The impact of domain-specific representations on BERT-based multi-domain spoken language understanding

Judith Gaspers, Quynh Do, Tobias Röding, Melanie Bradford
Amazon Alexa AI
{gaspers, doquynh, rodingtr, neunerm}@amazon.com

Abstract

This paper provides the first experimental study on the impact of using domain-specific representations on a BERT-based multi-task spoken language understanding (SLU) model for multi-domain applications. Our results on a real-world dataset covering three languages indicate that by using domain-specific representations learned adversarially, model performance can be improved across all of the three SLU subtasks domain classification, intent classification and slot filling. Gains are particularly large for domains with limited training data.

1 Introduction

Spoken Language Understanding (SLU) is a key task in voice-controlled devices, such as Amazon Alexa or Google Home. It is often divided into the two subtasks intent classification (IC) determining the user intent and slot filling (SF) extracting semantic constituents. For instance, given an utterance "play madonna", IC should determine *PlayMusic* as the intent, while SF should label "play" and "madonna" as *Other* and *Artist* slots, respectively. The third subtask, domain classification (DC) which classifies user utterances into different domains, is sometimes required, especially in large-scale industry applications. DC helps when the data comes from different domains and the SLU labeling results require some domain-specific post-processing steps.

Aiming at reducing model building efforts, in this paper, we focus on single multi-task models solving the three subtasks of DC, IC and SF jointly for multi-domain data. Traditionally, these models learn shared-representations for all domains (e.g. Kapoor and Tirkaz (2019); Hakkani-Tür et al. (2016); Kim et al. (2017)). However, using only domain-shared representations may limit model performance as it ignores potentially useful domain-specific knowledge. For instance, let us consider two user requests "play madonna" and "play star wars", belonging to the *Music* and *Video* domains, respectively. In this case, the carrier phrase "play" is shared while the slot values "madonna" and "star wars" are specific to *Music* and *Video*, respectively. Thus, both domain-shared and domain-specific representations could potentially be useful.

In order to shed a light on whether (missing) domain-specific knowledge may play an impact on the model performance, we carry out a wide range of experiments on real-world data comparing the performances of DC-IC-SF multi-task models with and without domain-specific representations. To assure that the models used in the experiments are close to the current state-of-the-art systems, we adapt one of the most recent BERT-based language-adversarial approaches for IC-SF (Do et al., 2020) to our problem. In particular, we use its adversarial architecture to learn domain-shared and domain-specific representations, and extend the model to solve DC in addition to IC and SF.

Our contribution in this paper is studying the impact of using domain-specific representations on a modern multi-task DC-IC-SF model for multi-domain data. Our experiments on a real-world dataset covering three languages (German, English, Japanese) indicate that the domain-specific representations can improve model performance across all of the three SLU subtasks, especially for domains with limited training data.

2 Related Work

A majority of existing work in SLU has focused on modeling IC and SF only, for which mostly DNN-based joint models are currently used (e.g. Liu and Lane (2016); Do and Gaspers (2019); Chen et al.

Proceedings of the Second Workshop on Domain Adaptation for NLP, pages 28–32
April 20, 2021.

(2019a)). There have been also several researches studying the use of different representation types in these DNN models. He et al. (2020); Chen et al. (2019b); Do et al. (2020) used adversarial training to learn language-shared and language-specific representations for SLU in multilingual settings. Our model architecture can be considered as an extension of the BERT-based language-adversarial model proposed by Do et al. (2020). However, instead of focusing on only IC and IF, our model can deal with an extra DC subtask, which can be used effectively in real-world applications. Moreover, the adversarial training is applied on multi-domain data instead of multilingual data.

Multi-domain joint SLU models have been explored aiming to solve all three tasks via multi-task learning (e.g. Kapoor and Tirkaz (2019), Hakkani-Tür et al. (2016), Kim et al. (2017)). However, none of these studied the impact of domain-shared and domain-specific representations on model performance.

