# SEMI-SUPERVISED TRAINING USING ADVERSARIAL MULTI-TASK LEARNING FOR SPOKEN LANGUAGE UNDERSTANDING

*Ouyu Lan, Su Zhu 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
{blue-0-0-,paul2204,kai.yu}@sjtu.edu.cn

## ABSTRACT

Spoken language understanding (SLU) usually requires human semantic annotation on collected data, but the process is expensive. In order to make better use of unlabeled data for robust SLU, we propose an adversarial multi-task learning method by merging a bidirectional language model (BLM) and a slot tagging model (STM). As a secondary objective, the BLM is used to learn generalized and unsupervised knowledge with abundant unlabeled data and improve the performance of STM on unseen data samples. We construct a shared space for both tasks and independent private spaces for each task respectively. Additional adversarial task discriminator is also used to obtain more task-independent sharing information. Experiments show that the proposed approaches achieve the state-of-the-art performance on the small scale ATIS benchmark and significantly improve the semi-supervised performance on a large-scale dataset.

***Index Terms***— Spoken language understanding, Multi-task learning, Semi-supervised learning, Adversarial task discriminator

## 1. INTRODUCTION

The Spoken Language Understanding (SLU) module is a key component of the goal-oriented spoken dialogue system (SDS), parsing the users' utterances into the corresponding semantic concepts. For example, the sentence "*Show me flights from Boston to New York*" can be parsed into (*fromloc.city_name=Boston, toloc.city_name=New York*)[1]. Typically, it is regarded as a slot filling task, assigning one predefined semantic slot tag to each word in the utterance [2].

Recent research about statistical slot filling in SLU has focused on recurrent neural network (RNN) [3] and its extensions, such as long-short memory networks (LSTM) [4], encoder-decoder model [5, 6, 7], etc. These traditional methods require large amounts of labeled data to achieve a good performance. However, it is difficult to get sufficient in-domain labeled data for training because the data annotation is labor-intensive and time-consuming [8]. When an existing domain expands or a new one is created, only limited data are available for supervised learning. In recent years, more and more applications of SDS have been released along with the development of the mobile internet, e.g. Apple Siri, Amazon Alexa, Google Home, Microsoft Cortana etc. Lack of supervised data generally results in a locally optimal solution. To alleviate the localization, semi-supervised learning can be used to access to those unseen inputs efficiently.

For semi-supervised learning in SLU, most effective methods exploit unlabeled data by language modeling. Celikyilmaz et al [9] adopted pseudo labels [10] of unlabeled data. Similar works like [11, 12] adopted an additional language model to improve the accuracy of sequence labeling tasks. Rei et al shared the whole hidden layers for all tasks [11], and Matthew et al separated them completely [12]. But both of them only showed the performance on named entity recognition, chunking, and POS-tagging tasks.

Inspired by the success of shared-private models [13], we propose an adversarial multi-task learning method for SLU which learns generalized and unsupervised knowledge and regularizes the slot tagging model. The motivation is to tune the slot tagging model by integrating general language information from unlabeled data. Specifically, a bidirectional language model (BLM) and a slot tagging model (STM) are combined by a shared space and two task-specific private spaces. The BLM learns underlying general patterns of semantic and syntactic composition [11] with abundant unsupervised data, while the STM obtains supervised knowledge with limited labeled data. The shared space is trained for both tasks. Furthermore, we investigate an adversarial task discriminator as a rival to the shared space. The aim of the task discriminator is to figure out which task are the shared features trained for at each time. In order to confuse the task discriminator, the shared space is forced to extract task-invariant knowledge and jettison task-specific information. The task discriminator

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.

is applied on word-level or sentence-level. Unlike Chen et al who trained their models only for Chinese word segmentation task by supervised learning on multiple segmentation criteria with the same data source, we adopt distinctive training objectives, methods and data sources for each task.

The experiments are conducted on the standard ATIS corpus and a large-scale dataset which contains about 30-thousand utterances from three different domains. For the small dataset, the proposed methods obtain the state-of-the-art performance compared to published models. For the large dataset, the models are evaluated on semi-supervised learning performance with different amounts of labeled data. The results show that the proposed approaches perform better than previous methods.

