging loss function is given by a negative log-likelihood

$$L^{tag}(\Theta^{tag}, \Theta^{shd}, \Omega^{tag}) = -\sum_{(\mathbf{w},\mathbf{y})} \sum_i p(y_i|\overline{h}_i)$$

2.2.2. *Reconstruction loss*

We also ground feature learning by reconstructing encoded sentence in an unsupervised way. By adding sentence reconstruction task for both transcripts and ASR hypotheses, it enforces BLSTM encoders to be close in the parameter space. Kim et al. used an attention-based sequence-to-sequence (S2S) [14] that fully re-generates the input sentence [8]. In contrast, we propose to use a bidirectional language modelling (BLM) for producing of the input sentence, which is more efficient.

Let $\mathbf{w} = (w_1 ... w_T)$ be a sentence in data source $d \in \{tscp, asr\}$. With the relevant encoders, we have

$$(h_1^d ... h_T^d) \leftarrow \text{BLSTM}_{\Theta^d}(\mathbf{w})$$
$$(h_1^{shd} ... h_T^{shd}) \leftarrow \text{BLSTM}_{\Theta^{shd}}(\mathbf{w})$$

The concatenated vector $h_i^l = \overleftarrow{h_i^d} \oplus \overleftarrow{h_i^{shd}}$ is fed into a simple feed-forward neural network (FFN) with only one layer to predict the last word, and $h_i^r = \overrightarrow{h_i^d} \oplus \overrightarrow{h_i^{shd}}$ is fed into another FFN to predict the next word. We use Ω^{rec} to denote the parameters of these two FFNs. The reconstruction loss is given by the negative log likelihood

$$L^{rec}(\Theta^d, \Theta^{shd}, \Omega^{rec}) = -\sum_{\mathbf{w} \in D_d} \sum_i (p(w_{i-1}|h_i^l) + p(w_{i+1}|h_i^r))$$

where $d \in \{tscp, asr\}$, w_0 is a sentence start tag <*s*> and w_{T+1} is a sentence end tag <*/s*>.

2.2.3. Adversarial task classification loss

The intuition is that the more task-invariant features we have, the easier it is to benefit from the transcript side training when decoding on the ASR hypotheses side. Following the previous work [8], we use random prediction training to force the shared encoder task-invariant. This adversarial training method makes the shared BLSTM encoder to be ASR-error robust by incorporating with the above reconstruction task.

Let $\mathbf{w} = (w_1, ..., w_T)$ be a sentence in data sources $\{tag, tscp, asr\}$. With the shared encoder, we have the hidden states

$$(h_1^{shd} ... h_T^{shd}) \leftarrow \text{BLSTM}_{\Theta^{shd}}(\mathbf{w})$$

where h_i^{shd} is fed into a task classifier which is a single layer FFN. Let Ω^{adv} denote the parameters of this classifier. Therefore, the adversarial loss can be formulated as

$$L^{adv}(\Theta^{shd}, \Omega^{adv}) = -\sum_{\mathbf{w}} \sum_i p(t_i | h_i^{shd})$$

where t_i is randomly set to be tag, $tscp$, asr with equal probability.

2.2.4. Joint objective function

For the unsupervised ASR-error adaptation, we optimize

$$\begin{aligned} L =& L^{tag}(\Theta^{tag}, \Theta^{shd}, \Omega^{tag}) + L^{rec}(\Theta^{tscp}, \Theta^{shd}, \Omega^{rec}) \\ &+ L^{rec}(\Theta^{asr}, \Theta^{shd}, \Omega^{rec}) + L^{adv}(\Theta^{shd}, \Omega^{adv}) \end{aligned} \quad (1)$$

In decoding stage, we use the encoder Θ^{tag} and the slot tagger Ω^{tag} on the ASR hypotheses to predict the slot-tags.

3. EXPERIMENTS

3.1. Dataset

In order to evaluate our proposed model, we conduct experiments on a dataset which is collected from a Chinese commercial dialogue system in the domain of car navigation. It contains 9008 utterances in total, as shown in Table 1. We randomly select 60% of the training data for model training, another 20% for validation and the remaining 20% to be test set. The training and validation sets for slot tagging are labelled on transcripts, and the test set is labelled on ASR top-hypotheses. For building an oracle baseline, the training and validation sets are also labelled on ASR top-hypotheses. There are 13 different slots included. For slot tagging, we follow the popular In/Out/Begin (IOB) representation as label.

data partitions		#sentence	CER
train+valid	labelled transcripts (*tag*)	7,205	21.52
	transcripts (*tscp*)	7,205	
	ASR top-hyp. (*asr*)	7,205	
test	labelled ASR top-hyp.	1,803	23.47

Table 1. Sentence numbers (#) and CER (Chinese Character Error Rate of speech recognition) of different data partitions in the dataset.