While some previous work followed the idea of leveraging domain-specific representations, we are not aware of a previous study using a full BERT-based multi-domain SLU model with domain-adversarial learning. Liu and Lane (2017) explored adversarial training in multi-domain SLU, but focus on SF only. In another work, Lee et al. (2019) applied adversarial training to obtain locale-specific and locale-agnostic features for the DC task. More recently, Qin et al. (2020) explored domain-specific parametrization for multi-domain SLU in a cascade approach, i.e. a domain classifier is applied as a first step instead of a completely joint system like our focus.

3 Method

In the following, we first describe the multi-domain baseline SLU model, which uses BERT as a single shared feature extractor, and subsequently the adversarial approach, where additionally a domain-specific feature extractor is used for each domain.

3.1 Multi-domain SLU model

We started from a common BERT-based IC-SF architecture (Do et al., 2020), and extended it for DC. In particular, our model (see Fig. 1) consists of: i) a single BERT encoder to learn domain-shared representations for words and utterances. ii) a CRF-based slot decoder for SF, iii) a 2-layer perceptron for both IC and DC. Since DC and IC are both sentence level classification tasks and there is a strong interaction between the two, as usually the intent is conditioned on the domain, we simply use the DC-IC decoder to predict a joint label of domain and intent which is of format DOMAIN_INTENT.

3.2 Domain-adversarial SLU model

We adapted the language-adversarial BERT-based architecture for joint SF and IC from Do et al. (2020) to our problem. In particular, in our system, the IC decoder predicts the DC-IC joint labels instead of only IC labels, and the language identification information is changed to domain identification information.

Our model (see Fig. 2) comprises: i) a regular pre-trained BERT encoder (enc_{bert}), ii) a 1-layer CNN encoder (enc_{shared}) to learn domain-shared representations, iii) n 1-layer CNN encoders ($enc_X, ...$) to learn domain-specific representations for n domains, iv) a 1-layer CNN encoder (enc_{domain}) is used to learn features for predicting the domain, v) a domain predictor, vi) a domain discriminator, vii) a DC-IC decoder, and viii) an SF decoder. Here, the decoders have the same architectures as in 3.1, while the domain predictor and domain discriminator are simply soft-max output layers. The information flow of the token and sentence representations between the model components can be seen in Fig. 2. Given the sentence representations from enc_{domain}, the domain predictor predicts the domain distributions of the input data, which are in turn used as weights to compute the domain-specific features from the token-level outputs of $enc_X, ...$ Meanwhile, the domain discriminator receives the sentence representations from the shared encoder as inputs and also predicts domain distributions. However, the discriminator is trained to fool the system such that the domains become indistinguishable. The domain-shared and domain-specific representations are concatenated before being fed to the DC-IC and SF decoders.

The model is trained via an adversarial training strategy. For the SF task, we use CRF loss L_s, and for the DC-IC decoder, language predictor and language discriminator we use cross-entropy loss denoted by L_{di}, L_p and L_d, respectively. Given training data annotated with intent, slot and domain labels, and $\alpha_d, \alpha_{di}, \alpha_s, \alpha_p, \beta_d$ being model hyper-parameters, i) some data batches are generated randomly, ii) $L = \alpha_d L_d$ is computed, iii) weights are updated, iv) some data batches are generated ran-

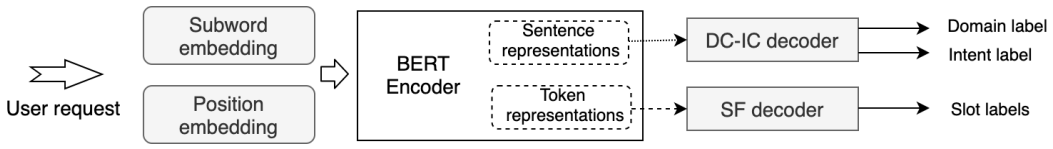

Figure 1: A BERT-based multi-task multi-domain SLU model.

