INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Unsupervised Domain Adaptation for Dialogue Sequence Labeling Based on Hierarchical Adversarial Training

Shota Orihashi, Mana Ihori, Tomohiro Tanaka, Ryo Masumura

NTT Media Intelligence Laboratories, NTT Corporation, Japan

shota.orihashi.bt@hco.ntt.co.jp

Abstract

This paper presents a novel unsupervised domain adaptation method for dialogue sequence labeling. Dialogue sequence labeling is a supervised learning task that estimates labels for each utterance in the given dialogue document, and is useful for many applications such as topic segmentation and dialogue act estimation. Accurate labeling often requires a large amount of labeled training data, but it is difficult to collect such data every time we need to support a new domain, such as contact centers in a new business field. In order to solve this difficulty, we propose an unsupervised domain adaptation method for dialogue sequence labeling. Our key idea is to construct dialogue sequence labeling using labeled source domain data and unlabeled target domain data so as to remove domain dependencies at utterance-level and dialogue-level contexts. The proposed method adopts hierarchical adversarial training; two domain adversarial networks, an utterance-level context independent network and a dialogue-level context dependent network, are introduced for improving domain invariance in the dialogue sequence labeling. Experiments on Japanese simulated contact center dialogue datasets demonstrate the effectiveness of the proposed method.

Index Terms: dialogue sequence labeling, unsupervised domain adaptation, hierarchical adversarial training

1. Introduction

With the progress of automatic speech recognition, expectations for the understanding and utilization of linguistic information of human-to-human conversation are increasing. For example, by understanding a telephone conversation document in a contact center, a service for discovering customer needs and issues with the center was developed [1-6].

In this paper, we focus on utterance-level dialogue sequence labeling, a key component that is important for document understanding. Dialogue sequence labeling is often modeled as a supervised learning task that estimates labels for each utterance when given a dialogue document, and is useful for many applications such as topic segmentation, dialogue act estimation, and call scene segmentation [7-14]. In particular, to understand conversation documents in a contact center, it is necessary to consider who spoke what and in what order. It has been reported that the method of modeling a long-term context across utterance boundaries is effective in achieving both [12].

In order to realize highly accurate dialogue sequence labeling, a large labeled training dataset is required. Accordingly, we have to set labels to dialogue transcripts for comprehensive dialogue understanding every time a new domain emerges to challenge the contact center, such as a new business field. However, it is difficult to collect a large amount of labeled target domain data, and this difficulty is a major problem for expanding the solutions for contact centers.

To overcome the difficulty in collecting labeled target domain data, we focus on unsupervised domain adaptation that uses both source domain datasets with annotated labels and target domain datasets without labels to train a model for the target domain. In recent years, many successful cases have been reported in the image processing field, and most of them have been realized by adversarial training using a domain classification network that identifies the domain of the input image [15-20]. Unsupervised domain adaptation is considered to be a suitable approach to reduce labeling costs because the model for the target domain can be constructed without labeled target domain data. However, no truly effective unsupervised domain adaptation techniques for dialogue sequence labeling have been described so far.

In this paper, we propose a novel unsupervised domain adaptation method for dialogue sequence labeling. Our method, similar to the approach used for image processing, consists of adversarial training with domain classification networks; domain classification networks are used for improving the domain invariance of the main network. Unlike the method in the field of image processing, which uses one domain classification network, our method, hierarchical adversarial training, hierarchically structures two domain classification networks which individually identify domain labels from long-term and short-term contexts. By achieving domain-invariant dialogue sequence labeling via hierarchical adversarial training, our method is expected to perform labeling in the target domain with high accuracy. To the best of our knowledge, this is the first method to achieve unsupervised domain adaptation for fully neural networks based dialogue sequence labeling through adversarial training. Our experiments on Japanese simulated contact center dialogue datasets demonstrate the effectiveness of the proposed method.

