LREC-COLING 2024

Source-free Domain Adaptation for Aspect-based Sentiment Analysis

Zishuo Zhao, Ziyang Ma, Zhenzhou Lin, Jingyou Xie, Yinghui Li, Ying Shen

School of Intelligent Systems Engineering, Sun Yat-Sen University, Shenzhen, China
Tsinghua Shenzhen International Graduate School, Tsinghua University

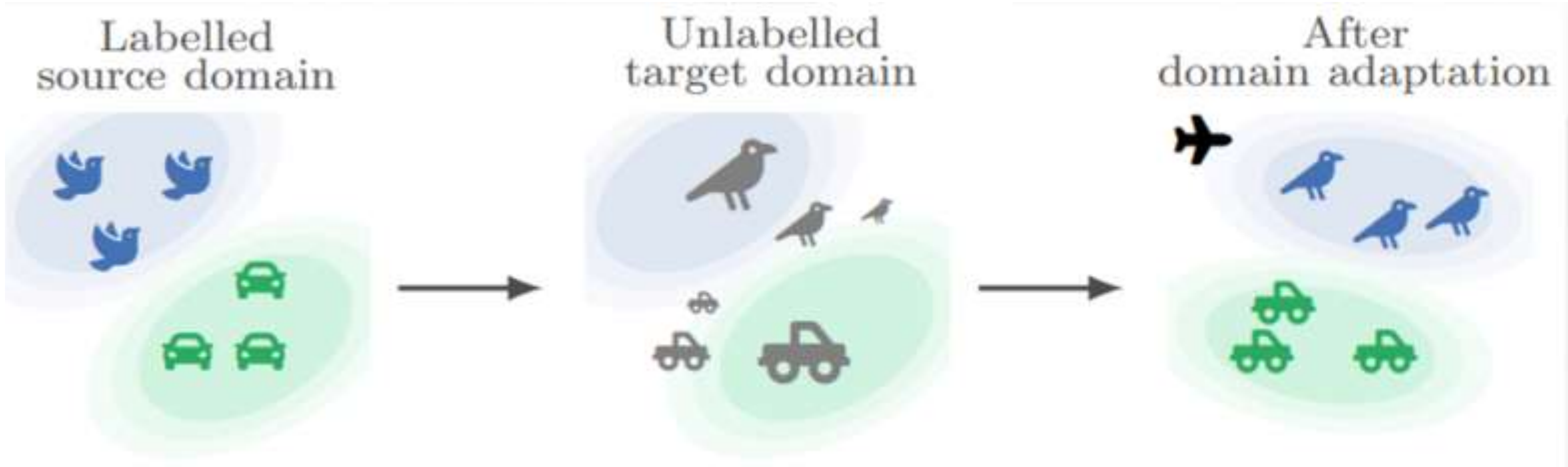
The 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation

Outline

- Introduction
- Motivation
- Methodology
  - Feature-Based Domain Adaptation
  - Source Model Generation
  - Domain Adaptation with Self-supervised Pseudo-labeling
- Experiment
  - Comparison with source-free baselines
  - Comparison with source-needed SOTA methods
  - Comparison with source-free SHOT(CV baseline)
- Summary

Introduction

- What is Domain Adaptation ?



- What is source-free?

