Multi-Domain Adversarial Learning for Slot Filling in Spoken Language Understanding

Bing Liu[1], Ian Lane[1,2]
[1]Electrical and Computer Engineering, Carnegie Mellon University
[2]Language Technologies Institute, Carnegie Mellon University
{liubing, lane}@cmu.edu

Abstract

The goal of this paper is to learn cross-domain representations for slot filling task in spoken language understanding (SLU). Most of the recently published SLU models are domain-specific ones that work on individual task domains. Annotating data for each individual task domain is both financially costly and non-scalable. In this work, we propose an adversarial training method in learning common features and representations that can be shared across multiple domains. Model that produces such shared representations can be combined with models trained on individual domain SLU data to reduce the amount of training samples required for developing a new domain. In our experiments using data sets from multiple domains, we show that adversarial training helps in learning better domain-general SLU models, leading to improved slot filling F1 scores. We further show that applying adversarial learning on domain-general model also helps in achieving higher slot filling performance when the model is jointly optimized with domain-specific models.

1 Introduction

Spoken language understanding (SLU) system is a critical component in today's task-oriented conversational agents. A typical pipeline of SLU system is to first classify the domain of a user's query, and then identify user's intent and extract semantic frames from the user's natural language utterance. These three tasks are usually referred to as domain classification, intent identification, and slot filling. Many statistical learning methods have been developed for these tasks over the past years. Domain classification and intent detection can be treated as sequence classification problems, and classifiers like support vector machines and deep neural network [1, 2] can be applied. Slot filling can be framed as a sequence tagging task. Popular methods for solving sequence tagging include maximum entropy Markov models (MEMMs) [3], conditional random fields (CRFs) [4], and recurrent neural networks (RNNs) [5, 6, 7].

Many of the previously proposed SLU models focus on solving tasks in individual domains. Annotating semantic tags for each individual task domain is both financially costly and non-scalable. In this work, we focus on learning common knowledge and representations that can be shared across multiple domains for slot filling in SLU. This shared knowledge can help domain-specific SLU model trained with limited in-domain data to reduce the amount of annotated data required for developing a new domain. We want to train a domain-general model that focuses only on learning common patterns across multiple domains, and jointly optimize it with a domain-specific model that preserves in-domain knowledge for slot filling.

A straightforward manner to learn cross-domain representations is to train a single model on a combination of the data from all domains. Such model, however, may still learn disjoint domain-specific features due to the very different data distributions in different domains. To enforce the

model to discover features and structures that are common across multiple domains, we propose to apply domain adversarial learning in training the domain-general model. The intuition behind is that good common representations across domains are the ones that with which a system cannot recover the domain of the original inputs [8]. Once obtaining such a domain-general model, we ensemble it with domain-specific models for best slot filling performance in each task domain. We show in the experiment results that domain-general model trained on all source data can benefit from domain adversarial learning in achieving better slot filling F1 scores. Adversarial learning on domain-general model also brings benefit when the model is jointly optimized with domain-specific models.

The remainder of the paper is organized as follows. In section 2, we discuss related work on multi-domain slot filling. In section 3, we present our proposed model and learning method. In Section 4, we discuss the experiment setup and the results. In section 5 we conclude the work.

2   Related Work

Multi-domain spoken language understanding has been actively studied in literature. Hakkani-Tur et al. [9] proposed a multi-domain SLU model using Bi-directional LSTM. They train the multi-domain model by simply using data from all the domains and let the data from each domain to reinforce each other. Similarly, Jaech et al. [10] designed an LSTM based multi-domain model that has shared word embeddings and LSTM parameters across domains and separated softmax layer for each domain. The word embeddings and LSTM parameters capture the across-domain knowledge. In our proposed model, we introduce an additional domain adversarial loss in training domain-general model using data from all sources. This enforces the model to learn common features that are general across multiple domains.