3.2. Experimental Settings

We do slot tagging on Chinese character level since Chinese word segmentation may introduce alignment errors in a closed dialogue domain. The 'word' below refers to Chinese character. We deal with unseen words in the test set by marking any words with only one single occurrence in the training set as $\langle unk \rangle$. For BLSTM, we set the dimension of word embeddings to 100 (the vocabulary size is 1391) and the number of hidden units to 200. Only the current word is used as input without any context. For training, the network parameters are randomly initialized in accordance with the uniform distribution (-0.2, 0.2). The *dropout* with a probability of 0.5 is applied to the non-recurrent connections during the training stage. Maximum norm for gradient clipping is set to 5. We use Adam optimizer following the suggested parameter setup in [15].

The learning rate is initialized to be 0.001. We keep the learning rate for 100 epochs and save the parameters that give the best performance on the validation set. The metric used is F_1-score calculated using CoNLL evaluation script. [1]

We investigate our method with different combinations of the loss functions in Section 2.2. For comparison, we also set and implement several strong baselines and even oracle systems as follows:

[1] http://www.cnts.ua.ac.be/conll2000/chunking/output.html

- Baseline$_1$: It is trained and validated on the transcript data with SLU annotation, using only transcript side slot-tagging loss L^{tag} in Eqn. (1).
- Baseline$_2$: Traditional robust SLU method creates annotation on ASR hypotheses by the alignment trick [7]. Similarly, a word alignment between transcript and ASR top-hypotheses is performed by using the text alignment tool in Kaldi [2] to deliver slot tags from labelled transcript to top-hypothesis. With the auto-annotated ASR hypotheses and the transcript data, another baseline model is trained by using only loss L^{tag} in Eqn. (1).
- Oracle$_1$: It is trained and validated on the data of ASR top-hypothesis with SLU annotation, only supervised by L^{tag} in Eqn. (1).
- Oracle$_2$: It is trained on both SLU annotated transcripts and ASR top-hypotheses, only supervised by L^{tag} in Eqn. (1).
- Domain adaptation: The unsupervised domain adaptation [8] is applied to the ASR-error adaptation, which assumes $\Theta^{tag} = \Theta^{tscp}$ in Eqn. (1). This method treats the data of transcripts (including slot tags) as a source domain, and unlabelled ASR hypotheses as the target domain.

3.3. Experimental Results and Analysis

In this section, we evaluate our systems with different combinations of loss functions and compare them with several baseline systems. From Table 2 we can see a gap (2.75%) between slot tagging systems trained on transcripts ($Baseline_1$) and ASR top-hypotheses ($Oracle_1$). By combing the labelled transcripts, $Oracle_2$ obtains an additional improvement (0.99%) over $Oracle_1$. By incorporating the auto-annotated ASR hypotheses, the performance of $Baseline_2$ decreases, because the word alignment may cause wrong data samples for slot-tagging.

In our systems, bidirectional language modelling (BLM, row (h)) outperforms other two reconstruction tasks: W2W [3] (row (f)) and S2S (row (g), as indicated in Section 2.2.2). Our system with BLM (row (h)) also achieves a significantly better result (significant level 96%) than the domain adaptation method [8] (row (e)), which may benefit from separated encoders for tagging and reconstruction tasks (i.e. Θ^{tag} is different with Θ^{tscp}).

Compared to the system with BLM (row (h)), the separated reconstruction models (row (i)) cause a decrease in F1-score. By incorporating the adversarial task classification loss, out method can achieve the best performance (row (j)) outperforming all baseline systems and being very close (-0.53%) to the oracle system, due to the parameters sharing among tasks of transcript side slot-tagging, transcript and ASR hypotheses reconstructions.

system		Rec.	F1-score
(a)	Oracle$_1$ (L^{tag})	-	84.65
(b)	Oracle$_2$ (L^{tag})	-	85.64
(c)	Baseline$_1$ (L^{tag})	-	81.90
(d)	Baseline$_2$ (L^{tag})	-	78.71
(e)	Domain Adaptation	S2S	82.52
(f)	$L^{tag} + L^{rec}$	W2W	82.82
(g)	$L^{tag} + L^{rec}$	S2S	83.31
(h)	$L^{tag} + L^{rec}$	BLM	**84.87**
(i)	$L^{tag} + L^{rec}$	BLM^{sep}	84.02
(j)	$L^{tag} + L^{rec} + L^{adv}$	BLM	**85.11**

Table 2. Comparison of the oracle systems, baselines, and our method. Different reconstruction tasks are also evaluated: W2W [3], S2S, BLM (as indicated in Section 2.2.2). BLM^{sep} refers to the separated reconstruction models of BLM.