We describe the proposed adversarial multi-task learning methods in Section 2 and the training procedure in Section 3. The experimental results and analysis are given in Section 4. Then Section 5 provides the conclusion of this paper.

## 2. ADVERSARIAL MULTI-TASK LEARNING

The slot filling is typically considered as a sequence labeling problem. Given an input sequence with $n$ words $\mathbf{w} = \{w_1, w_2, \cdots, w_n\}$, slot filling is to predict the output (slot-tag) sequence $\mathbf{t} = \{t_1, t_2, \cdots, t_n\}$.

The traditional slot tagging model is only optimised based on the ground truth of labels. Because the number of each word in the input is not greater than one and the size of labeled data is limited, slot tags actually contribute little to a generalized SLU model. Inspired by taking language modeling as a supplementary objective [11], we integrate a unidirectional or bidirectional language model with the slot tagging model. The LM can learn more general patterns of the semantic and syntactic composition without any additional labeled data. The unidirectional LM (ULM) predicts the next word, while the bidirectional LM (BLM) consists of two separate ULMs, predicting the next word and the previous word without share-weighting.

Instead of simply adding an additional parallel output layer, we investigate an adversarial multi-task model using the shared-private framework as illustrated in Fig.1. Each task has its own private space (STM in green, LM in blue) and shares a joint space (in yellow). The first step is to map the current word $w_i$ to a word embedding $\mathbf{e}_i$. A BLSTM is adopted as the hidden layer for the shared, STM-specific and BLM-specific space, and an LSTM for ULM-specific space. Each LSTM takes as input the hidden state from the previous time step and the word embedding at the current step $i$:

$$\overrightarrow{\mathbf{h}_i^k} = LSTM_k(\mathbf{e}_i, \overrightarrow{\mathbf{h}_{i-1}^k}) \quad (1)$$

$$\overleftarrow{\mathbf{h}_i^k} = LSTM_k(\mathbf{e}_i, \overleftarrow{\mathbf{h}_{i+1}^k}) \quad (2)$$

where $k \in \{t, l, s\}$, $t$ refers to slot tagging space, $l$ refers to language modeling space, and $s$ refers to the shared space.

The task-specific output layer estimates the probability of the slot tag or word respectively at time frame $i$:

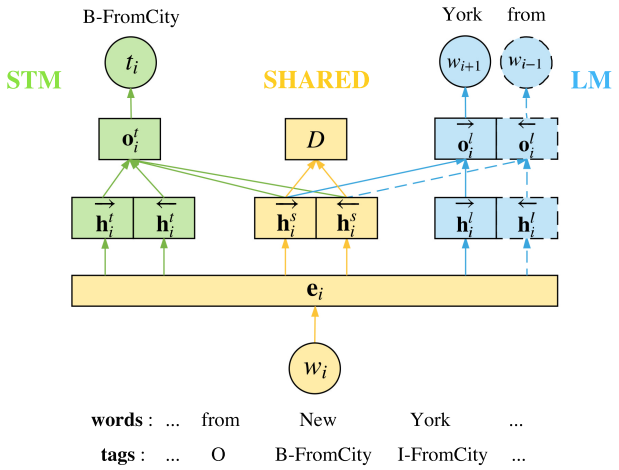


**Fig. 1**. The overview of the adversarial multi-task model. It contains a shared space (in yellow) and two private spaces (slot tagging model in green and language model in blue). The dashed frame is added for BLM. In addition, the task discriminator $D$ is used to force the shared features task-invariant.