This paper is organized as follows. Section 2 describes work related to unsupervised domain adaptation for dialogue sequence labeling. We introduce dialogue sequence labeling based on a fully neural network in Section 3. Section 4 details our proposed method. In Section 5, we describe the evaluation conducted and the results gained. Section 6 concludes this paper.

2. Related work

2.1. Utterance-level dialogue sequence labeling

Utterance-level sequence labeling is used for topic segmentation and dialogue act estimation [7-14]. Hierarchical recurrent neural networks based on token units and utterance units are often used to efficiently capture short-term contexts between tokens and long-term contexts between utterances. In this paper, we focus on the utterance-level sequence labeling of hierarchical recurrent neural networks specialized for conversation documents [12]. In order to eliminate the need to collect a large

http://dx.doi.org/10.21437/Interspeech.2020-2010

amount of labeled training data, we introduce an unsupervised domain adaptation technique into utterance-level dialogue sequence labeling.

2.2. Unsupervised domain adaptation

Unsupervised domain adaptation is the technique to convert a machine learning model from a source domain into the target domain equivalent by using unlabeled data of the target domain. Recently, unsupervised domain adaptation methods for classification models based on fully neural networks have been proposed in the image processing field [15-20]. The typical method matches the distribution of the intermediate representation of the target domain with that of the source domain. In particular, algorithms based on adversarial learning with domain classification network, i.e. domain adversarial network, that identify domains of input have been applied to many tasks [15]. This paper also employs domain adversarial network-based unsupervised domain adaptation. Our method uses two domain adversarial networks to efficiently remove the dependencies of short-term contexts between tokens and the long-term contexts between utterances.

3. Utterance-level dialogue sequence labeling

This section describes utterance-level dialogue sequence labeling in conversation documents using neural networks. This task estimates utterance-level label sequence $\boldsymbol{Y} = \{y^1, \cdots, y^T\}$ from input utterance sequence $\boldsymbol{X} = \{x^1, \cdots, x^T\}$ using neural networks, where the t-th label $y^t \in \mathcal{Y}$ and $\mathcal{Y}$ is the set of labels. Label types are task dependent, for example call scene labels for call scene segmentation in the contact center.

In our dialogue sequence labeling, the t-th label y^t is estimated from $\{x^1, \cdots, x^t\}$. For this, conditional probability $P(y^t \mid x^1, \cdots, x^t, \boldsymbol{\Theta})$ is modeled where $\boldsymbol{\Theta}$ represents a model parameter. The t-th label can be categorized by:

$$\hat{y}^t = \arg\max_{y^t \in \mathcal{Y}} P(y^t \mid x^1, \cdots, x^t, \boldsymbol{\Theta}). \tag{1}$$

In this paper, we assume that $P(y^t \mid x^1, \cdots, x^t, \boldsymbol{\Theta})$ is modeled by hierarchical long short-term memory recurrent neural networks (LSTM-RNNs) which are based on fully neural networks [12]. Figure 1 shows the structure of the labeling network.

The t-th utterance x^t consists of token sequence $\{w_1^t, \cdots, w_{K^t}^t\}$, where K^t is number of tokens in the t-th utterance. Each token is first converted into a continuous vector representation. The continuous vector representation of the k-th token in the t-th utterance is given by:

$$\boldsymbol{w}_k^t = \mathrm{EMBED}(w_k^t; \boldsymbol{\theta}^{\boldsymbol{w}}), \tag{2}$$

where $\mathrm{EMBED}()$ is linear transformational function that embeds a symbol into a continuous vector and $\boldsymbol{\theta}^{\boldsymbol{w}}$ is the trainable parameter. In the token-level LSTM-RNN, each token vector representation is first converted into a hidden representation that takes neighboring context into consideration. The hidden representation for the k-th token in the t-th utterance is calculated as:

$$\boldsymbol{h}_k^t = \mathrm{BLSTM}(\boldsymbol{w}_1^t, \cdots, \boldsymbol{w}_{K^t}^t; \boldsymbol{\theta}^{\boldsymbol{h}}), \tag{3}$$