4. RELATION TO PRIOR WORK

Our work benefits from the recent success of domain adaptation in the neural network. An adversarial training method for unsupervised domain adaptation was firstly proposed in the area of computer vision [12, 13]. This method splits the model parameters into two parts: domain-specific features which are private and domain-invariant features which are shared. The domain-invariant parameters are adversarially trained by reversing gradient to make domain classifier poor and domain agnostic. The adversarial domain adaptation method is also applied in sentence classification [16] and spoken language understanding (SLU) [8]. We are the first to investigate the ASR-error adaptation for SLU by adversarial adaptation method and propose the unsupervised task-adaptation architecture for robust SLU. Meanwhile, we incorporate a novel bidirectional language modelling (via forward and backward respectively) as the unsupervised task.

5. CONCLUSION

In this paper, an unsupervised ASR-error adaptation method is proposed for slot tagging task to improve the robustness of SLU model. We newly incorporate adversarial task adaptation method and bidirectional language modelling to transfer a SLU model from transcript to ASR hypotheses. This approach doesn't require semantically annotated ASR hypotheses, which can save the workload of data annotation and has potential advantages for deployment of the commercial system. Finally, the experimental results show that our method can achieve a significant improvement over the strong baselines, while still being resilient to ASR-errors.

[2] http://kaldi-asr.org/doc/align-text_8cc.html

[3] W2W simply reproduces the current word at each time step by $p(w_i|h_i^l, h_i^r)$ as indicated in Section 2.2.2.

6. REFERENCES

[1] Roberto Pieraccini, Evelyne Tzoukermann, Zakhar Gorelov, J-L Gauvain, Esther Levin, C-H Lee, and Jay G Wilpon, "A speech understanding system based on statistical representation of semantics," in *Acoustics, Speech, and Signal Processing, 1992. ICASSP-92., 1992 IEEE International Conference on*. IEEE, 1992, vol. 1, pp. 193–196.

[2] Gakuto Kurata, Bing Xiang, Bowen Zhou, and Mo Yu, "Leveraging sentence-level information with encoder lstm for semantic slot filling," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, Texas, November 2016, pp. 2077–2083, Association for Computational Linguistics.

[3] Ngoc Thang Vu, "Sequential convolutional neural networks for slot filling in spoken language understanding," in *17th Annual Conference of the International Speech Communication Association (InterSpeech)*, 2016.

[4] Bing Liu and Ian Lane, "Attention-based recurrent neural network models for joint intent detection and slot filling," in *17th Annual Conference of the International Speech Communication Association (InterSpeech)*, 2016.

[5] Grégoire Mesnil, Xiaodong He, Li Deng, and Yoshua Bengio, "Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding.," in *INTERSPEECH*, 2013, pp. 3771–3775.

[6] Srinivas Bangalore, Dilek Hakkani-Tür, and Gokhan Tur, "Introduction to the special issue on spoken language understanding in conversational systems," *Speech Communication*, vol. 48, no. 3, pp. 233–238, 2006.

[7] Gökhan Tür, Anoop Deoras, and Dilek Hakkani-Tür, "Semantic parsing using word confusion networks with conditional random fields.," in *INTERSPEECH*, 2013, pp. 2579–2583.

[8] Young-Bum Kim, Karl Stratos, and Dongchan Kim, "Adversarial adaptation of synthetic or stale data," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, vol. 1, pp. 1297–1307.

[9] Mike Schuster and Kuldip K Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[10] Alex Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*, Springer Berlin Heidelberg, 2012.

[11] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[12] Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan, "Domain separation networks," in *Advances in Neural Information Processing Systems*, 2016, pp. 343–351.

[13] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky, "Domain-adversarial training of neural networks," *Journal of Machine Learning Research*, vol. 17, no. 59, pp. 1–35, 2016.

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

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

[16] Pengfei Liu, Xipeng Qiu, and Xuanjing Huang, "Adversarial multi-task learning for text classification," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vancouver, Canada, July 2017, pp. 1–10, Association for Computational Linguistics.