Another line of research that is closely related to our work is domain adaptation for language understanding. Kim et al. [11] proposed a neural generalization of the feature augmentation domain adaptation method [12]. Their model uses an LSTM to captures global patterns by training on data across all domains, and adapt it to one domain with another LSTM trained on only in-domain data. Our proposed method differs in how we train the domain-general model by introducing domain adversarial loss. Adversarial learning has also been applied in domain adaptation. Zhang et al. [13] used adversarial training to learn domain-invariant features for document classification task. In their model, an adversarial domain classification loss is subtracted directly from the overall system loss function. All model parameters are thus discouraged to make correct domain prediction. In our method, we enforce the domain classifier to take the best classification efforts, and only discourage the underlying network parameters to make correct domain prediction so as to force them to learn domain-invariant features. This is closer to the training of generative adversarial networks [14].

3   Proposed Method

3.1   Bi-LSTM Slot Filling Model

Let $\mathbf{w} = (w_1, ..., w_T)$ represent an utterance with T words, and let $\mathbf{y} = (y_1, ..., y_T)$ represent the aligned slot label sequence with the same length as the input word sequence. We use a bidirectional LSTM (bi-LSTM) to encode the sequence of words to continuous representations. The bi-LSTM encoder outputs at each step t is obtained by concatenating the forward and backward LSTM state outputs: $h_i = [\overrightarrow{h_i}, \overleftarrow{h_i}]$. Based on the bi-LSTM encoder output at each step t, a probability distribution $P(y_t)$ over all slot labels for the word input is produced:

$$P(y_t \,|\, \mathbf{w}; \theta_y) = \text{SlotLabelDist}(h_t) \tag{1}$$

where SlotLabelDist is a multi-layer perceptron (MLP) parameterized by θ_y with softmax outputs over all slot label candidates. We train the bi-LSTM slot filling model by optimizing the parameters θ to minimize the cross-entropy loss L_y of the predicted and true distributions for slot label at each step:

$$\min_{\theta_s, \theta_y} L_y = \min_{\theta_s, \theta_y} -\frac{1}{T} \sum_{t=0}^{T} \log P(y_t^* | \mathbf{w}; \theta_s, \theta_y) \tag{2}$$

where y_t^* is the ground truth slot label at step t, and θ_s represents the parameters of the bi-LSTM and the word embeddings.
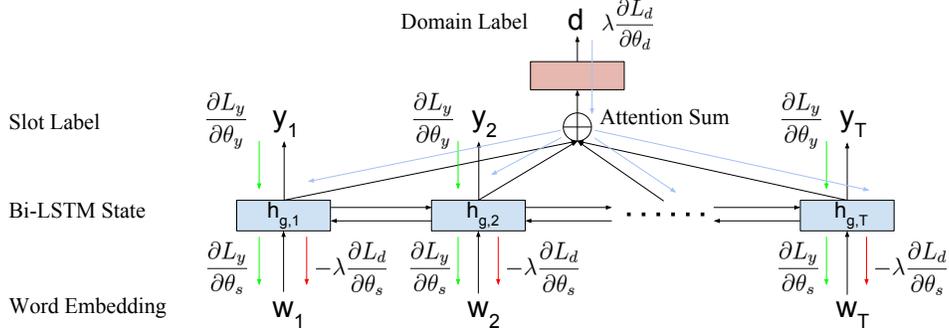
Figure 1: Domain-general slot filling model with adversarial learning.

3.2 Domain Adversarial Learning

We collect annotated data sets in multiple task domains, and train slot filling models for each domain using only in-domain data. We further train a domain-general model using a combination of the data from all sources. To enforce the model to learn domain-invariant features, we apply adversarial learning when training the domain-general model. We introduce a domain classifier whose job is to identify the domain of the input utterance by taking in the representations from the bi-LSTM state outputs. We train the domain classifier to minimize the domain classification loss. On the other hand, we update the parameters of the network underlying the domain classifier to maximize the domain classification loss, which works adversarially towards the domain classifier. We thus expect the model to learn features and structures that are general across domains.