where $\mathrm{BLSTM}()$ is a function of the bidirectional LSTM-RNN layer, and $\boldsymbol{\theta}^{\boldsymbol{h}}$ is the trainable parameter. The hidden representations are then summarized as an utterance representation by a self-attention mechanism. The t-th utterance continuous representation is calculated as:

$$\boldsymbol{s}^t = \mathrm{SelfAttention}(\boldsymbol{h}_1^t, \cdots, \boldsymbol{h}_{K^t}^t; \boldsymbol{\theta}^{\boldsymbol{s}}), \tag{4}$$

where $\mathrm{SelfAttention}()$ is a transformational function that converts to a fixed-length vector by weighting and adding individual hidden representations that consider the importance of each element [21]; $\boldsymbol{\theta}^{\boldsymbol{s}}$ is the trainable parameter.

In the utterance-level LSTM-RNN, interaction information from start-of-dialogue to the t-th utterance is incrementally embedded into a continuous vector representation. The t-th continuous vector representation that embeds all dialogue context sequential information up to the t-th utterance is given as:

$$\boldsymbol{u}^t = \mathrm{LSTM}(\boldsymbol{s}^1, \cdots, \boldsymbol{s}^t; \boldsymbol{\theta}^{\boldsymbol{u}}), \tag{5}$$

where $\mathrm{LSTM}()$ is a function of the unidirectional LSTM-RNN layer, and $\boldsymbol{\theta}^{\boldsymbol{u}}$ represents the trainable parameter.

In the output layer, predictive probabilities of the labels for the t-th utterance are defined as:

$$\boldsymbol{o}^t = \mathrm{SOFTMAX}(\boldsymbol{u}^t; \boldsymbol{\theta}^{\boldsymbol{o}}), \tag{6}$$

where $\mathrm{SOFTMAX}()$ is a softmax function, and $\boldsymbol{\theta}^{\boldsymbol{o}}$ is a model parameter for the softmax function. $\boldsymbol{o}^t$ corresponds to $P(y^t \mid x^1, \cdots, x^t, \boldsymbol{\Theta})$.

The model parameters $\boldsymbol{\Theta} = \{\boldsymbol{\theta}^{\boldsymbol{w}}, \boldsymbol{\theta}^{\boldsymbol{h}}, \boldsymbol{\theta}^{\boldsymbol{s}}, \boldsymbol{\theta}^{\boldsymbol{u}}, \boldsymbol{\theta}^{\boldsymbol{o}}\}$ can be optimized by preparing training dataset $\mathcal{D} = \{(\boldsymbol{X}_1, \overline{\boldsymbol{Y}}_1), \cdots, (\boldsymbol{X}_N, \overline{\boldsymbol{Y}}_N)\}$, where $\boldsymbol{X}_n$ and $\overline{\boldsymbol{Y}}_n$ are input utterance sequence and reference utterance-level label sequence, respectively. In this case, cross-entropy loss is computed by:

$$\mathcal{L} = -\frac{1}{N} \sum_{n=1}^{N} \sum_{t=1}^{T_n} \sum_{y \in \mathcal{Y}} \overline{o}_{n,y}^t \log o_{n,y}^t, \tag{7}$$

where $\overline{\boldsymbol{o}}_n^t = [\overline{o}_{n,1}^t, \cdots, \overline{o}_{n,|\mathcal{Y}|}^t]$ and $\boldsymbol{o}_n^t = [o_{n,1}^t, \cdots, o_{n,|\mathcal{Y}|}^t]$ are the reference and estimated probabilities of label y for the t-th end-of-utterance in the n-th conversation, respectively; T_n is the number of utterances in the n-th conversation. The optimization can be conducted by mini-batch stochastic gradient decent (SGD). The model parameters are updated by:

$$\boldsymbol{\Theta} \longleftarrow \boldsymbol{\Theta} - \mu \frac{\partial \mathcal{L}^b}{\partial \boldsymbol{\Theta}}, \tag{8}$$

where μ is the learning rate and $\mathcal{L}^b$ is the cross-entropy loss for the b-th mini-batch.