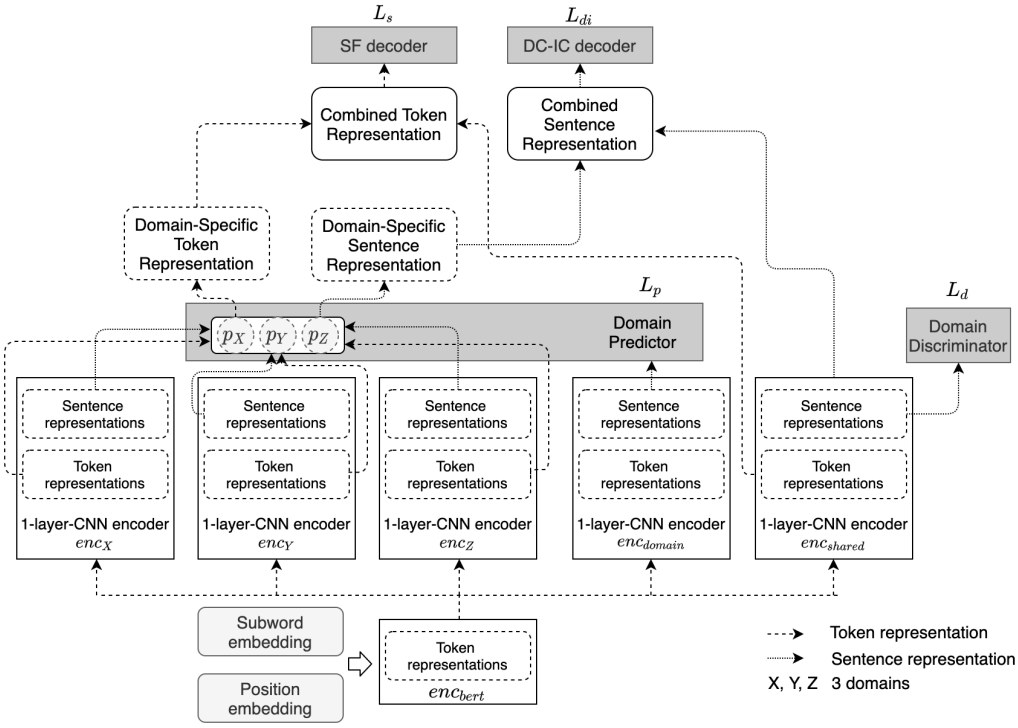

Figure 2: An adversarial model architecture for multi-domain SLU. The figure is adapted from Do et al. (2020).

domly, v) $L = \alpha_{di}L_{di} + \alpha_s L_s + \alpha_p L_p - \beta_d L_d$ is computed, and vi) model weights are updated.

4 Experiments

4.1 Datasets and settings

We extracted random data samples from large-scale commercial SLU systems for English, German and Japanese. The data is comprised of user requests to voice-controlled devices which were suitably anonymized, and each request was manually annotated with domain, intent and slot labels. For each language we used data from three domains, namely *Music*, *Books* and *Video*. In each experiment, the data is composed from all of these three domains with different data distributions. To reflect a real-world scenario, we randomly sampled different training and development data amounts per domain. For *Video*, we randomly sampled data samples of 100, 500, 1,000 and 5.000 utterances. to study domain bootstrapping with growing data amounts. For each sample, we used 90% and 10% of the data to create a training and a development set, respectively. The dataset statistics per domain are summarized in Table 2. Note that for *Video* we have different samples with growing data amounts.

For our experiments, we use pre-trained multilingual BERT (size 768) (Devlin et al., 2018). We use two dense layers of size 768 with gelu activation for the task decoders, and dropout values of 0.5 and 0.2 for the DC-IC and SF decoders, respectively. Each domain decoder has one dense layer of size 768 with gelu activation and a dropout value of 0.5. Each CNN encoder has one layer with a kernel size of three and a hidden dimension of 512. Max-pooling is used in all encoders for computing sentence representations. For adversarial models, the $\alpha_d, \alpha_{di}, \alpha_s, \alpha_p$, and β_d hyper-parameters are set to 1.0, 1.0, 1.0, 1.0, and 0.2, respectively. We train our models with a batch size of 64 for 80 epochs with early stopping using Adam optimizer with a learning rate of 0.1 and a Noam learning rate scheduler. We report F1 score for SF (computed with the CoNLL2002 script) and accuracy for IC