Attention Domain Classifier   The domain classifier takes in the representation of an utterance and estimate its domain. A probability distribution $P(d)$ over all domain labels is generated by:

$$P(d\,|\,\mathbf{w};\theta_d) = \text{DomainLabelDist}(c) \tag{3}$$

where c is an attention sum over encoder state outputs $(h_1, ..., h_T)$:

$$c = \sum_{t=1}^{T} \alpha_t h_t \tag{4}$$

and

$$\alpha_t = \frac{\exp(e_t)}{\sum_{k=1}^{T} \exp(e_k)}, \quad e_t = g(h_t) \tag{5}$$

g a feed-forward neural network with a single output node.

Adversarial Training   We train the domain classifier by optimizing its parameter set θ_d to minimize the cross-entropy loss L_d of the predicted and true domain class distributions:

$$\max_{\theta_s} \min_{\theta_d} L_d = \max_{\theta_s} \min_{\theta_d} - \log P(d^*|\mathbf{w};\theta_s,\theta_d) \tag{6}$$

where d^* is the ground truth domain class label.

In updating the network parameters θ_s underlying the domain classifier, we reverse the gradient $\frac{\partial L_d}{\partial \theta_s}$ during back-propagation with the aim to force the model to produce utterance representations that maximize domain classification confusion. This operation is performed by connecting the domain classifier with underlying network via a gradient reversal layer [15]. The overall loss function for slot filling and adversarial domain classification is then calculated by:

$$L = L_y + \lambda L_d \tag{7}$$

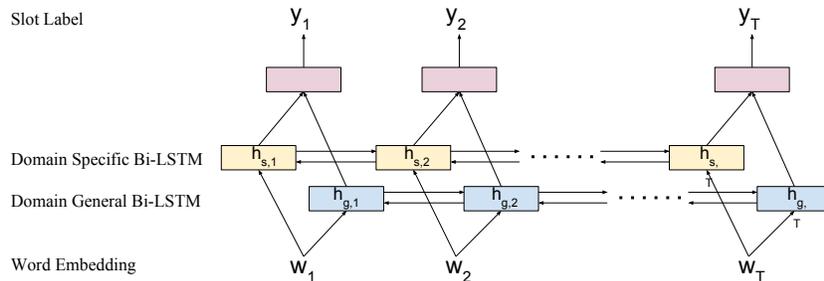where λ is the coefficient for the adversarial domain classification loss.

3

Figure 2: Joint domain-specific and domain-general slot filling model.

3.3 Joint Optimization

Figure 2 shows the network architecture for the joint domain-specific and domain-general slot filling model. At the input layer, embedding of a word is connected to both domain-specific and domain-general Bi-LSTMs. Outputs of the two models are fed to an MLP. A softmax function on top of the MLP produces the probability distribution over all slot labels. We pre-train the domain-specific and domain-general Bi-LSTMs separately. During joint optimization, we fix the parameters of the two Bi-LSTMs, and only optimize the MLP parameters in the output layer.

4 Experiments

4.1 Data Sets

We evaluate the proposed model and learning methods on data sets across multiple domains. Each data set consists of semantically tagged training and test corpus in standard BIO format. Table 1 shows the data set statistics.

ATIS   Airline Travel Information Systems (ATIS) data set is widely used in SLU research. It consists of spoken queries on flight related information.

MIT Restaurant Corpus[1]   The MIT restaurant data set consists of spoken queries on restaurant searching and booking.

MIT Movie Corpus   The MIT movie data set consists of two sets of semantically tagged corpora. The eng corpus contains simple queries, and the trivia10k13 corpus contains more complex queries. There are a few shared slot labels between these two corpora.

Table 1: Statistics of Data Sets.