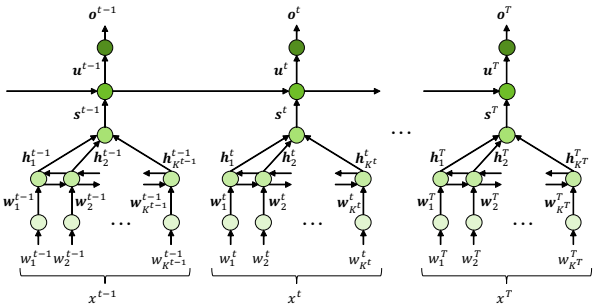

Figure 1: *Structure of dialogue sequence labeling network.*

4. Proposed method

This section details the proposed unsupervised domain adaptation method for utterance-level dialogue sequence labeling. To optimize a model in the target domain, this training uses both source domain datasets with annotated labels and target domain datasets without labels. To this end, we compose hierarchical domain adversarial networks with a gradient reversal layer (GRL). GRL accepts input vectors during forward propagation, and sign inversion of the gradients during back propagation [15].

The main idea of the proposed method, hierarchical adversarial training, is to use hierarchically-structured two domain adversarial networks, which individually identify a domain label from long-term and short-term contexts. The first domain adversarial network, named context-independent domain adversarial network (CIDAN), classifies the domain of the input document by capturing utterance-level hidden representation of the labeling network. In CIDAN, feedback by backpropagation is processed on each utterance in order to efficiently cut the domain dependency of specific utterances that may have domain dependency, such as utterances that include domain specific tokens. The second domain adversarial network, named context-dependent domain adversarial network (CDDAN), classifies the domain of the input document by capturing dialogue-level hidden representation of the labeling network. In CDDAN, feedback by backpropagation is processed at the utterance-level in order to remove all domain dependency present in the utterance flow. By using two domain adversarial networks, our method efficiently achieves domain-invariant dialogue sequence labeling.

Figure 2 shows structure of the proposal. The CIDAN estimates input domain $d^t_{\mathrm{CI}} \in \{d_s, d_t\}$ from utterance-level hidden representation $\boldsymbol{s}^t$ using neural networks, where d_s and d_t represent labels for source and target domain, respectively. In CIDAN, the hidden representation of each utterance is embedded into one vector. The vector representation that embeds the t-th utterance is given as:

$$\boldsymbol{v}^t = \mathrm{DENSE}(\boldsymbol{s}^t; \boldsymbol{\theta}^{\boldsymbol{v}}), \tag{9}$$

where $\mathrm{DENSE}()$ is a function of fully-connected layer, and $\boldsymbol{\theta}^{\boldsymbol{v}}$ represents the trainable parameter. In the output layer, predictive probabilities of the domain label are defined as:

$$\boldsymbol{q}^t = \mathrm{SOFTMAX}(\boldsymbol{v}^t; \boldsymbol{\theta}^{\boldsymbol{q}}), \tag{10}$$

where $\boldsymbol{\theta}^{\boldsymbol{q}}$ is a trainable parameter for the softmax function.

CDDAN estimates input domain $\{d^1_{\mathrm{CD}}, \cdots, d^T_{\mathrm{CD}}\}$ sequentially from the dialogue-level hidden representation $\{\boldsymbol{u}^1, \cdots, \boldsymbol{u}^T\}$ using neural networks, where $d^t_{\mathrm{CD}} \in \{d_s, d_t\}$. In CDDAN, dialogue-level hidden representation is embedded into a continuous vector representation. The t-th continuous vector representation that embeds all hidden representation behind the t-th utterance is given as:

$$\boldsymbol{z}^t = \mathrm{LSTM}(\boldsymbol{u}^1, \cdots, \boldsymbol{u}^t; \boldsymbol{\theta}^{\boldsymbol{z}}), \tag{11}$$

where $\boldsymbol{\theta}^{\boldsymbol{z}}$ represents the trainable parameter. In the output layer, predictive probabilities of the domain label are defined as:

$$\boldsymbol{r}^t = \mathrm{SOFTMAX}(\boldsymbol{z}^t; \boldsymbol{\theta}^{\boldsymbol{r}}), \tag{12}$$

where $\boldsymbol{\theta}^{\boldsymbol{r}}$ is a trainable parameter for the softmax function.