$$\mathbf{o}_i^t = \sigma(\mathbf{W}^t[\overrightarrow{\mathbf{h}_i^t}; \overleftarrow{\mathbf{h}_i^t}; \overrightarrow{\mathbf{h}_i^s}; \overleftarrow{\mathbf{h}_i^s}]) \quad (3)$$

$$\overrightarrow{\mathbf{o}_i^l} = \sigma(\overrightarrow{\mathbf{W}^l}[\overrightarrow{\mathbf{h}_i^l}; \overrightarrow{\mathbf{h}_i^s}]); \quad \overleftarrow{\mathbf{o}_i^l} = \sigma(\overleftarrow{\mathbf{W}^l}[\overleftarrow{\mathbf{h}_i^l}; \overleftarrow{\mathbf{h}_i^s}]) \quad (4)$$

where $[\cdot]$ is the concatenation operator, $\mathbf{W}^t, \overrightarrow{\mathbf{W}^l}, \overleftarrow{\mathbf{W}^l}$ are independent weight matrices and $\sigma$ denotes the *softmax* layer which predicts a normalized distribution over all possible tags or words. Then the model can be trained by minimizing the cross-entropy loss between predictive distribution $o_i$ and the ground truth label (slot tag $t_i$, next word $w_{i+1}$ or previous word $w_{i-1}$).

Inspired by the impressive success of adversarial methods in deep generative modeling [14], cross-lingual task [15], and domain adaptation [16], we investigate a task discriminator to make sure the shared space only contains task-invariant features. Specifically, the task discriminator takes as input shared features and predicts whether slot tagging task or language modeling task the inputs are trained for. To confuse the discriminator, the shared space is encouraged to extract task-invariant features. To make a strong rival to the shared model, we analyze the task discriminator on word-level and sentence-level.

The word-level discriminator $D^{(w)}$ calculates the average of the shared feature $\mathbf{h}_i^s = [\overrightarrow{\mathbf{h}_i^s}; \overleftarrow{\mathbf{h}_i^s}]$ at each time-frame $i$ after linear transformation, while the sentence-level discriminator $D^{(s)}$ selects the most salient feature from the sequence of shared features. Then they predict the probability of the task indication $y$, which equals 1 when $\mathbf{w}$ is used for STM training and equals 0 for LM training.

$$P_d(y|\mathbf{w}; \theta^d, \theta^s) = \begin{cases} \frac{1}{n}\sum_{i=1}^{n} \sigma(\mathbf{W}^d \mathbf{h}_i^s) & for\ D^{(w)}; \\ \sigma(\mathbf{W}^d \max_{1 \le i \le n} \mathbf{h}_i^s) & for\ D^{(s)} \end{cases} \quad (5)$$

where $\mathbf{W}^d$ is a weighting matrix in the task discriminator space, $\theta^d$ and $\theta^s$ are the parameters in the discriminator and shared space respectively.

## 3. TRAINING PROCEDURE

We first present the training objective for each component, then show the overall training algorithm. The training objective of $D$ is to maximize the probability of correctly distinguishing the task which input features are used for, while the shared space attempts to confuse the task discriminator:

$$\max_{\theta^d} \mathcal{L}^d = \sum_{(\mathbf{w},y)\in data} \log P_d(y|\mathbf{w};\theta^d,\theta^s) \quad (6)$$

$$\min_{\theta^s} \mathcal{L}^s = \sum_{(\mathbf{w},y)\in data} \log P_d(y|\mathbf{w};\theta^d,\theta^s) \quad (7)$$

where the dataset $data$ could be labeled or unlabeled.

For slot tagging task and language modeling task, the objective functions can be computed as:

$$\max_{\theta^s,\theta^t} \mathcal{L}^t = \sum_{(\mathbf{w},\mathbf{t})\in data^l} \sum_{i=1}^{|\mathbf{w}|} \log P_t(t_i|w_i;\theta^s,\theta^t) \quad (8)$$

$$\max_{\theta^s,\overrightarrow{\theta^l}} \overrightarrow{\mathcal{L}^l} = \sum_{\mathbf{w}\in data} \sum_{i=1}^{|\mathbf{w}|} \log P_l(w_{i+1}|w_i;\theta^s,\overrightarrow{\theta^l}) \quad (9)$$

$$\max_{\theta^s,\overleftarrow{\theta^l}} \overleftarrow{\mathcal{L}^l} = \sum_{\mathbf{w}\in data} \sum_{i=1}^{|\mathbf{w}|} \log P_l(w_{i-1}|w_i;\theta^s,\overleftarrow{\theta^l}) \quad (10)$$

where $data^l$ is the labeled part of $data$, in which each word $w_i$ is annotated with a slot tag $t_i$. $P_t(\cdot|w_i)$ is the probability over slot tags, and $P_l(\cdot|w_i)$ is that over the vocabulary. $w_0$ and $w_{|\mathbf{w}|+1}$ are assigned to the sentence start $< s >$ and the sentence end $< /s >$ respectively.