Datasets	ATIS	MIT Rest.	MIT Mov. eng	MIT Mov. trivia10k13	Combined
Train set size	4978	7660	9775	7816	30229
Test set size	893	1521	2443	1953	6810
Vocab size	572	4166	7481	12145	16049
Slot label size	127	17	25	25	191

4.2 Training Settings

We set the number of units in LSTM cell as 128. The default LSTM forget gate bias is set to 1. Word embedding size is set as 128. We randomly initialize the word embedding matrix and fine-tune it with other model parameters during model training. Dropout rate 0.5 is applied to the non-recurrent connections during model training for regularization. Maximum norm for gradient clipping is set to 5. We perform mini-batch training (batch size 16) using Adam optimization with initial learning rate of 1e-3.

---

[1]The MIT semantic tagging corpus can be downloaded from: https://groups.csail.mit.edu/sls/downloads

4.3 Results and Analysis

Table 2: Slot Filling F1 scores of the proposed method for data sets in different task domains.

Model	ATIS	MIT Rest.	MIT Mov. eng	MIT Mov. trivia10k13	Comb.
Deep LSTM [16]	95.08	-	-	-	-
RNN-EM [17]	95.25	-	-	-	-
Encoder-labeler LSTM [18]	95.40	-	-	-	74.41
Attention Bi-LSTM [6]	95.75	-	-	-	-
BLSTM-LSTM (focus) [19]	**95.79**	-	-	-	-
Dom-Spec	95.55	72.42	83.43	63.64	-
Dom-Gen	94.09	**74.25**	82.95	63.34	76.03
Dom-Gen-Adv (λ=0.01)	**94.51**	73.87	**83.03**	**63.51**	**76.55**
Dom-Gen-Adv (λ=0.1)	93.88	73.98	82.31	62.83	76.01
Dom-Gen-Adv (λ=1.0)	84.65	62.47	75.05	52.82	66.66
Joint Dom Spec & Gen	95.62	**74.47**	84.87	65.16	-
Joint Dom Spec & Gen-Adv (λ=0.01)	**95.63**	74.23	**85.33**	**65.33**	-
Joint Dom Spec & Gen-Adv (λ=0.1)	95.52	74.36	85.32	64.95	-
Joint Dom Spec & Gen-Adv (λ=1.0)	95.52	73.57	84.26	64.38	-

Table 2 shows the slot filling evaluation results. We compare the performance of domain-specific models, domain-general models, and joint domain-specific and domain-general models on data sets over multiple domains. For adversarial learning on domain-general models, we compare the model performance by applying different scales of domain adversarial losses.

We first compare the performance between our domain-specific models and domain-general models. Our domain-specific (Dom-Spec) model achieves near state-of-the-art slot filling F1 score on ATIS comparing to recently published results. Our first domain-general model (Dom-Gen) is trained on a combination of data sets across multiple domains. This Dom-Gen model outperforms Dom-Spec on MIT restaurant data set, but obtains lower F1 scores on ATIS and the two MIT movie data sets. By further applying domain adversarial training (with λ=0.01), the Dom-Gen-Adv model achieves improved F1 scores on three out of four data sets over the Dom-Gen model. Using larger scale of domain adversarial loss is not beneficial, resulting in decreased F1 scores across all four data sets. Although domain adversarial learning helps the domain-general models to achieve better slot filling F1 scores by learning common features and structures across different domains, their performance is still below that of the domain-specific models. By looking into the prediction mistakes made by domain-general models on ATIS data set, we find that the model assigns label from other domains which is semantically similar but not matched to the ground truth label in ATIS. This label inconsistency issue across multiple domains makes it not feasible to train a general model by simply combining data from different domains.

We further evaluate the model performance by jointly optimizing the domain-specific and domain-general model. Results in Table 2 illustrate that the joint training model consistently outperforms the domain-specific models and the domain-general models. The joint training model achieves larger gain on the MIT movie data set, improving the F1 score by 1.90% and 1.69% respectively on the eng corpus and the trivia10k13 corpus over their domain-specific models. Similar to the results in domain-general model evaluation, adversarial learning on domain-general model also brings benefit to the joint model optimization. Joint model training with adversarial learning achieves advanced slot filling performance comparing to non-adversarial joint model training on three out of four data sets.