Introduction

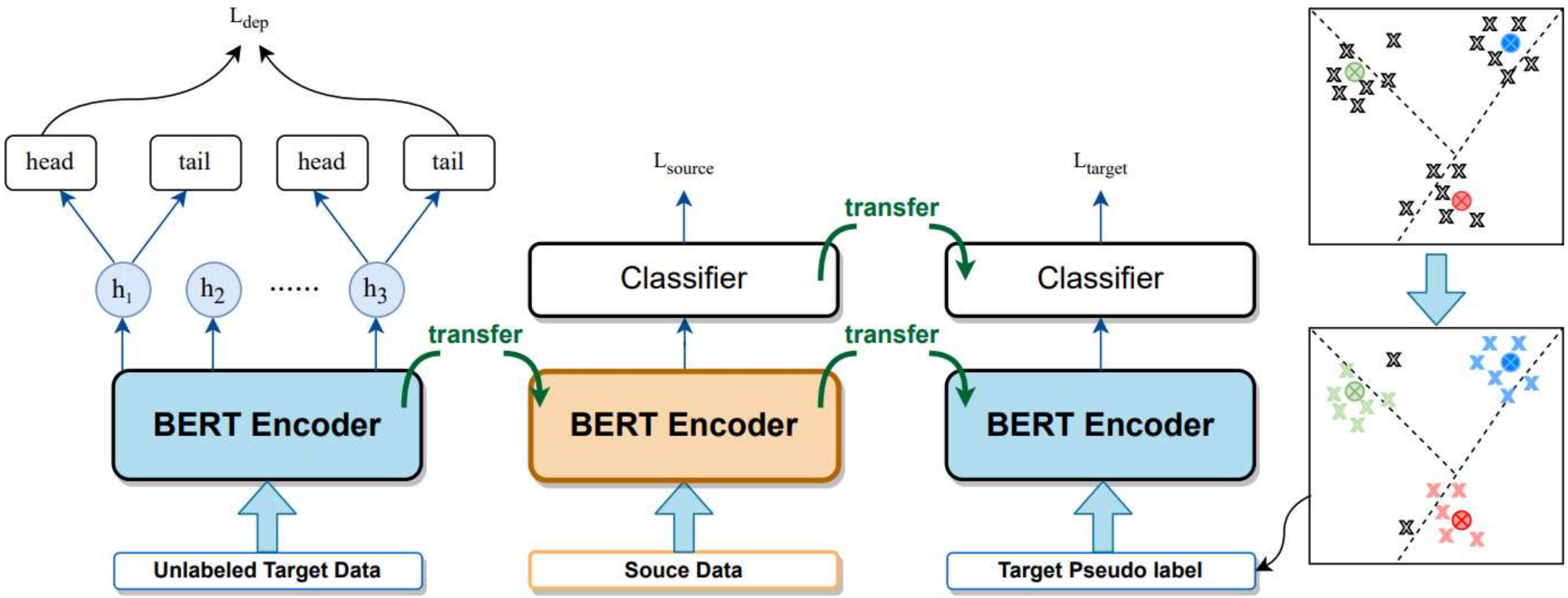
□ Challenges

- Considering that real-world data are distributed across different devices and often contain private information (e.g., data on personal phones or surveillance cameras), existing DA methods require access to source domain data during the learning process, potentially violating data privacy.

- Feature distribution shifts between different domains. For example, "battery" is an aspect term in the laptop domain but rarely in the service domain. The model needs some latent information to judge their potential similarity.

Methodology

- Overview of Our Method



Methodology

Feature-Based Domain Adaptation:

With domain-shared knowledge and using Mask Language Model (MLM) task, we make the model embed the target domain data into feature space in advance. Then we transport the feature-embedded model to the source domain to train on source domain data.

- First extract the characteristics of their head entities and tail entities:

$$h_i^{head} = \tanh(W_1 h_i + b_1),$$

$$h_i^{tail} = \tanh(W_2 h_i + b_2),$$

- Then the model predict their dependencies :

$$m_{ij} = \left[h_i^{head}; h_j^{tail}; h_i^{head} - h_j^{tail}; h_i^{head} \cdot h_j^{tail}\right],$$

$$p_{ij}^{dep} = softmax\left(W_{dep} m_{ij} + b_{dep}\right),$$

- Finally, cross-entropy prediction is used to correct the relationship predicted by the model:

$$L_{dep} = \sum_{X^t} \sum_{i}^{T} \sum_{j}^{T} \mathbb{I}(ij)\, CrossEntropy\left(p_{ij}^{dep}, y_{ij}^{dep}\right),$$

Methodology

Source Model Generation

- After the feature embedding, we obtain the model M_0 embedded in the feature space of the target domain data. We transfer the model from the target domain to the location of the source domain and use this model as the basic model for source domain training. Then we train the model on the source domain data by minimizing the cross entropy where the source domain is located:

$$L_{source} = CrossEntropy\left(M_0(x_s), y^{label}\right)$$

  - where x_s is the source domain data and y^{label} is the ABSA label corresponding to the source domain data.