The model parameters for CIDAN $\boldsymbol{\Theta}_{\mathrm{CI}} = \{\boldsymbol{\theta}^{\boldsymbol{v}}, \boldsymbol{\theta}^{\boldsymbol{q}}\}$, CDDAN $\boldsymbol{\Theta}_{\mathrm{CD}} = \{\boldsymbol{\theta}^{\boldsymbol{z}}, \boldsymbol{\theta}^{\boldsymbol{r}}\}$, and labeling network $\boldsymbol{\Theta}_{\mathrm{S}} = \{\boldsymbol{\theta}^{\boldsymbol{w}}, \boldsymbol{\theta}^{\boldsymbol{h}}, \boldsymbol{\theta}^{\boldsymbol{s}}\}$, $\boldsymbol{\Theta}_{\mathrm{U}} = \{\boldsymbol{\theta}^{\boldsymbol{u}}\}$ and $\boldsymbol{\Theta}_{\mathrm{O}} = \{\boldsymbol{\theta}^{\boldsymbol{o}}\}$, can be optimized by preparing both source domain training dataset $\mathcal{D}_S = \{(\boldsymbol{X}_1, \overline{\boldsymbol{Y}}_1), \cdots, (\boldsymbol{X}_N, \overline{\boldsymbol{Y}}_N)\}$ and target domain training dataset $\mathcal{D}_T = \{\boldsymbol{X}_{N+1}, \cdots, \boldsymbol{X}_{N+M}\}$. In this case, the cross-entropy loss for CIDAN, CDDAN and labeling network is computed from:

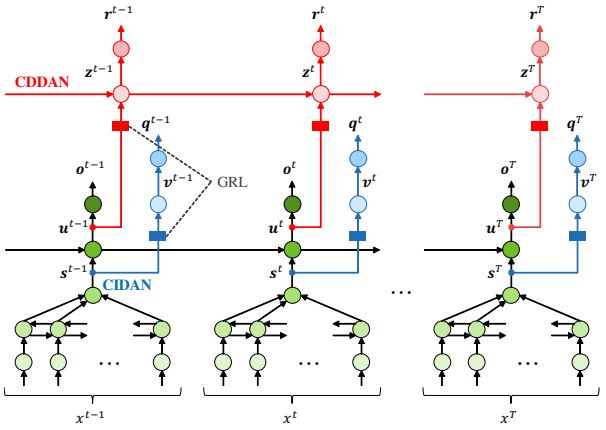

Figure 2: *Structure of proposed domain adversarial networks.*

Table 1: *Details of the dialogue dataset.*

Business type	#calls	#utterances	#tokens
Finance	59	6,081	55,933
Internet provider	57	3,815	47,668
Government unit	73	5,617	48,998
Mail-order	56	4,938	46,574
PC repair	55	6,263	55,101
Mobile phone	61	5,738	51,061

$$\mathcal{L}_{\mathrm{CI}} = -\frac{1}{N+M}\sum_{n=1}^{N+M}\sum_{t=1}^{T_n}\sum_{d\in\{d_s,d_t\}} \overline{q}^t_{n,d}\log q^t_{n,d}, \tag{13}$$

$$\mathcal{L}_{\mathrm{CD}} = -\frac{1}{N+M}\sum_{n=1}^{N+M}\sum_{t=1}^{T_n}\sum_{d\in\{d_s,d_t\}} \overline{r}^t_{n,d}\log r^t_{n,d}, \tag{14}$$

$$\mathcal{L}_{\mathrm{Y}} = -\frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T_n}\sum_{y\in\mathcal{Y}} \overline{o}^t_{n,y}\log o^t_{n,y}, \tag{15}$$