**Algorithm 1:** Adversarial Multi-task Learning for SLU

**Input :** Labeled training data $\{(\mathbf{w}^l, \mathbf{t}^l)\}$
Unlabeled data $\{\mathbf{w}^u\}$
**Output:** Adversarially enhanced slot tagging model

Initialize parameters $\{\theta^s, \theta^t, \theta^l, \theta^d\}$ randomly.
**repeat**
  /* Sample from $\{(\mathbf{w}^l, \mathbf{t}^l)\}$ */
  Train the STM and shared model by Eq.(8).
  Train the task discriminator and the shared model by Eq.(6) or Eq.(7) as slot tagging task ($y = 1$).
  /* Sample from $\{\mathbf{w}^l\}$ and $\{\mathbf{w}^u\}$ */
  Train the LM and shared models by Eq.(9) (and Eq.(10) for BLM).
  Train the task discriminator and the shared model by Eq.(6) or Eq.(7) as LM task ($y = 0$).
**until** *convergence*;

Algorithm 1 shows the overall adversarial training procedure. The discriminator and shared model conduct a minimax competition through Eq.(6) and Eq.(7) which improve each other until their feature representations are close enough. The shared model is encouraged to extract the generalized features from abundant raw utterances. In addition, Eq.(9) and Eq.(10) learn underlying semantic and syntactic language knowledge. Eq.(8) as a traditional supervised learning objective drives the slot tagging model to perform well on labeled data and transfer the supervised information to unlabeled data.

## 4. EXPERIMENTS

The proposed model and other methods are first evaluated on the Air Travel Information System (ATIS) benchmark. Then we demonstrate the effectiveness of the proposed model on semi-supervised learning with the different numbers of labeled utterances from a large-scale dataset. The experimental results show that our methods substantially improve over traditional semi-supervised methods in the slot filling task.

### 4.1. Experimental Setup

For all architectures, the dimensions of word embeddings and BLSTM hidden units are set to 100. At each time-frame, the SLU model takes the current word as input without any context words. For training, the network parameters are randomly initialized in accordance with the uniform distribution (-0.2, 0.2) and updated by stochastic gradient descent (SGD). The dropout with a probability of 0.5 is applied to the non-recurrent connections for regularization. Different learning rates are tried by grid-search in the range of [0.008, 0.03] and keep it for 100 epochs. The $F1$-scores of slot filling on the test set whose corresponding models perform best on validation are reported.

For adversarial training, the task discriminator and the multi-task model are optimized with the minibatch size of 10. At each iteration, the slot tagging model is trained on labeled data by the supervised algorithm, and the language model is trained on labeled and unlabeled data by self-supervising. Meanwhile, the shared model and the task discriminator are trained by a minimax game.

### 4.2. ATIS Experiment

ATIS includes 4,978 training sentences and 893 test ones from the only air travel domain. Because a slot may be mapped to several continuous words, we follow the popular In/Out/Begin (IOB) representation. The number of different slot tags is 84 (127 if IOB prefixes are considered). We randomly select 80% from the training sentences as the training set while the rest as validation [17]. The following methods are investigated:

**STM:** It is a simple supervised model, using BLSTM as the hidden layer for slot filling task [4].

**STM+LM$_e$:** It pre-trains a language model first, then initializes the word embeddings of an STM by those of the

well-trained LM. The word embeddings are updated during the training process for SLU.

