Knowledge-Based Systems 260 (2023) 110173

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Multi-source domain adaptation method for textual emotion classification using deep and broad learning

Sancheng Peng [a], Rong Zeng [b], Lihong Cao [a,*], Aimin Yang [c], Jianwei Niu [d], Chengqing Zong [e], Guodong Zhou [f]

[a] Laboratory of Language Engineering and Computing, Guangdong University of Foreign Studies, Guangzhou, 510006, China
[b] Guangdong Provincial Key Laboratory of Nanophotonic Functional Materials and Devices, South China Normal University, Guangzhou, 510006, China
[c] School of Computer Science and Intelligence Education, Lingnan Normal University, Zhanjiang, 524048, China
[d] State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, 100191, China
[e] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China
[f] School of Computer Science and Technology, Soochow University, China

ARTICLE INFO

Article history:
Received 9 May 2022
Received in revised form 29 November 2022
Accepted 4 December 2022
Available online 6 December 2022

Keywords:
Multi-domain
Emotion classification
BERT
Broad learning
Bi-LSTM

ABSTRACT

Existing domain adaptation methods for classifying textual emotions have the propensity to focus on single-source domain exploration rather than multi-source domain adaptation. The efficacy of emotion classification is hampered by the restricted information and volume from a single source domain. Thus, to improve the performance of domain adaptation, we present a novel multi-source domain adaptation approach for emotion classification, by combining broad learning and deep learning in this article. Specifically, we first design a model to extract domain-invariant features from each source domain to the same target domain by using BERT and Bi-LSTM, which can better capture contextual features. Then we adopt broad learning to train multiple classifiers based on the domain-invariant features, which can more effectively conduct multi-label classification tasks. In addition, we design a co-training model to boost these classifiers. Finally, we carry out several experiments on four datasets by comparison with the baseline methods. The experimental results show that our proposed approach can significantly outperform the baseline methods for textual emotion classification.

© 2022 Published by Elsevier B.V.

1. Introduction

Nowadays, existing textual emotion classification methods focus on investigating single-domain [1]. However, few of them explore cross-domain emotion classification. In addition, most existing approaches for domain adaptation focus on performing cross-domain sentiment classification (CDSC) under a single-source scenario. If there is a prominent difference in feature distribution between source and target domains, its performance will decline heavily, which is called "negative transfer" [2–4]. However, multi-source domain adaptation (MDA) is more feasible and valuable than that single-source domain adaptation, which has attracted more and more attention in academia and industry.

Existing methods for CDSC may be classified into two types: single-domain and multi-domain. The single-domain method aims to utilize the useful knowledge learned from a source domain to help the target domain. While the multi-domain method needs to learn useful knowledge from multiple source domains. For example, Khan et al. [5] adopted SVM for CDSC; Lin et al. [6] adopted generative adversarial networks (GAN) [7] for CDSC. Khan's method usually applies the similarity-based transfer learning approach for capturing domain invariant features. However, they may neglect the long-distance interdependent features. Lin's method usually utilizes GAN to extract the domain-invariant features. However, it needs to solve the intrinsic problem of GAN variants in training stability.

Despite the progress made by the above methods, there are still many challenges for MDA. The first is that the same word may deliver different emotions in different domains. For instance, the word "long" may deliver many different emotions. In the domain of smartphones, "long" expresses a positive emotional tendency, while "long" expresses a negative emotional tendency in catering service. The second is that long-distance interdependent features are difficult to be captured effectively. The last is that most existing methods on MDA are DL-based models, which have some disadvantages (e.g., GAN may fall into the problem of non-convergence and model collapse).

* Corresponding author.
E-mail addresses: psc346@aliyun.com (S. Peng), zengrong980302@163.com (R. Zeng), 201610130@oamail.gdufs.edu.cn (L. Cao), amyang18@163.com (A. Yang), niujianwei@buaa.edu.cn (J. Niu), cqzong@nlpr.ia.ac.cn (C. Zong), gdzhou@suda.edu.cn (G. Zhou).

https://doi.org/10.1016/j.knosys.2022.110173
0950-7051/© 2022 Published by Elsevier B.V.

To address the above challenges, we present a novel approach, called Multi-source Broad Learning (MBL), for MDA-based emotion classification. Specifically, we first design a model to extract domain-invariant feature (DIF) from each source domain to the same target domain, by adopting bidirectional encoder representation from transformers (BERT) [8] and bi-directional long short-term memory network (Bi-LSTM) [9]. Then, based on DIFs, multiple classifiers can be trained by exploiting broad learning (BL) [10] with the labeled data. In addition, we design a co-training model to boost together these classifiers. Finally, extensive experiments are conducted on four datasets by comparison with the baseline methods. The experimental results show that MBL outperforms the baseline methods.

The main contributions in this article are summarized as follows:

- We present a new transfer learning approach to address the MDA-based emotion classification task, by combining DL (i.e., Bi-LSTM and BERT) and BL models.
- We adopt BERT and Bi-LSTM to design an extraction model for DIF, which is exploited to train multiple classifiers. And then a co-training model is designed to boost these classifiers.
- We collect four real-world datasets involving four different domains from public E-commerce platforms, for MDA-based emotion classification. The relevant experimental results show that our proposed approach can significantly outperform the baseline methods.

The remainder of this article is organized as follows: In Section 2, we introduce the research progress of multi-source CDSC and BL. In Section 3, we describe the structure and principle of MBL. In Section 4, we introduce experimental datasets, compare the proposed method with baseline methods, and provide analysis of experimental results. In Section 5, we conclude this article and discuss our future work.

2. Related work

In this section, we introduce related work from three dimensions. The first dimension is related to the multi-source domain adaptation method; the second is related to the cross-domain sentiment classification method; and the last is related to BL.

2.1. Multi-source domain adaptation

Li et al. [11] investigated the domain discrepancy between pairwise sources and provided a better bound for it. Hoffman et al. [12] proposed a method to derive a bound for the domain discrepancy by using DC programming. Guo et al. [4] presented a method for multi-source CDSC, based on a point-to-set distance metric. Chen et al. [2] presented a GAN-based framework for MDA, by using generative adversarial nets. Wright and Augenstein [13] explored the problem of MDA by using large pre-training transformer models, domain adversarial training, and the mixture of expert techniques. Yin et al. [14] proposed a universal framework for MDA by using a pseudo-margin vector.

2.2. Cross-domain sentiment classification

Existing methods for CDSC may be classified into as follows: single-source CDSC and multi-source CDSC.