where $\overline{\boldsymbol{q}}^t_n = [\overline{q}^t_{n,d_s}, \overline{q}^t_{n,d_t}]$ and $\overline{\boldsymbol{r}}^t_n = [\overline{r}^t_{n,d_s}, \overline{r}^t_{n,d_t}]$ are the reference probability of domain label d for the t-th utterance in the n-th conversation, and $\boldsymbol{q}^t_n = [q^t_{n,d_s}, q^t_{n,d_t}]$ and $\boldsymbol{r}^t_n = [r^t_{n,d_s}, r^t_{n,d_t}]$ are their estimated probabilities. The optimization can be conducted by mini-batch SGD. Due to use of GRL, the model parameters are updated by:

$$\boldsymbol{\Theta}_{\mathrm{CI}} \longleftarrow \boldsymbol{\Theta}_{\mathrm{CI}} - \lambda\frac{\partial\mathcal{L}^b_{\mathrm{CI}}}{\partial\boldsymbol{\Theta}_{\mathrm{CI}}}, \tag{16}$$

$$\boldsymbol{\Theta}_{\mathrm{CD}} \longleftarrow \boldsymbol{\Theta}_{\mathrm{CD}} - \sigma\frac{\partial\mathcal{L}^b_{\mathrm{CD}}}{\partial\boldsymbol{\Theta}_{\mathrm{CD}}}, \tag{17}$$

$$\boldsymbol{\Theta}_{\mathrm{S}} \longleftarrow \boldsymbol{\Theta}_{\mathrm{S}} - \mu\left\{\frac{\partial\mathcal{L}^b_{\mathrm{Y}}}{\partial\boldsymbol{\Theta}_{\mathrm{S}}} - \lambda\frac{\partial\mathcal{L}^b_{\mathrm{CI}}}{\partial\boldsymbol{\Theta}_{\mathrm{S}}} - \sigma\frac{\partial\mathcal{L}^b_{\mathrm{CD}}}{\partial\boldsymbol{\Theta}_{\mathrm{S}}}\right\}, \tag{18}$$

$$\boldsymbol{\Theta}_{\mathrm{U}} \longleftarrow \boldsymbol{\Theta}_{\mathrm{U}} - \mu\left\{\frac{\partial\mathcal{L}^b_{\mathrm{Y}}}{\partial\boldsymbol{\Theta}_{\mathrm{U}}} - \sigma\frac{\partial\mathcal{L}^b_{\mathrm{CD}}}{\partial\boldsymbol{\Theta}_{\mathrm{U}}}\right\}, \tag{19}$$

Table 2: *Experimental result for BLSTM-LSTM model in terms of classification accuracy (%).*

Target domain	Finance	Internet provider	Government unit	Mail-order	PC repair	Mobile phone
Train on target	84.95	83.95	88.82	75.35	87.19	83.56
Source only	77.08	73.60	76.73	67.50	77.56	74.33
CIDAN only	77.93	74.92	77.99	72.76	83.30	81.49
CDDAN only	76.49	74.06	77.07	70.86	76.34	76.41
CIDAN+CDDAN	**78.87**	**76.66**	**80.05**	**74.63**	**83.60**	**81.94**

Table 3: *Experimental result for LSTM-LSTM model in terms of classification accuracy (%).*

Target domain	Finance	Internet provider	Government unit	Mail-order	PC repair	Mobile phone
Train on target	82.62	83.10	88.43	75.85	86.62	85.59
Source only	75.33	72.06	77.41	68.83	78.69	73.26
CIDAN only	76.44	73.19	77.47	71.13	81.58	79.92
CDDAN only	75.42	72.37	75.03	70.27	76.79	76.39
CIDAN+CDDAN	**77.25**	**75.56**	**79.08**	**73.43**	**82.48**	**81.87**

$$\boldsymbol{\Theta}_{\mathrm{O}} \longleftarrow \boldsymbol{\Theta}_{\mathrm{O}} - \mu \frac{\partial \mathcal{L}_{\mathrm{Y}}^{b}}{\partial \boldsymbol{\Theta}_{\mathrm{O}}}, \tag{20}$$

where λ, σ and μ define the learning rate, and $\mathcal{L}_{\mathrm{CI}}^{b}$, $\mathcal{L}_{\mathrm{CD}}^{b}$ and $\mathcal{L}_{\mathrm{Y}}^{b}$ are the cross-entropy loss for the b-th mini-batch.