Lang.	#Video samples	Music			Books			Video		
		DC acc.	IC acc.	SF F1	DC acc.	IC acc.	SF F1	DC acc.	IC acc.	SF F1
German	100	-0.58	+2.11	+6.14	+0.79	+7.44	+2.56	+338.65	+10.5	+13.13
German	500	+0.03	+2.14	+5.98	-0.35	+5.29	+4.11	+75.62	+26.26	+24.92
German	1,000	-0.37	+1.93	+6.3	+1.09	+6.2	+4.09	+32.5	+22.76	+17.34
German	5,000	+0.7	+2.1	+4.32	+1.05	+4.37	+5.21	+1.69	+6.37	+6.73
German	Avg.	-0.06	+2.07	+5.69	+0.64	+5.82	+3.99	+66.38	+16.47	+15.53
English	100	-0.3	+2.02	+3.55	+1.15	+5.23	+2.65	+504.9	+153.31	+27.15
English	500	-0.61	+1.59	+4.9	+1.39	+5.25	+2.36	+47.69	+41.83	+12.85
English	1,000	-1.49	+0.46	+5.74	+0.88	+3.67	+2.31	+27.98	+29.7	+13.32
English	5,000	-2.1	-0.43	+4.57	+0.83	+3.66	+3.12	+4.44	+7.51	+5.12
English	Avg.	-1.12	+0.91	+4.69	+1.06	+4.45	+2.61	+146.25	+58.9	+14.61
Japanese	100	+0.03	+3.99	+19.84	+1.96	+8.42	+15.57	+191.02	+3.1	+100.24
Japanese	500	+0.77	+2.4	+11.62	+0.88	+5.61	+7.62	+23.53	+17.61	+44.6
Japanese	1,000	+1.2	+3.25	+14.22	+2.15	+6.61	+8.88	+4.68	+11.12	+40.6
Japanese	5,000	+1.17	+1.58	+17.31	+2.75	+6.11	+10.33	+2.72	+15.07	+19.93
Japanese	Avg.	+0.79	+2.8	+15.75	+1.93	+6.69	+10.6	+55.49	+11.72	+51.34
Avg.	Avg.	-0.13	+1.93	+8.71	+1.21	+5.65	+5.73	+89.37	+29.03	+27.16

Table 1: Relative change in domain classification (DC) accuracy, intent classification (IC) accuracy and slot (SF) F1 for domain-adversarial training compared to multi-domain training using a single shared feature as the baseline. For the *Music* domain 10,000 samples are used for model training, and 5,000 are used for *Books*. For *Video*, a growing amount of samples is used, ranging from 100 to 5,000.

Domain	Train	Dev.	Test
Music	9,000	1,000	3,000
Books	4,500	500	3,000
Video - samples 100	90	10	3000
Video - samples 500	450	50	3000
Video - samples 1,000	900	100	3000
Video - samples 5,000	4,500	500	3000

Table 2: Number of utterances per domain and dataset (available in English, Japanese and German).

and DC tasks.

5 Results

In our experiments *Video* is considered as a new domain, where initially few domain data samples are available, and data amounts are growing over time. In particular, we consider samples with sizes of 100, 500, 1,000 and 5,000 utterances for the new domain. For each experiment, the total data amounts for the domains *Music* and *Books* are kept constant at 10,000 and 5,000 utterances, respectively. For each considered Video sample size, we combine the corresponding *Video* training and development data with the other domain's training and development data, respectively. We then train i) a multi-domain SLU model as described in Section 3.1, and ii) an adversarial SLU model as described in Section 3.2, which are subsequently applied on the test datasets for all domains. The results are presented in Table 1. Due to confidentiality reasons, relative numbers are reported.

Overall, the domain-adversarial approach consistently outperforms the baseline model across all tasks and languages, except for some small fluctuations for the DC task in the *Music* domain, which has the largest data amounts in our scenario. In turn, DC performance is improved slightly for *Books* and greatly for *Video*. In particular, averaged across languages and sample sizes, DC accuracy is improved by 1.21% and 89.37% relative for the *Books* and *Video* domains, respectively. Performance gains in the *Video* domain are particularly large for smaller samples sizes. For the IC and SF tasks, the largest gains are also achieved for the *Video* domain. Averaged across languages and sample sizes, we achieve relative improvements in IC and SF of 29.0% and 27.16%, for the *Video* domain. Again, the largest relative improvements are achieved when only a few *Video* samples are available.