- After training at the location of the source domain, we obtain the well-trained source model M_s. We transfer the source model M_s to the target domain location for the domain transfer process of the target domain.

Methodology

Domain Adaptation with Self-supervised Pseudo-labeling:

we utilize the source domain model to predict the labels and get the feature vectors from the hidden layer of the model. We use the labels and corresponding features as anchor points to reassign the labels of the target domain data according to the distance.

- First, predict the data of the target domain based on the source model:

$$y^t = argmax(\delta_k(M_s(x_t))),$$

- Then, based on the predicted labels, calculate the center point of the category:

$$A_k^{(0)} = \frac{\sum_{x_t \in X_t} \delta_k\big(M_s(x_t)\big) * F_s(x_t)}{\sum_{x_t \in X_t} \delta_k\big(M_s(x_t)\big)},$$

- Then we reassign the labels of the target domain samples by calculating the distance between each feature and the center feature of the category:

$$\hat{y}^t = arg\min_k Dist(A_k^{(0)}, F_s(x_t)).$$

Methodology

Domain Adaptation with Self-supervised Pseudo-labeling

Using the allocated pseudo labels, recalculate the center point and iterate through the loop.

- Formula for calculating category center points :

$$A_k^{(m)} = \frac{\sum_{x_t \in X_t} \mathbb{I}(\hat{y}^t = \mathrm{k}) * F_s(x_t)}{\sum_{x_t \in X_t} 1(\hat{y}^t = \mathrm{k})},$$

$$\hat{y}^t = arg \min_k Dist(A_k^{(m)}, F_s(x_t)),$$

  - Where m represents the current number of clustering iterations, $\mathbb{I}(\cdot)$ is an indicator function.

- When the feature center point of the target domain converges, the iteration will stop. When the iteration stops, we use the confidence threshold $\tau \in (0,1)$ to filter out lower quality answers and improve the quality of pseudo labels.

$$D_t' = \left\{(x_i^t, \hat{y}_i^t) \middle| Dist\left(F_s(x_i^t), A_{\hat{y}_i^t}\right) < \tau\right\}_{i=1}^{n_t}.$$

Experiment

Datasets

We conduct experiments on four benchmark datasets: Laptop(L), Restaurant (R), Device (D) and Service (S).

- **Laptop(L)**: Restaurant (R) is the union set of the restaurant datasets from SemEval ABSA challenge 2014, 2015 and 2016
- **Restaurant (R)**: Laptop (L) containing user reviews from the laptop domain, it is from SemEval-2014 ABSA challenge.
- **Device (D)**:Device (D) is a combination of device reviews from 5 different digital products.
- **Service (S)**:Service (S) contains reviews from web services, which is introduced by Hu and Liu.

Dataset	Domain	Sentences	Training	Testing
L	Laptop	3,845	3,045	800
R	Restaurant	6,035	3,877	2,158
D	Device	3,836	2,557	1,279
S	Service	2,239	1,492	747

Experiment

Comparison with SOTA methods

Methods	R→S	L→S	D→S	S→R	L→R	D→R	S→L	R→D	AVG
Source-Only									
BERT-Base (Devlin et al., 2018)	19.48	25.78	30.31	42.2	40.38	30.06	29.20	29.47	30.36
Source-Required									
Hier-Joint (Ding et al., 2017)	15.56	13.90	19.04	31.10	33.54	32.87	22.65	24.53	23.71
RNSCN (Wang et al., 2018)	20.04	16.59	20.03	33.21	35.65	34.60	18.87	33.26	26.09
AD-SAL (Li et al., 2019a)	28.01	27.20	26.62	41.03	43.04	41.01	27.04	35.44	33.71
BERT-DANN (Gong et al., 2020)	21.60	25.10	18.62	45.84	41.73	34.68	30.47	34.41	30.83
BERT-UDA (Gong et al., 2020)	33.12	27.89	28.03	47.09	45.46	42.68	**34.77**	34.93	35.98
CDRG (Indep) (Yu et al., 2021)	34.10	33.97	31.08	44.46	44.96	39.42	26.81	25.25	34.27
CDRG (Merge) (Yu et al., 2021)	35.14	**38.14**	37.22	47.92	**49.79**	**47.64**	33.69	27.46	**38.98**
Source-Free									
SF-ABSA (Feature-based only)	26.44	26.05	31.83	**48.78**	40.72	41.16	34.33	**36.64**	35.87
SF-ABSA (All)	**35.67**	29.62	**45.93**	44.62	44.23	35.43	34.01	28.56	37.14