**MTL$_e$:** It exploits the multi-task learning for STM and LM. These two tasks share the embedding layer.

**MTL$_{e+h}$:** Similar to [11], STM and LM share the embedding and hidden layer.

**SPM:** It uses the shared-private model for multi-task learning. Compared to MTL$_e$, it adds a shared hidden space to improve the performance. Compared to MTL$_{e+h}$, it adds private hidden spaces for each task. The output layer inputs both shared and private features. Unidirectional SPM (USPM) contains an STM and a forward LM while bidirectional SPM (BSPM) has an additional backward LM (the dashed blocks in Fig.1).

**SPM+D:** It is the exact model illustrated in Fig.1. Compared to SPM, a task discriminator is added to the framework.

**SPM$^1$+D:** Compared to SPM+D, it eliminates the LM-specific space and remains others unchanged.

| Method | STM | STM+LM$_e$ | MTL$_e$ | MTL$_{e+h}$ |
|---|---|---|---|---|
| $F1^U$ | 95.63 | 95.24 | 94.78 | 94.67 |
| $F1^B$ | | 95.61 | 95.54 | 94.51 |
| **Method** | **SPM** | **SPM+D$^{(w)}$** | **SPM+D$^{(s)}$** | **SPM$^1$+D$^{(w)}$** |
| $F1^U$ | 95.54 | 95.28 | 95.44 | 95.33 |
| $F1^B$ | 95.26 | **95.94** | 95.52 | |

**Table 1**. Experimental results ($F1$-score%) on ATIS dataset. The superscript of $F1$ indicates the LM in the model is unidirectional ($F1^U$) or bidirectional ($F1^B$).

Table 1 shows the performance of these methods on ATIS corpus. Compared with other methods, BSPM+D$^{(w)}$ achieves the state-of-the-art performance of 95.94%. The best published result is 95.86%, proposed by Zhai et al.[18]. Additionally, models equipped with BLM mostly perform better than their counterparts with ULM. It means that considering both sides of context is beneficial to capture the generalized feature for slot tagging. Furthermore, we investigate another update method for the task discriminator. The task indication of shared features is assigned randomly to confuse the discriminator. In this case, the test $F1$-score on BSPM+D$^{(w)}$ declines from 95.94% to 95.28%, which proves the effectiveness of the proposed method described in Alg.1.

### 4.3. Large-scale Experiment

Considering the limited size of ATIS and the necessity to build a slot filling model for multiple domains, we follow the work of Kurata et al.[5, 18], combining the MIT Restaurant Corpus, MIT Movie Corpus [19] and ATIS corpus into a single large-scale data set, denoted as LARGE. This merged dataset contains 30,229 training and 6,810 test sentences from three different domains. The words are assigned to 116 different slot tags (191 with IOB prefix).

For semi-supervised learning, {5k, 10k, 15k} sentences of training data are randomly chosen as labeled and the rest as unlabeled. For each labeled set, we randomly select 80% as training set while the rest as validation. All experiments are evaluated on the same test set. For example, the 5k set has 4,000 labeled training, 1,000 labeled developing, 25,299 unlabeled training, and 6,810 test sentences.

| Method | 5k | 10k | 15k | all |
|---|---|---|---|---|
| STM | 67.25 | 71.04 | 73.94 | 76.60 |
| MTL$_e$ | 69.57 | 73.04 | 75.00 | 77.24 |
| PSEUDO | 69.82 | 72.55 | 74.80 | - |
| BSPM | 68.46 | 72.52 | **75.05** | **77.52** |
| BSPM+D$^{(w)}$ | **71.55** | **73.67** | 74.61 | 77.42 |
| BSPM+D$^{(s)}$ | 70.99 | 73.58 | 74.22 | 77.24 |

**Table 2**. Experimental results ($F1$-score%) on LARGE dataset. {5k, 10k, 15k, all} sets select 5,000, 10,000, 15,000 and 30,229 sentences from the training set as labeled.

The results are illustrated in Table 2. Only bidirectional methods are shown, which have been proved to be more effective on ATIS. As a successful method, PSEUDO performs a pipeline in three stages: training an STM with the labeled data, generating pseudo labels for the unlabeled data by the pre-trained model, and retain an STM with labeled and pseudo-labeled data simultaneously [9, 10].