5 Conclusions

In this work, we propose applying domain adversarial training in learning common features and representations that can be shared among multiple domains for slot filling task in spoken language understanding. Domain-general models producing such shared representations can be combined with domain-specific models to reduce the amount of data required for training SLU model in a new

task domain. Experimental results show that domain-general models trained on a combination of data from all domains can benefit from adversarial learning in achieving higher slot filling F1 scores. We further show that applying adversarial training to domain-general model also helps to achieve advanced slot filling performance when the model is jointly optimized with domain-specific models.

References

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

[2] Yoon Kim. Convolutional neural networks for sentence classification. In *EMNLP*, 2014.

[3] Andrew McCallum, Dayne Freitag, and Fernando CN Pereira. Maximum entropy markov models for information extraction and segmentation. In *ICML*, 2000.

[4] Christian Raymond and Giuseppe Riccardi. Generative and discriminative algorithms for spoken language understanding. In *INTERSPEECH*, pages 1605–1608, 2007.

[5] Grégoire Mesnil, Yann Dauphin, Kaisheng Yao, Yoshua Bengio, Li Deng, Dilek Hakkani-Tur, Xiaodong He, Larry Heck, Gokhan Tur, Dong Yu, et al. Using recurrent neural networks for slot filling in spoken language understanding. *in TASLP*, 2015.

[6] Bing Liu and Ian Lane. Attention-based recurrent neural network models for joint intent detection and slot filling. In *Interspeech 2016*, pages 685–689, 2016.

[7] Yun-Nung Chen, Dilek Hakkani-Tür, Gökhan Tür, Jianfeng Gao, and Li Deng. End-to-end memory networks with knowledge carryover for multi-turn spoken language understanding. In *INTERSPEECH*, pages 3245–3249, 2016.

[8] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

[9] Dilek Hakkani-Tür, Gökhan Tür, Asli Celikyilmaz, Yun-Nung Chen, Jianfeng Gao, Li Deng, and Ye-Yi Wang. Multi-domain joint semantic frame parsing using bi-directional rnn-lstm. In *INTERSPEECH*, pages 715–719, 2016.

[10] Aaron Jaech, Larry Heck, and Mari Ostendorf. Domain adaptation of recurrent neural networks for natural language understanding. In *INTERSPEECH*, 2016.

[11] Young-Bum Kim, Karl Stratos, and Ruhi Sarikaya. Frustratingly easy neural domain adaptation. In *COLING*, pages 387–396, 2016.

[12] Hal Daume III. Frustratingly easy domain adaptation. In *Proceedings of ACL*, 2007.

[13] Yuan Zhang, Regina Barzilay, and Tommi Jaakkola. Aspect-augmented adversarial networks for domain adaptation. *arXiv preprint arXiv:1701.00188*, 2017.

[14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *in NIPS*, 2014.

[15] 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*, 17(59):1–35, 2016.

[16] Kaisheng Yao, Baolin Peng, Yu Zhang, Dong Yu, Geoffrey Zweig, and Yangyang Shi. Spoken language understanding using long short-term memory neural networks. In *in SLT*, 2014.

[17] Baolin Peng, Kaisheng Yao, Li Jing, and Kam-Fai Wong. Recurrent neural networks with external memory for spoken language understanding. In *Natural Language Processing and Chinese Computing*. Springer, 2015.

[18] Gakuto Kurata, Bing Xiang, Bowen Zhou, and Mo Yu. Leveraging sentence-level information with encoder lstm for semantic slot filling. In *Proceedings of EMNLP*, 2016.

[19] Su Zhu and Kai Yu. Encoder-decoder with focus-mechanism for sequence labelling based spoken language understanding. In *ICASSP*, 2017.