5. Experiment

5.1. Datasets

We evaluated topic segmentation tasks using a simulated Japanese contact center dialogue dataset consisting of 361 dialogues in six business fields. One dialogue means one telephone call between one operator and one customer; all utterances were manually transcribed. Each dialogue was divided into speech units using LSTM-RNN based speech activity detection [22] trained from various Japanese speech samples. As topic labels, we set five labels corresponding to the scenes of opening, requirement confirmation, response, customer confirmation, and closing scenes [12].

The evaluation involved six-fold cross validation open to business type, in which five business types used for source domain and the remaining business type used for target domain. For each business type, we split the dataset into training dataset and test dataset at a rate of $8:2$. Training used training data of source domain and target domain, and testing used test data of the target domain. Table 1 shows the details of the dialogue dataset with labels.

5.2. Conditions

In our experiments, we evaluated the proposed method by the two kinds of dialogue sequence labeling models below.

- BLSTM-LSTM model as described in Section 3.
- LSTM-LSTM model in which token-level LSTM-RNN is changed by replacing Eq. (3) with $\boldsymbol{h}_k^t = \mathrm{LSTM}(\boldsymbol{w}_1^t, \cdots, \boldsymbol{w}_k^t; \boldsymbol{\theta}^{\boldsymbol{h}})$ to emphasize low-computation overhead.

We unified the network configurations as follows. For the labeling network, we defined the token vector representation as a 256-dimensional vector. Tokens that appeared only once in the training datasets were treated as unknown tokens. Unit size of LSTM-RNN and fully-connected layer were set to 256. Dropout was used for $\mathrm{BLSTM}()$, $\mathrm{LSTM}()$ and $\mathrm{DENSE}()$, and the dropout rate was set to 0.2.

For training, the mini-batch size was set to five calls. The optimizer was Adam [23] with the default setting. For unsupervised domain adaptation, we incremented (identically) hyperparameters λ and σ from 0.0 to 0.5 in steps of 0.1. Note that a part of the training sets was used as the datasets employed for early stopping. We constructed five models by varying the initial parameters and evaluated them from their average accuracy.

5.3. Results

The resulting classification accuracy values for BLSTM-LSTM model and LSTM-LSTM model are shown in Tables 2 and 3, respectively. In the tables, target domain means business type of target domain (other five types are source domain). Line 1 of each table shows ideal results achieved by using labeled target domain datasets. Line 2 shows results yielded by utilizing labeled source domain datasets. The results show that there is a performance gap between line 1 and line 2. This indicates that domain bias is clearly present in different business fields. Lines 3-5 show the results of unsupervised domain adaptation methods. Lines 3 and 4 show results achieved by using only CIDAN and CDDAN, respectively, and line 5 show the results of applying both domain adversarial networks. The results show that proposed method outperformed the labeled source domain datasets used in isolation. In addition, the use of hierarchical adversarial training yields performance that exceeds the use of just one domain adversarial network. This shows that the proposed method is an effective way of improving performance in the target domain.

6. Conclusion

This paper has proposed a novel unsupervised domain adaptation method, hierarchical adversarial training, for dialogue sequence labeling. The key advance of our method is to remove domain dependencies from main networks using hierarchically-structured two domain adversarial networks. This efficiently achieves domain adaptation even though the dialogue dataset of the target domain is unlabeled. Experiments showed that our method yields better performance in the target domain.

7. References

[1] J. Mamou, D. Carmel, and R. Hoory, "Spoken document retrieval from call-center conversations," *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 51-58, 2006.