From Table 2, we can see the proposed BSPM and BSPM+D constantly achieve better performance than other methods for different labeled sets. Our method improves significantly (99.9%) over STM on all datasets. Compared with MTL$_e$, our method has a significant level of 99.9% on 5k set, and of 99.5% on 10k set. However, the improvement is not significant on 15k set. Similarly, our method improves significantly (99.8%) over PSEUDO on 5k and 10k set, but not significantly (over 95%) in 15k set.

The experiments indicate that our semi-supervised learning model is more efficient when the labeled data is limited and the data for LM is more sufficient. While BLM exploits the unsupervised knowledge, the shared-private framework and adversarial training make the slot tagging model more generalized and perform better on unseen samples.

## 5. CONCLUSION

In this paper, we propose an adversarial multi-task learning method for semi-supervised training on SLU, which alleviates the dependence on labeled data. A bidirectional language model is intersected with the slot tagging model by sharing the joint space and monopolizing a private LM space. Therefore, the slot tagging model acquires generalized language knowledge from the shared space and obtains supervised information from its private STM space. In addition, a task discriminator is used to force the shared space to discard task-specific information. The proposed methods achieve the state-of-the-art performance on ATIS benchmark, and significantly outperform previous models with limited labeled data on the large-scale dataset.

## 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 *IEEE International Conference on Acoustics, Speech, and Signal Processing, 1992.*, 1992, vol. 1, pp. 193–196.

[2] Gokhan Tur, Dilek Hakkani-Tr, and Robert E. Schapire, "Combining active and semi-supervised learning for spoken language understanding," *Speech Communication*, vol. 45, no. 2, pp. 171 – 186, 2005.

[3] Grégoire Mesnil, Xiaodong He, Li Deng, and Yoshua Bengio, "Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding," in *14th Annual Conference of the International Speech Communication Association*, 2013, pp. 3771–3775.

[4] 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 Workshop on Spoken Language Technology*. IEEE, 2014, pp. 189–194.

[5] 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, 2016, pp. 2077–2083.

[6] 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*, 2016.

[7] Su Zhu and Kai Yu, "Encoder-decoder with focus-mechanism for sequence labelling based spoken language understanding," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 5675–5679.

[8] Gokhan Tur, Dilek Hakkani-Tür, and Larry Heck, "What is left to be understood in atis?," in *2010 IEEE Workshop on Spoken Language Technology*, 2010, pp. 19–24.

[9] Asli Celikyilmaz, Ruhi Sarikaya, Dilek Hakkani-Tur, Xiaohu Liu, Nikhil Ramesh, and Gokhan Tur, "A new pre-training method for training deep learning models with application to spoken language understanding," in *17th Annual Conference of the International Speech Communication Association*, 2016, pp. 3255–3259.

[10] Dong Hyun Lee, "Pseudo-label : The simple and efficient semi-supervised learning method for deep neural networks," in *Workshop on Challenges in Representation Learning, ICML*, 2013.

[11] Marek Rei, "Semi-supervised multitask learning for sequence labeling," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, July 2017, pp. 2121–2130.

[12] Matthew Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power, "Semi-supervised sequence tagging with bidirectional language models," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, July 2017, pp. 1756–1765.

[13] Xinchi Chen, Zhan Shi, Xipeng Qiu, and Xuanjing Huang, "Adversarial multi-criteria learning for chinese word segmentation," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, Canada, July 2017, pp. 1193–1203.

[14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems 27*, 2014, pp. 2672–2680.

[15] Xilun Chen, Yu Sun, Ben Athiwaratkun, Claire Cardie, and Kilian Weinberger, "Adversarial deep averaging networks for cross-lingual sentiment classification," *arXiv preprint arXiv:1606.01614*, 2016.

[16] 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.

[17] 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," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 23, no. 3, pp. 530–539, 2015.

[18] Feifei Zhai, Saloni Potdar, Bing Xiang, and Bowen Zhou, "Neural models for sequence chunking.," in *AAAI*, 2017, pp. 3365–3371.

[19] J. Liu, P. Pasupat, S. Cyphers, and J. Glass, "Asgard: A portable architecture for multilingual dialogue systems," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 8386–8390.