The results suggest that the adversarial approach is particularly useful to boost performance of domains with a limited amount of data, which could be new or low-frequency domains. While the highest gains are achieved when only few domain data is available, consistent gains in IC and SF performance are achieved for all domains and data sizes, with gains being larger for the SF task. A potential reason could be that the amount of relevant domain-specific information is higher for SF than for IC, and thus larger gains are possible by adversarial training. Note that domain-shared information can also be leveraged by the baseline model (though it may not be domain-agnostic). The results also reveal that there are language-specific

differences. Specifically, the domain-adversarial training approach seems to be particularly useful for Japanese. Averaged across sample sizes, relative improvements in slot filling performance of 15.75% and 10.6% for *Music* and *Books*, respectively, are achieved.

6 Conclusion

We studied the impact of using both domain-specific and domain-shared representations vs. using only domain-shared representations on the application of multi-domain SLU. In particular, we compared a baseline which uses BERT as a single shared feature extractor to learn domain-shared representations to an adversarial model which additionally uses a domain-specific feature extractor for each domain to learn domain-specific representations in addition to the standard domain-shared representations. Our results on a real-world dataset covering three languages indicate that by using a domain-adversarial approach, model performance can be improved across all of the three SLU subtasks. Performance gains were particularly large for the use case of bootstrapping a new domain, where little target domain data are available.

References

Q. Chen, Z. Zhuo, and W. Wang. 2019a. Bert for joint intent classification and slot filling. *arXiv:1902.10909*.

Xilun Chen, Ahmed Hassan Awadallah, Hany Hassan, Wei Wang, and Claire Cardie. 2019b. Multi-source cross-lingual model transfer: Learning what to share. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3098–3112, Florence, Italy. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Quynh Do, Judith Gaspers, Tobias Roeding, and Melanie Bradford. 2020. To what degree can language borders be blurred in BERT-based multilingual spoken language understanding? In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2699–2709, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Quynh Ngoc Thi Do and Judith Gaspers. 2019. Cross-lingual transfer learning for spoken language understanding. *Proceedings of the 2019 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*.

Anuj Kumar Goyal, Angeliki Metallinou, and Spyros Matsoukas. 2018. Fast and scalable expansion of natural language understanding functionality for intelligent agents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 3 (Industry Papers)*, pages 145–152, New Orleans - Louisiana. Association for Computational Linguistics.

Dilek Hakkani-Tür, Gökhan Tür, Asli Çelikyilmaz, Yun-Nung Chen, Jianfeng Gao, Li Deng, and Ye-Yi Wang. 2016. Multi-domain joint semantic frame parsing using bi-directional RNN-LSTM. In *Interspeech 2016, 17th Annual Conference of the International Speech Communication Association, San Francisco, CA, USA, September 8-12, 2016*, pages 715–719. ISCA.

K. He, W. Xu, and Y. Yan. 2020. Multi-level cross-lingual transfer learning with language shared and specific knowledge for spoken language understanding. *IEEE Access*, 8:29407–29416.

Shubham Kapoor and Caglar Tirkaz. 2019. Bootstrapping nlu models with multi-task learning.

Young-Bum Kim, Sungjin Lee, and Karl Stratos. 2017. ONENET: joint domain, intent, slot prediction for spoken language understanding. In *2017 IEEE Automatic Speech Recognition and Understanding Workshop, ASRU 2017, Okinawa, Japan, December 16-20, 2017*, pages 547–553. IEEE.

Jihwan Lee, Ruhi Sarikaya, and Young-Bum Kim. 2019. Locale-agnostic universal domain classification model in spoken language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 2 (Industry Papers)*, pages 9–15. Association for Computational Linguistics.

Bing Liu and Ian Lane. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. In *INTERSPEECH*.

Bing Liu and Ian Lane. 2017. Multi-domain adversarial learning for slot filling in spoken language understanding.

Libo Qin, Minheng Ni, Yue Zhang, Wanxiang Che, Yangming Li, and Ting Liu. 2020. Multi-domain spoken language understanding using domain- and task-aware parameterization.