[2] R. J. Byrd, M. S. Neff, W. Teiken, Y. Park, K. S. F. Cheng, S. C. Gates, and K. Visweswariah, "Semi-automated logging of contact center telephone calls," *Proceedings of the 17th ACM Conference on Information and Knowledge Management (CIKM)*, pp. 133-142, 2008.

[3] R. Higashinaka, Y. Minami, H. Nishikawa, K. Dohsaka, T. Meguro, S. Takahashi, and G. Kikui, "Learning to model domain-specific utterance sequences for extractive summarization of contact center dialogues," *Proceedings of the 23rd International Conference on Computational Linguistics (COLING)*, pp. 400-408, 2010.

[4] A. Tamura, K. Ishikawa, M. Saikou, and M. Tsuchida, "Extractive summarization method for contact center dialogues based on call logs," *Proceedings of the 5th International Joint Conference on Natural Language Processing (IJCNLP)*, pp. 500-508, 2011.

[5] C. Chastagnol and L. Devillers, "Analysis of anger across several agent-customer interactions in French call centers," *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4960-4963, 2011.

[6] A. Ando, R. Masumura, H. Kamiyama, S. Kobashikawa, and Y. Aono, "Hierarchical LSTMs with joint learning for estimating customer satisfaction from contact center calls," *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 1716-1720, 2017.

[7] J. Yu, X. Xiao, L. Xie, E. S. Chng, and H. Li, "A DNN-HMM approach to story segmentation," *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 1527-1531, 2016.

[8] E. Tsunoo, P. Bell, and S. Renals, "Hierarchical recurrent neural network for story segmentation," *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 2919-2923, 2017.

[9] E. Tsunoo, O. Klejch, P. Bell, and S. Renals, "Hierarchical recurrent neural network for story segmentation using fusion of lexical and acoustic features," *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 525-532, 2017.

[10] Q. H. Tran, I. Zukerman, and G. Haffari, "A hierarchical neural model for learning sequences of dialogue acts," *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, vol. 1, pp. 428-437, 2017.

[11] H. Kumar, A. Agarwal, R. Dasgupta, and S. Joshi, "Dialogue act sequence labeling using hierarchical encoder with CRF," *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, pp. 3440-3447, 2018.

[12] R. Masumura, S. Yamada, T. Tanaka, A. Ando, H. Kamiyama, and Y. Aono, "Online call scene segmentation of contact center dialogues based on role aware hierarchical LSTM-RNNs," *Proceedings of the Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pp. 811-815, 2018.

[13] W. Jiao, H. Yang, I. King, and M. R. Lyu, "HiGRU: Hierarchical gated recurrent units for utterance-level emotion recognition," *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 397-406, 2019.

[14] Y. Yu, S. Peng, and G. H. Yang, "Modeling long-range context for concurrent dialogue acts recognition," *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*, pp. 2277-2280, 2019.

[15] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," *Proceedings of the 32nd International Conference on Machine Learning (ICML)*, vol. 37, pp. 1180-1189, 2015.

[16] K. Bousmalis, G. Trigeorgis, N. Silberman, D. Krishnan, and D. Erhan, "Domain separation networks," *Proceedings of the 30th Conference on Neural Information Processing Systems (NIPS)*, pp. 343-351, 2016.

[17] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7167-7176, 2017.

[18] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3723-3732, 2018.

[19] G. Kang, L. Jiang, Y. Yang, and A. G. Hauptmann, "Contrastive adaptation network for unsupervised domain adaptation," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4893-4902, 2019.

[20] W. G. Chang, T. You, S. Seo S. Kwak, and B. Han, "Domain-specific batch normalization for unsupervised domain adaptation," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7354-7362, 2019.

[21] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pp. 1480-1489, 2016.

[22] F. Eyben, F. Weninger, S. Squartini, and B. Schuller, "Real-life voice activity detection with LSTM recurrent neural networks and an application to Hollywood movies," *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 483-487, 2013.

[23] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2015.