(1) Single-source CDSC

Wang et al. [15] presented a CDSC method by integrating two non-negative matrix tri-factorizations into a joint optimization framework. Zhang et al. [16] presented a CDSC method by using an interactive attention transfer network. Li et al. [17] presented a CDSC approach based on a hierarchical attention transfer network. Du et al. [18] proposed a model to derive the domain-invariant features using BERT. Du et al. [19] proposed a framework to share domain-invariant information between the source and target domain, based on the Wasserstein-based transfer network. Zhou et al. [20] proposed a sentiment-aware pre-trained model (i.e., SENTIX) to learn domain-invariant sentiment knowledge from large-scale review datasets. Peng and Zhang [21] proposed a weighted domain-invariant representation learning framework for CDSC.

(2) Multi-source CDSC

Khan et al. [5] presented a method for multi-source CDSC, based on cosine similarity and the SVM model. Zhao et al. [22] presented a method for multi-source CDSC tasks based on MDA and joint learning. Xu et al. [23] proposed an MDA approach for CDSC, called HANN. Yang et al. [3] presented a MDA method with a Granger-causal objective for CDSC. Lin et al. [6] presented an MDA method for visual sentiment classification, called MSGAN. Dai et al. [24] proposed a method for multi-source CDSC by using multi-task learning. Zhao et al. [25] presented an instance-level MDA framework, called C–CycleGAN, for cross-domain textual sentiment classification with multi-sources. Dai et al. [26] proposed an MDA approach for unsupervised CDSC by designing an adversarial shared-private model. Fu and Liu [27] proposed an MDA method for unsupervised CDSC based on the Wasserstein distance.

2.3. Broad learning

BL was presented by Chen and Liu [10], which includes feature mapping nodes and enhancement nodes in a wide manner, instead of stacking and deeply expand neurons. Then, the output weight is calculated by the pseudo inverse. Compared with DL methods, BL takes on some advantages like simple network structure, short training time, and strong generalization ability. It is an effective alternative method for DL, and need not be retrained when new nodes are added in the training process, only need to extend the incremental learning model. BL has achieved better results in many applications, such as emotion classification [28], cross-domain emotion classification [29], and negative emotion detection [30].

In summary, different from these methods, we introduce Kullback Leibler (KL) [31], DL, and BL to extract DIFs concurrently, and train multiple classifiers based on DIFs.

3. Approach

At first, we provide a problem definition for MDA-based emotion classification and introduce the KL metric to measure the divergence between the probability distribution of any two random variables. Then, we provide a detailed description of how to obtain DIF between the target domain and each source domain. At last, we further describe how to fuse them by using the co-training algorithm. The framework of MBL is shown in Fig. 1.

3.1. Problem definition

For MDA task, we define k source domains $D_s = \{D_{s1}, D_{s2}, \ldots, D_{sk}\}$ and a target domain D_t. Suppose that we have labeled data $X_{sj} = \{x_{sj}^i, y_{sj}^i\}_{i=1}^{N_{sj}}, j = 1, 2, \ldots, k$ in the jth source domain D_{sj}, where N_{sj} denotes the total number of labeled data for D_{sj}. In addition, suppose that we also have a set of unlabeled data $X_{tu} = \{x_{tu}^i\}_{i=1}^{N_{tu}}$ and a few labeled data $X_{tl} = \{x_{tl}^i, y_{tl}^i\}_{i=1}^{N_{tl}}$ in D_t, where N_{tu} and N_{tl} denote the total number of unlabeled data and labeled data for D_t, respectively. The purpose of multi-source cross-domain emotion classification is to train k robust classifiers by utilizing the labeled data in multi-source and target domains and then adopting them to predict the unlabeled target data.

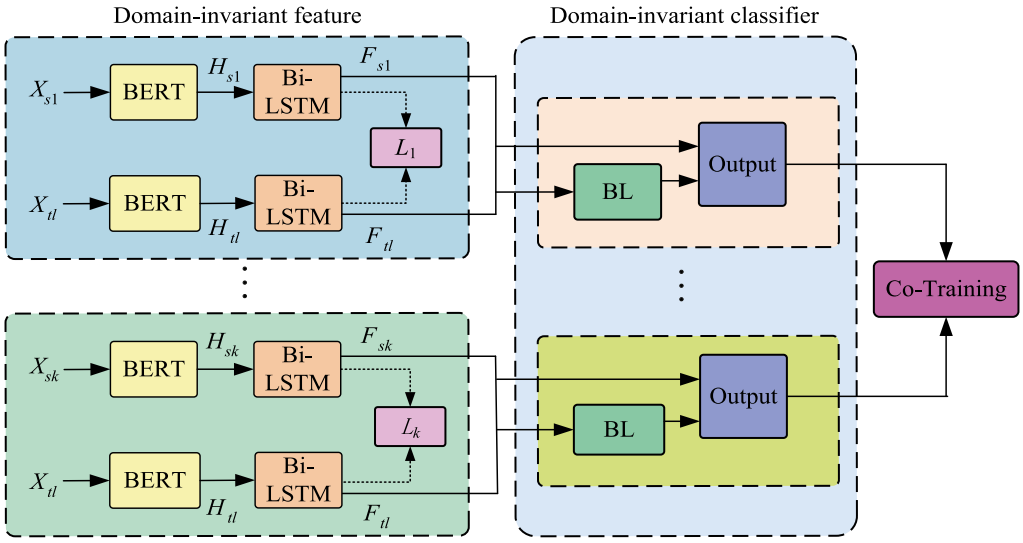

Fig. 1. The framework of MBL.

3.2. Kullback–Leibler divergence (KL)

KL was proposed to measure the probability distribution divergence between any two random variables. As a measure for the asymmetry of the two probability distributions, KL has been widely adopted to reduce domain shifts in domain adaptation.

Given two domains D_s and D_t, KL is defined as follows:

$$KL\left(X_{sj}, X_t\right) = \sum_{i=1}^{N_{sj}} x_{sj}^i \log\left(\frac{x_{sj}^i}{x_{tl}^i}\right) \tag{1}$$

where N_{sj} denotes the amount of samples of jth source domain. When two distributions are the same, KL between them is set to zero.

3.3. Overview of broad learning

The main idea of BL is that word embeddings generated by BERT are linearly mapped into m groups of feature nodes and then feature nodes are nonlinearly mapped into n groups of enhancement nodes. Finally, feature nodes and enhancement nodes are input into the output layer to obtain the probability distribution of emotions. During the training process of BL, the weights of feature nodes and enhancement nodes are generated randomly and fixed, and the weights of the output layer are optimized by the ridge regression method.

Given a train set $\{X, Y\} \in \mathrm{R}^{N\times(D+C)}$, where N denotes the total number of samples, D denotes the dimension of word embedding, and C denotes the number of emotion types, word embeddings X are linearly mapped into m groups of feature nodes. The hth group of feature nodes F_h is represented as follows:

$$F_h = \phi\left(X\theta_{fh} + \beta_{fh}\right) \in \mathrm{R}^{N\times p}, h = 1, 2, \ldots, m \tag{2}$$

where ϕ denotes linear activation function, p denotes the number of feature nodes in each group, and θ_{fh} and β_{fh} denote weight and bias generated randomly, respectively.

Suppose $F^m \triangleq [F_1, F_2, \ldots, F_m]$ denotes the matrix concatenated by m groups of feature nodes. F^m is nonlinearly mapped into n groups of enhancement nodes. The ith group of enhancement nodes E_i is represented as follows:

$$E_i = \varphi\left(F^m\theta_{ei} + \beta_{ei}\right) \in \mathrm{R}^{N\times q}, i = 1, 2, \ldots, n \tag{3}$$

where φ denotes nonlinear activation function (e.g., tanh, sigmoid), q denotes the total number of enhancement nodes in each group, and θ_{ei} and β_{ei} denote weight matrix and bias matrix generated randomly, respectively.

Suppose $E^n \triangleq [E_1, E_2, \ldots, E_n]$ denotes the matrix concatenated by n groups of enhancement nodes. The actual input of BL is concatenated by F^m and E^n, which can be represented as follows:

$$A = \left[F^m, E^n\right] \in \mathrm{R}^{N\times(mp+nq)} \tag{4}$$

Thus, the final output of BL can be represented as follows:

$$\begin{aligned}\hat{Y} &= \left[g\left(X\theta_{f1} + \beta_{f2}\right), \ldots, g\left(X\theta_{fm} + \beta_{fm}\right)\right. \\ &\quad \left.|\varphi\left(F^m\theta_{e1} + \beta_{e1}\right), \ldots, \varphi\left(F^m\theta_{en} + \beta_{en}\right)\right]\theta \\ &= [F_1, F_2, \ldots, F_n\,|E_1, E_2, \ldots, E_m\,]\,\theta \\ &= \left[F^n, E^m\right]\theta \\ &= A\theta\end{aligned} \tag{5}$$

where θ denotes the weights of output layer.

Thus, according to matrix analysis theory, θ can be represented as follows:

$$\theta = A^+Y \tag{6}$$

where A^+ denotes the pseudo inverse matrix of A.

The overall training process of BL is listed in Algorithm 1.

Algorithm 1 The training process of BL

Input: input data X
Output: output weights θ
1: **for** $k = 1; k \le m$ **do**
2: randomly generate θ_{fk}, β_{fk}
3: calculate $F_k = \phi\left(X\theta_{fk} + \beta_{fk}\right)$
4: **end for**
5: set feature nodes $F^m \triangleq [F_1, F_2, ..., F_m]$
6: **for** $i = 1; i \le n$ **do**
7: randomly generate θ_{ei}, θ_{ei}
8: calculate $E_i = \varphi\left(F^m\theta_{ei} + \beta_{ei}\right)$
9: **end for**
10: set enhancement nodes $E^n \triangleq [E_1, E_2, ..., E_n]$
11: set $A = [F^m, E^n]$ and calculate output weights θ

According to the analysis on Algorithm 1, the time complexity of BL is $O(rmp + rnq)$.

3.4. Feature extraction

Since BERT is a self-supervised learning method, it conducts pre-training on large-scale unlabeled corpus by using a Transformer Encoder structure and converts the distance between two words at any position into vector representation by using the attention mechanism. It can effectively solve the problem of long-distance dependency in natural language processing, and simultaneously obtain rich semantic information in a text. In addition, compared with LSTM, CNN, and RNN, Bi-LSTM can better capture contextual information in a text. Thus, to effectively conduct MDA-based emotion classification, we adopt BERT and Bi-LSTM to extract DIF.

Firstly, we utilize BERT to generate word embeddings for source labeled data X_{sj} in jth source domain X_{sj}, and target labeled data X_{tl}, which are represented as follows:

$$\begin{aligned} H_{sj} &= BERT\left(X_{sj};\theta_j^1\right) \in \mathrm{R}^{(N_{sj}l)\times 768} \\ H_{tl} &= BERT\left(X_{tl};\theta_j^1\right) \in \mathrm{R}^{(N_{tl}l)\times 768} \end{aligned} \tag{7}$$

where *BERT* denotes the representations of DIF encoded by BERT, θ_j^1 denotes the corresponding parameter, and l denotes the sequence length.

Then, based on these representations, the contextual features and long-distance dependencies can be extracted by Bi-LSTM. The encoding results for jth source domain X_{sj} and X_{tl} can be described as follows:

$$\begin{aligned} F_{sj} &= BiLSTM\left(H_{sj};\theta_j^2\right) \in \mathrm{R}^{N_{sj}\times r} \\ F_{tl} &= BiLSTM\left(H_{tl};\theta_j^2\right) \in \mathrm{R}^{N_{tl}\times r} \end{aligned} \tag{8}$$

where *BiLSTM* denotes the representations of DIF encoded by Bi-LSTM, θ_j^2 denotes the corresponding parameter, and r denotes the representation dimension.

As to DIF, we hope it can encode features shared by both source and target domains. From the probability distribution view, we hope that the distributions of the mapped outputs obtained by DIF from source and target data are similar. Thus, we utilize KL regularizer onto the features of source data F_{sj} and target data F_{tl}. KL divergence can be defined as follows:

$$L_j^{kl} = KL\left(F_{sj}, F_{tl}\right) \tag{9}$$

The distribution discrepancy between F_{sj} and F_{tl} can be reduced by minimizing the loss L_j^{kl}, which contributes to obtain DIF.

3.5. BL-based classifier

BL has two advantages: (1) The features can be nonlinearly mapped into high-dimensional feature space to further extract deep semantic information; (2) It needs only to update the weights of the output layer by ridge regression, so the computational cost is very low. Based on textual features extracted by BERT and Bi-LSTM, BL is adopted to further capture semantic features in texts from both source and target domains, and to design a domain-invariant classifier (DIC) based on DIF.

According to BL theory, F_{sj} and F_{tl} are nonlinearly mapped into n groups of enhanced nodes. Thus, the ith group of enhanced nodes for the jth source domain can be represented as follows:

$$E_{ij} = \varphi\left(\left[F_{sj}, F_{tl}\right]\theta_{ei} + \beta_{ei}\right) \in \mathrm{R}^{(N_{sj}+N_{tu})\times q},\ i = 1, 2, \ldots, n \tag{10}$$

Suppose $E_j^n \triangleq \left[E_{1j}, E_{2j}, \ldots, E_{nj}\right]$ denotes the matrix concatenated by n groups of enhancement nodes of the jth source domain. Therefore, the output of DIC can be described as follows:

$$\hat{Y}_j = \left[F_{sj}, F_{tl}, E_j\right]\theta_j^3 = A_j\theta_j^3 \tag{11}$$

where A_j denotes all the input features of DIC, and θ_j^3 denotes the weight of output layer of DIC.

In the inference stage to predict X_{tu}, we weight and sum the output of each DIC to obtain the emotion of the unlabeled target samples, which can be described as follows:

$$\hat{Y} = \sum_{j=1}^{k} \hat{Y}_j \tag{12}$$

3.6. Co-training

The co-training process of this method is divided into two steps: (i) training an encoder of DIF for each source–target domain pair; (ii) training a classifier DIC for each DIF. As to DIF, the classification loss on DIF is utilized to measure the difference between ground truth and the prediction of the source domain and target domain. Since minimum entropy can help to constrain the predicted values of target domain samples and to increase the distance between the target sample and classification decision boundary, to obtain higher confidence for the prediction results in the target domain, we conduct entropy minimization for the probability distribution of each kind of sample, and its loss function is represented as follows:

$$\begin{aligned} L_j^c = &-\frac{1}{N_{sj}+N_{tl}}\sum_{i=1}^{N_{sj}} y_{sj}^i \log Q\left(y_{sj}^i \middle| F_{sj}^i\right) \\ &-\eta_j\frac{1}{N_{sj}+N_{tl}}\sum_{m=1}^{N_{tl}} y_{tl}^m \log Q\left(y_{tl}^m \middle| F_{tl}^m\right) \end{aligned} \tag{13}$$

where F_{sj}^i and F_{tl}^m denote the encoding result of Bi-LSTM for the ith source example x_{sj}^i and the mth target example x_{tl}^m, respectively, and η_j denotes the weight of target domain loss.

Thus, the total loss function includes the difference measured by KL divergence between feature distributions of each source and target domain, and the minimum entropy loss in each source and target domain, which is represented as follows:

$$L = \sum_{j=1}^{k}\left(L_j^c + \delta_j L_j^{kl}\right) \tag{14}$$

where δ_j denotes the weight of loss L_j^{kl}.

As to DIC, we need to obtain an appropriate θ_j^3 to maintain the difference between Y_j and $\hat{Y}_j$ as small as possible. Thus, the ridge regression is adopted as objective function, which is described as follows:

$$\underset{\theta_j^3}{\operatorname{argmin}}\left(\left\|Y_j - \hat{Y}_j\right\|_2^2 + \lambda\left\|\theta_j^3\right\|_2^2\right) \tag{15}$$

where λ denotes the regularization parameter and Y_j denotes the ground truth label for the data of jth source and target.

Thus, according to the regularized least square method, θ_j^3 can be represented as follows:

$$\theta_j^3 = \left(\lambda I + A_j A_j^{\mathrm{T}}\right)^{-1} A_j^{\mathrm{T}} Y_j \tag{16}$$

Here, we regard DIF for each source–target domain pair as independent spaces given because of target domain data. Based on DIF, we train DIC concerning for to parameter θ_j^3 and employ both a small number of target-labeled data and source-labeled data to train DIC.

Thus, through the above deducing, the algorithm of MBL is shown in Algorithm 2.

Table 1
Statistics of datasets.

Name	Happy	Moving	Sad	Angry	Fear	Disgusted	Surprise	Total
Clothing	1699	1035	1819	1483	1014	2014	1237	10 301
Electronics	2316	1453	1727	1602	1405	1405	1410	11 318
Hotel	1426	1498	1661	1537	1175	1456	1256	10 009
Movie	1500	1464	1375	1669	1337	1145	1510	10 000

Algorithm 2 MBL Algorithm

Input: T: max iteration;
k: the total number of source domain;
X_{sj}: the source domain labeled data;
X_{tl}: the target domain labeled data;
X_{tu}: the target domain unlabeled data;
Output: emotion classification results $\hat{Y}$

Training:
set $t = 1$;
while $t \leq T$ **do**
  for each source domain labeled data $(X_{s1}, X_{s2}, ..., X_{sk})$ **do**
    obtain DIF based on X_{sj} and X_{tl} by Equ. 7 and Equ. 8;
    calculate KL divergence onto F_{sj} and F_{tl} by Equ. 9;
    obtain DIC based on F_{sj} and F_{tl} by Equ. 10 and Equ. 11;
    train DIF by optimizing Equ. 14;
    train DIC by calculating Equ. 16;
    predict X_{tu} by Equ. 12;
    select p samples X_{tu}^{j} with the highest confidence from X_{tu};
  end for
  remove samples $X_{tu}^{1} \cup X_{tu}^{2} \cup \cdots \cup X_{tu}^{k}$ from X_{tu} and add them to X_{tl};
  $t \leftarrow t + 1$
end while
Testing:
use MBL to predict X_{tu} and obtain $\hat{Y}$;

According to Algorithm 2, the time complexity of MBL is $O(lr^2+rmp+rnq)$. The time complexity of SVM, Bi-GRU, TextCNN, DANN, and UMDA is $O(rl^2)$, $O(lr^2)$, $O(uvlr^2)$, $O(rl^2)$, and $O(rl^2)$, respectively, where l denotes the sequence length, r denotes the representation dimension, v denotes the kernel size of convolution, and u denotes the number of convolution layers.

In BL, m, n, p, and q are set to a small value, respectively. In general, they are smaller than r, and l is far larger than m, n, p, and q. Thus, the time complexity of MBL approximately is $O(lr^2)$ and is smaller than that of SVM, TextCNN, DANN, and UMDA. In addition, the time complexity of Bi-LSTM is close to Bi-GRU.

4. Experiments

4.1. Datasets

Since there is a lack of public datasets for cross-domain emotion classification, we have collected product reviews as the experimental datasets from well-known E-commerce platforms, such as Douban, Ctrip, Jingdong, and Taobao. These datasets contain four domains: movie (M), hotel (H), electronics (E), and clothing (C). There is a total number of 41,628 reviews for the four datasets, and there are seven emotion classes, including happy, moving, angry, sad, fear, disgusted, and surprise. The detailed statistics are shown in Table 1.

According to these data, we constructed 12 cross-domain seven classification tasks. In each domain adaptation task, there are 1000 labeled source domain examples, 1000 unlabeled target domain examples, and 50 labeled target domain examples selected for training data. In addition, to effectively fine-tune the hyper-parameters, 500 target examples are selected as developing data, and the remainder of examples are treated as testing data. Each baseline method and MBL adopt this setting.

4.2. Baseline methods

In this paper, many experiments are carried out to compare our proposed MBL with the following baseline methods: SVM [32], Bi-GRU [33], TextCNN [34], DANN [35], and UMDA [14].

- SVM: It is a non-domain-adaptive method.
- Bi-GRU: It is a non-domain-adaptive method, which is implemented by adopting Bi-GRU.
- TextCNN: It is a convolutional neural network for emotion classification without considering contextual information of a text.
- DANN: It is a domain-adaptive method, which utilizes the domain classifier for minimizing the discrepancy between two domains in an adversarial training manner.
- UMDA: It is an adversarial training method for universal multi-source adaptation.

4.3. Implementation detail

In our experiments, all the word embeddings for texts are initialized to generate 768-dimension vectors. MBL was implemented with the Chinese BERT pre-training model presented by Cui et al. [36], which is composed of 12 transformer blocks and is pre-trained by a large amount of Chinese corpus (including Chinese Wikipedia, news). The nodes of the hidden layer for Bi-LSTM are set to 200. The enhancement nodes of BL are composed of 20 groups, with 50 nodes in each group, and the activation function is tanh.

Model optimization was performed by using the AdamW update strategy [37] with the weight decay set to 0.01 and the initial learning rate set to 1e−5. The corresponding hyperparameters to the best performance for the validation set are obtained by grid search, and the weight of the loss item is set by $\delta = 0.5$, $\eta = 1$, the factor for each iteration of co-training p is set by 5, and the regularization parameter λ is set by 0.001.

4.4. Main results

To verify MBL's effectiveness, we compare it with the baseline methods. We adopt classification accuracy to evaluate these methods. The experimental results for each method are shown in Tables 2 and 3, respectively. The overall comparison for accuracy is shown in Fig. 2. As to Table 2, C+E→H indicates that domains C and E are utilized as source domains, which are transferred to target domain H. As to Table 3, C+E+M→H indicates that domains C, E, and M are utilized as source domains, which are transferred to target domain H.

As to Tables 2 and 3, we evaluate our method over 12 transfer pairs and 4 transfer pairs, respectively, on a total number of 41,628 testing samples. From Tables 2 and 3, it is found that MBL can achieve consistently the best classification performance on accuracy for the benchmark datasets. Compared with SVM, Bi-GRU, TextCNN, DANN, and UMDA, MBL outperforms SVM by 28.18%, 23.33%, 7.76%, 6.41%, and 4.26% on average for two source

Table 2
The accuracy of different methods with two source domains.

Source→Target	SVM	Bi-GRU	TextCNN	DANN	UMDA	MBL-KL	MBL
C+E→H	0.5059	0.5975	0.7896	0.8318	0.842	0.8758	**0.9098**
C+M→H	0.6120	0.6512	0.8307	0.825	0.8871	0.9155	**0.9388**
M+E→H	0.7376	0.7927	0.8762	0.8562	0.8643	0.8764	**0.9247**
M+E→C	0.5790	0.6301	0.8172	0.7944	0.7905	0.8234	**0.8401**
M+H→C	0.5204	0.5844	0.7579	0.7803	0.7831	0.8130	**0.8236**
H+E→C	0.6128	0.6898	0.7611	0.7646	0.7768	0.7692	**0.8019**
H+E→M	0.5543	0.5862	0.7348	0.7539	0.8042	0.8013	**0.8216**
C+E→M	0.5129	0.5490	0.6867	0.7055	0.7378	0.7286	**0.7437**
C+H→M	0.5394	0.5771	0.7890	0.8103	0.8210	0.8273	**0.8570**
H+C→E	0.6197	0.6662	0.8266	0.8413	0.8625	0.8867	**0.9209**
H+M→E	0.6256	0.6556	0.8038	0.8628	0.8876	0.8960	**0.9285**
C+M→E	0.6389	0.6600	0.8354	0.8452	0.8713	0.8849	**0.9289**
Average	0.5882	0.6367	0.7924	0.8059	0.8274	0.8416	**0.8700**

Table 3
The accuracy of different methods with three source domains.

Source→Target	SVM	Bi-GRU	TextCNN	DANN	UMDA	MBL-KL	MBL
C+E+M→H	0.7365	0.8086	0.8668	0.8811	0.8881	0.9163	**0.9392**
C+E+H→M	0.568	0.5992	0.8135	0.8297	0.8348	0.8529	**0.8751**
C+H+M→E	0.6513	0.6853	0.8577	0.8683	0.9086	0.9053	**0.9336**
H+E+M→C	0.6374	0.7104	0.8203	0.8030	0.8054	0.8247	**0.8478**
Average	0.6483	0.7009	0.8396	0.8455	0.8592	0.8748	**0.8989**

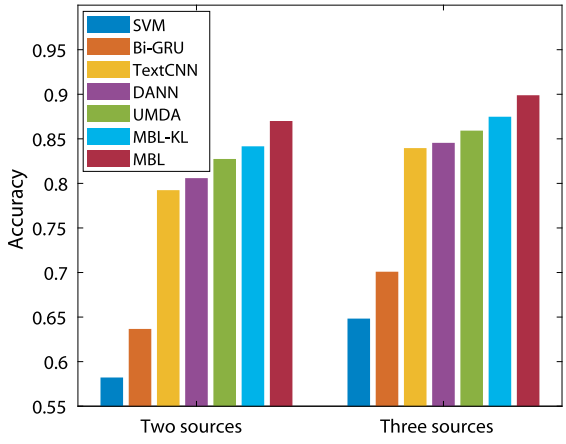

Fig. 2. The overall comparison for the accuracy of different methods.

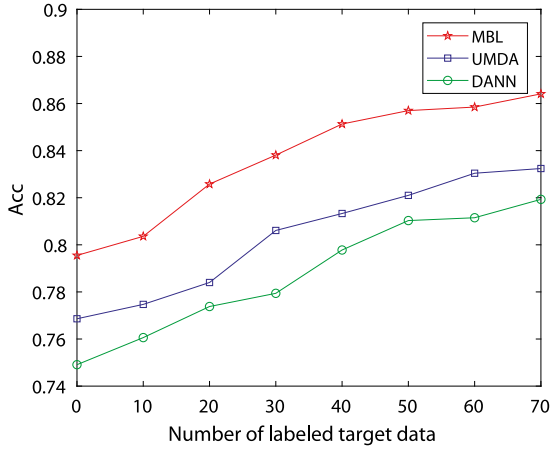

Fig. 3. The effect on accuracy from different amount of labeled target data on task C+H→M.

domains, respectively, and by 25.06%, 19.80%, 5.93%, 5.34%, and 3.97% on average for three source domains, respectively.

In addition, compared with the accuracy on two source domains, the result of three source domains for each baseline method is improved by 6.01%, 6.42%, 4.72%, 3.96%, and 3.18%, respectively, which shows that the more source domains, the better accuracy for cross-domain emotion classification.

To verify the effects of domain adaptation more intuitively, we also conduct an experiment by comparing MBL with a variant without KL metric, named MBL-KL. The average accuracy of MBL-KL is 84.16% and 87.48% for two source domains and three source domains, respectively, which are 2.84% and 2.41% lower than MBL, respectively. The comparison between MBL-KL and MBL, it implies the importance of KL to improve the accuracy of MDA.

4.5. Effect of labeled target data

To effectively analyze the effect on MBL from the number of labeled target data, we conduct the experiment to analyze the effect on accuracy under different amount of labeled target data on tasks C+H→M and C+H+M→E, respectively, by comparing DBL with DANN and UMDA. The specific results are shown in Figs. 3 and 4.

From Figs. 3 and 4, we provide the comparison between baseline methods UMDA, DANN, and MBL under a setting that some labeled target data are randomly selected, and then mixed with the training data, and a similar trend on two kinds of transferring methods from the experimental results can be obtained in other pairs. The effect on accuracy under different amounts of labeled target data on the tasks: C+H→M and C+H+M→E are shown in Figs. 3 and 4, respectively.

From Fig. 4, it is found that the accuracy of MBL, UMDA, and DANN is 88%, 85.3%, and 83%, respectively, when the number of labeled target data is zero. When the number of labeled target data increases, the accuracies of MBL, UMDA, and DANN are improved by 5%, 5.8%, and 4%, respectively, and reached 50. However, the accuracies of MBL, UMDA, and DANN increase slightly, when the number of labeled target data changes from 50 to 70.

In addition, similar phenomena can also be found in Fig. 3. Thus, Figs. 3 and 4, it implies that the difference between these

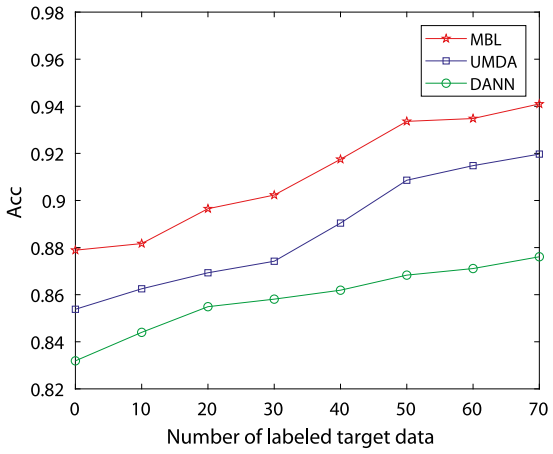

Fig. 4. The effect on accuracy from different amount of labeled target data on task C+H+M→E.

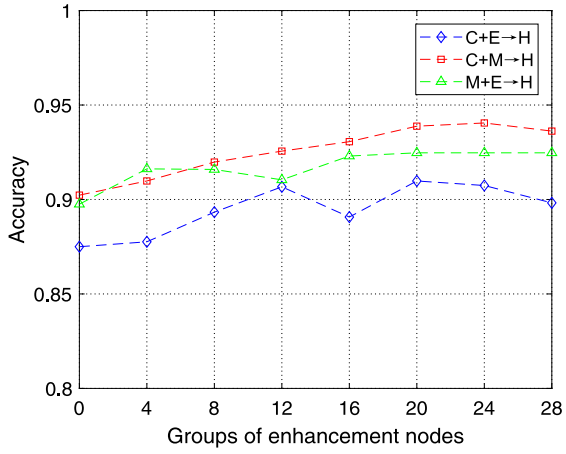

Fig. 5. The effect on accuracy from different groups of enhancement nodes on the tasks C+E→H, C+M→H, and M+E→H.

three methods almost remains unchanged as the amount of labeled target data increases, and MBL can also maintain its dominant position. These trends show that MBL is more effective in the case of less labeled target data, and further benefits from more labeled target data.

4.6. Effect of groups of enhancement nodes

To effectively analyze the effect on MBL from different groups of enhancement nodes, we experiment to analyze the effect on accuracy under different amount of groups of enhancement nodes. We demonstrate the accuracy variation for MDA-based emotion classification under different groups of enhancement nodes. The specific results are shown in Figs. 5, 6, 7, and 8.

From Figs. 5, 6, 7, and 8, it is found that the accuracy increases slightly as to most tasks, as the number of groups of enhancement nodes increases. It means that the enhancement nodes can effectively improve the performance of MBL. For example, as to tasks H+M→E and C+M→E, when the group of enhancement nodes is 20, MBL achieves the best performance. However, the accuracy of MBL begins to decrease when the group of enhancement nodes continues to increase. The main reason is that the ability of each group of enhancement nodes in BL is limited, rather than infinitely increasing with the increase of the group of enhancement nodes. Thus, we set the group of enhancement nodes to 20 in our other experiments.

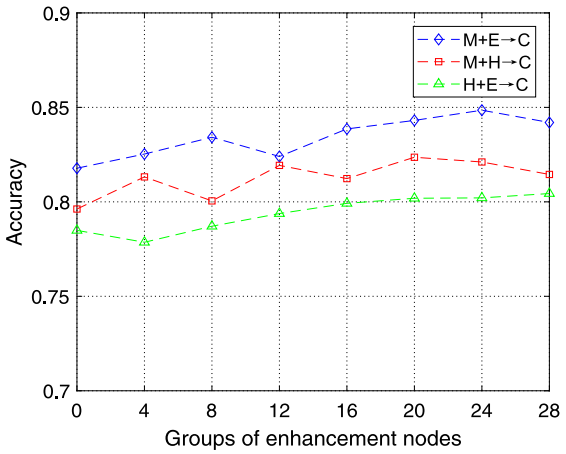

Fig. 6. The effect on accuracy from different groups of enhancement nodes on the tasks M+E→C, M+H→C, and H+E→C.

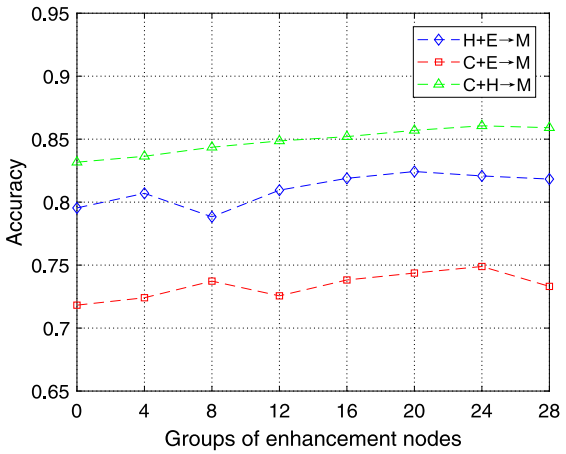

Fig. 7. The effect on accuracy from different groups of enhancement nodes on the tasks H+E→M, C+E→M, and C+H→M.

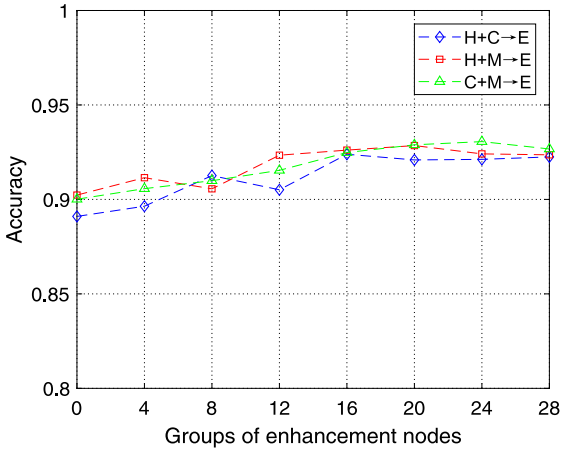

Fig. 8. The effect on accuracy from different groups of enhancement nodes on the tasks H+C→E, H+M→E, and C+M→E.

4.7. Effect of loss function weight

In the final loss function, the hyperparameters η and δ are defined in Eqs. (13) and (14), respectively, which can determine the importance of KL and BL, respectively. We explore the effect

Table 4
Accuracy comparison of BERT+Bi-LSTM+BL with BERT+CNN+BL, BERT+RNN+BL, and BERT+LSTM+BL.

Source→Target	Bi-LSTM	CNN	RNN	LSTM
C+E→H	0.9098	0.9041	0.8956	0.8925
M+H→C	0.8236	0.8255	0.8101	0.8181
C+H+M→E	0.9336	0.9287	0.9264	0.9227
Average	**0.8890**	0.8861	0.8774	0.8778

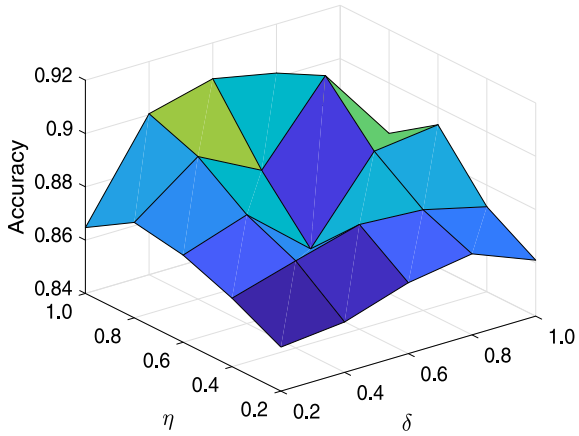

Fig. 9. The effect on accuracy from the parameters η and δ on the task C+E→H.

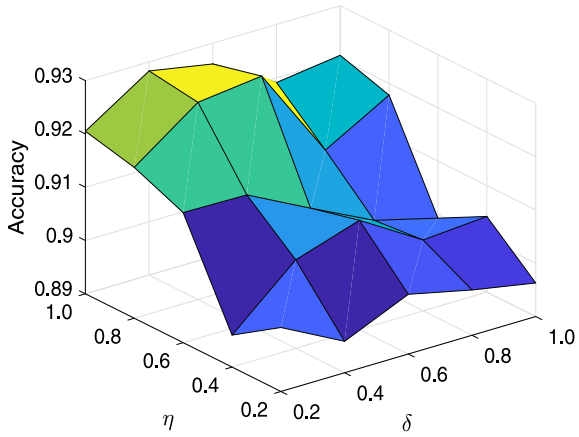

Fig. 10. The effect on accuracy from the parameters η and δ on the task C+M→E.

of η and δ on the overall performance of MBL by changing their values from 0.2 to 1 with a step size of 0.2, respectively. For example, the experimental results on the tasks C+E→H and C+M→E are shown in Figs. 9 and 10, respectively.

From Figs. 9 and 10, it is found that the classification accuracy increases as the value of η increases from 0.9 to 1. The main reason is that large target domain weights can help MBL to learn target domain knowledge more effectively. It is also found that MBL can achieve better classification accuracy when $\eta \in [0.9, 1]$ and $\delta \in [0.4, 0.6]$ for most tasks. Thus, we set $\eta = 1$ and $\delta = 0.5$ in our experiments.

4.8. Superiority of BERT+Bi-LSTM+BL

To verify the superiority of combining BERT and BI-LSTM for extracting features, we experiment by comparing MBL (i.e., BERT+Bi-LSTM+BL) with BERT+CNN+BL, BERT+RNN+BL, and BERT+LSTM+BL, on the tasks C+E→H, M+H→C, and C+M+H →E. The specific results are shown in Table 4.

From Table 4, it is found that the accuracy of BERT+Bi-LSTM+BL outperforms BERT+CNN+BL, BERT+RNN+BL, and BERT+LSTM+BL in most tasks. Thus, we adopt BERT and Bi-LSTM to extract textual features in MBL.

4.9. Discussion

We provide a brief analysis of the limitations of our proposed method MBL.

(1) To validate the framework presented in this article, we have constructed a multi-domain emotion classification dataset in Chinese and conducted experiments on it. Compared with available methods and relevant parameters, MBL can achieve good results on the cross-domain dataset. Due to the lack of an English dataset for multi-domain emotion classification, it fails to verify the performance and adaptation of MBL on the English dataset.

(2) MBL may be regarded as a combining method, feature-based as well as model-based. Though it can utilize the Chinese dataset for multi-domain emotion classification, MBL may depend on specific data and network models heavily. Another aspect is that MBL needs to train a model with numerous data in the target domain and to perform a wider range of NLP tasks. Thus, we need to verify MBL with other NLP tasks in the future.

5. Conclusion and future work

In this article, we explored the practicability of MDA emotion classification by using Bi-LSTM and BL and conducted extensive experiments on four different domain datasets from the public E-commerce platforms. The experimental results showed that our proposed method can reach a performance of 87.00% and 89.89% via two and three source domains, respectively, in terms of accuracy, which is a significant improvement over the baselines. Compared with most of the previous methods, our proposed method MBL can effectively improve performance by using BERT, KL metric, and Bi-LSTM to extract DIFs, and the combination of Bi-LSTM and BL could be beneficially employed in the MDA task. As to our future work, we plan to combine other deep learning models and BL for MDA tasks, and to classify multi-modal cross-domain emotion. We hope this work can potentially provide some new insights and perspectives for research on MDA.

CRediT authorship contribution statement

Sancheng Peng: Methodology, Funding acquisition, Writing – original draft. **Rong Zeng:** Conceptualization, Software, Writing – original draft. **Lihong Cao:** Data curation, Investigation. **Aimin Yang:** Supervision. **Jianwei Niu:** Methodology. **Chengqing Zong:** Writing – reviewing. **Guodong Zhou:** Writing – reviewing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

This work is partially supported by the National Natural Science Foundation of China under Grant No. 61876205, and the Ministry of Education of Humanities and Social Science project under Grant No. 20YJAZH118.

References

[1] S. Peng, L. Cao, Y. Zhou, Z. Ouyang, A. Yang, X. Li, W. Jia, S. Yu, A survey on deep learning for textual emotion analysis in social networks, Digit. Commun. Netw. 8 (5) (2022) 745–762.
[2] C. Chen, W. Xie, Y. Wen, Y. Huang, X. Ding, Multiple-source domain adaptation with generative adversarial nets, Knowl.-Based Syst. 199 (2020) 105962.
[3] M. Yang, Y. Shen, X. Chen, C. Li, Multi-source domain adaptation for sentiment classification with granger causal inference, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2020, pp. 1913–1916.
[4] J. Guo, D.J. Shah, R. Barzilay, Multi-source domain adaptation with mixture of experts, in: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 2018, pp. 4694–4703.
[5] F.H. Khan, U. Qamar, S. Bashir, Enhanced cross-domain sentiment classification utilizing a multi-source transfer learning approach, Soft Comput. 23 (14) (2019) 5431–5442.
[6] C. Lin, S. Zhao, L. Meng, T. Chua, Multi-source domain adaptation for visual sentiment classification, in: Proceeding of the Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI 2020), 2020, pp. 2661–2668.
[7] I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A.C. Courville, Y. Bengio, Generative adversarial nets, in: Generative Adversarial Nets. Proceedings of the 27th International Conference on Neural Information Processing Systems, 2014, pp. 2672–2680.
[8] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186.
[9] M. Schuster, K.K. Paliwal, Bidirectional recurrent neural networks, IEEE Trans. Signal Process. 45 (11) (1997) 2673–2681.
[10] C.P. Chen, Z. Liu, Broad learning system: An effective and efficient incremental learning system without the need for deep architecture, IEEE Trans. Neural Netw. Learn. Syst. 29 (1) (2018) 10–24.
[11] Y. Li, M. Murias, S. Major, G. Dawson, D.E. Carlson, Extracting relationships by multi-domain matching, in: Proceedings of the 32nd International Conference on Neural Information Processing Systems, 2018, pp. 6799–6810.
[12] J. Hoffman, M. Mohri, N. Zhang, Algorithms and theory for multiple-source adaptation, in: Proceedings of the 32nd International Conference on Neural Information Processing Systems, 2018, pp. 8256–8266.
[13] D. Wright, I. Augenstein, Transformer based multi-source domain adaptation, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2020, pp. 7963–7974.
[14] Y. Yin, Z. Yang, H. Hu, X. Wu, Universal multi-Source domain adaptation for image classification, Pattern Recognit. 121 (2022) 108238.
[15] D. Wang, C. Lu, J. Wu, H. Liu, W. Zhang, F. Zhuang, H. Zhang, Softly associative transfer learning for cross-domain classification, IEEE Trans. Cybern. 50 (11) (2020) 4709–4721.
[16] K. Zhang, H. Zhang, Q. Liu, H. Zhao, H. Zhu, E. Chen, Interactive attention transfer network for cross-domain sentiment classification, in: Proceedings of the 33rd AAAI Conference on Artificial Intelligence, 2019, pp. 5773–5780.
[17] Z. Li, Y. Wei, Y. Zhang, Q. Yang, Hierarchical attention transfer network for cross-domain sentiment classification, in: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, 2018, pp. 5852–5859.
[18] C. Du, H. Sun, J. Wang, Q. Qi, J. Liao, Adversarial and domain-aware BERT for cross-domain sentiment analysis, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 4019–4028.
[19] Y. Du, M. He, L. Wang, H. Zhang, Wasserstein based transfer network for cross-domain sentiment classification, Knowl.-Based Syst. 204 (2020) 106162.
[20] J. Zhou, J. Tian, R. Wang, Y. Wu, W. Xiao, L. He, Sentix: A sentiment-aware pre-trained model for cross-domain sentiment analysis, in: Proceedings of the 28th International Conference on Computational Linguistics, 2020, pp. 568–579.
[21] M. Peng, Q. Zhang, Weighed domain-invariant representation learning for cross-domain sentiment analysis, in: Proceedings of the 28th International Conference on Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 251–265.
[22] C. Zhao, S. Wang, D. Li, Multi-source domain adaptation with joint learning for cross-domain sentiment classification, Knowl.-Based Syst. 191 (2020) 105254.
[23] Z. Xu, L. von Ritter, G. Serra, Hierarchical adversarial training for multi-domain adaptive sentiment analysis, in: Complex Pattern Mining, Vol. 880, 2020, pp. 17–32.
[24] Y. Dai, J. Liu, X. Ren, Z. Xu, Adversarial training based multi-source unsupervised domain adaptation for sentiment analysis, in: Proceedings of Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI 2020), 2020, pp. 7618–7625.
[25] S. Zhao, Y. Xiao, J. Guo, X. Yue, J. Yang, R. Krishna, P. Xu, K. Keutzer, Curriculum CycleGAN for textual sentiment domain adaptation with multiple sources, in: Proceedings of the International World Wide Web Conference (WWW 2021), Ljubljana, Slovenia, 2021, pp. 541–552.
[26] Y. Dai, J. Liu, J. Zhang, H. Fu, Z. Xu, Unsupervised sentiment analysis by transferring multi-source knowledge, Cogn. Comput. 13 (2021) 1185–1197.
[27] Y. Fu, Y. Liu, Contrastive transformer based domain adaptation for multi-source cross-domain sentiment classification, Knowl.-Based Syst. 245 (2022) 108649.
[28] S. Peng, R. Zeng, H. Liu, G. Chen, R. Wu, A. Yang, S. Yu, Emotion classification of text based on BERT and broad learning system, in: Proceeding of the Asia Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint International Conference on Web and Big Data (APWeb-WAIM), Vol. 12858, 2021, pp. 382–396.
[29] G. Chen, S. Peng, R. Zeng, Z. Hu, L. Cao, Y. Zhou, Z. Ouyang, X. Nie, P-norm broad learning for negative emotion classification in social networks, Big Data Min. Anal. 5 (3) (2022) 245–256.
[30] R. Zeng, H. Liu, S. Peng, L. Cao, A. Yang, C. Zong, G. Zhou, CNN-based broad learning for cross-domain emotion classification, Tsinghua Sci. Technol. 28 (2) (2023) 360–369.
[31] Q. Zhu, W. Bi, X. Liu, X. Ma, X. Li, D. Wu, A batch normalized inference network keeps the KL vanishing away, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 2636–2649.
[32] R. Fan, K. Chang, C. Hsieh, X. Wang, C. Lin, LIBLINEAR: A library for large linear classification, J. Mach. Learn. Res. 9 (2008) 1871–1874.
[33] K. Cho, B. van Merrienboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder-decoder for statistical machine translation, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014), Doha, Qatar, 2014, pp. 1724–1734.
[34] Y. Kim, Convolutional neural networks for sentence classification, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar, 2014, pp. 1746–1751.
[35] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, V. Lempitsky, Domain-adversarial training of neural networks, J. Mach. Learn. Res. 17 (1) (2016) 2030–2096.
[36] Y. Cui, W. Che, T. Liu, B. Qin, Z. Yang, Pre-training with whole word masking for Chinese BERT, IEEE/ACM Trans. Audio Speech Lang. Process. (2021) 1–8.
[37] I. Loshchilov, F. Hutter, Fixing weight decay regularization in adam, in: Proceedings of International Conference on Learning Representations, 2018.