- First, the proposed SF-ABSA framework performs better compared to BERT-base, which indicates that SF-ABSA has better domain adaptability. Furthermore, our SF-ABSA framework achieves comparable performance to source-required methods on the premise of only transferring model parameters without accessing source-domain data, which demonstrates the significance of our approach in the privacy-preserving domain.

Experiment

Comparison with SOTA methods

Methods	R→S	L→S	D→S
Source-Required			
BERT-DANN (Gong et al., 2020)	21.60	25.10	18.62
BERT-UDA (Gong et al., 2020)	33.12	27.89	28.03
CDRG (Yu et al., 2021)	35.14	**38.14**	37.22
GCDDA (Li et al., 2022)	32.07	27.22	28.52
Source-Free			
SF-ABSA(our)	**35.67**	29.62	**45.93**

- In addition, when the target domain is the Service domain dataset, the pseudo-label based approach has a significant effect on performance improvement, as shown in the Table , which we believe is due to the overall smaller size of the Service dataset and the relatively small total number of words contained in the sentences, which results in less noise in the computation of the category feature anchors and higher accuracy of pseudo-label in the reassignment.

Experiment

Comparison with SHOT methods(CV baseline)

Methods	BERT-Base	SHOT	SF-ABSA
R→S	42.2	26.52	**35.67**
L→S	20.99	17.00	**29.62**
D→S	13.64	10.45	**45.93**
S→R	42.2	37.95	**44.62**
L→R	39.14	33.57	**44.45**
D→R	30.06	26.94	**35.43**

- In the above experiments, the conditions on which the various models are based are not consistent, and it is unfair to directly compare with each other, so we choose the classic baseline SHOT in the source-free field for comparison. SHOT is a method in the field of computer vision, but it has a strong generalization ability and can be generalized to our ABSA task for comparison. Tabel shows the comparison of our proposed method with other methods under the same source-free setting. We compared our method with SHOT under the same source-free setting, and we found that our method has obvious advantages. SHOT's method, although highly generalizable, still does not achieve great performance, illustrating the progress of our method in the field of privacy

Experiment

The effect of the hyperparameter

Domain	m=1	m=2	m=3
R→S	**35.67**	31.61	30.72
L→S	**29.62**	25.12	25.68
D→S	**45.93**	41.98	40.56
S→R	**44.62**	41.79	41.32
L→R	**44.45**	42.54	40.19
D→R	**35.43**	32.15	31.08

- Table shows the effect of the hyperparameter m on the performance of the model in the pseudo label-based method. m represents the number of times to iteratively calculate the category center point and then reassign the pseudo-label. We found that the pseudo-label obtained after one iteration is the best.

Summary

- To the best of our knowledge, we are the first to explore domain adaptation for the ABSA task without access to source domain data, which has great significance for protecting privacy.

- We propose the SF-ABSA framework, encompassing both feature-based adaptation and pseudo-label-based adaptation, which can transfer knowledge using only the parameters of the model.

- Experimental results show that our framework achieves competitive results compared to the state-of-the-art results on ABSA for unsupervised domain adaptation.

Thanks

Thank you !

Zishuo Zhao (zhaozsh@mail2.sysu.edu.cn)

School of Intelligent Systems Engineering, Sun Yat-Sen University (SYSU)

The 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation