IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 21, 2020, accepted March 27, 2020, date of publication March 31, 2020, date of current version April 16, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2984571

Multi-Group Transfer Learning on Multiple Latent Spaces for Text Classification

JIANHAN PAN[1,2], TENG CUI[1], THUC DUY LE[2], XIAOMEI LI[2], AND JING ZHANG[3,4], (Senior Member, IEEE)

[1]School of Computer Science and Technology, Jiangsu Normal University, Xuzhou 221116, China
[2]School of Information Technology and Mathematical Sciences, University of South Australia, Adelaide, SA 5000, Australia
[3]School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China
[4]Department of Electrical and Computer Engineering, University of Pittsburgh, Pittsburgh, PA 15260, USA

Corresponding author: Teng Cui (cuicui@jsnu.edu.cn)

This work was supported by grants from the National Natural Science Foundation of China (No. 61703187 and 91846104).

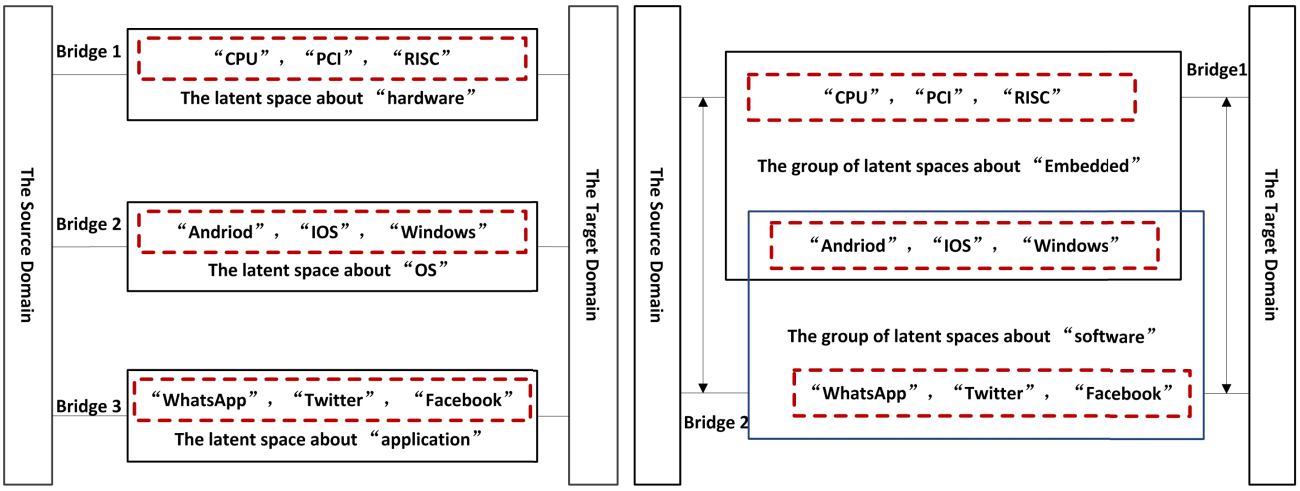
ABSTRACT Transfer learning aims to leverage valuable information in one domain to promote the learning tasks in the other domain. Some recent studies indicated that the latent information, which has a close relationship with the high-level concepts, are more suitable for cross-domain text classification than learning raw features. To obtain more latent information existing in the latent feature space, some previous methods constructed multiple latent feature spaces. However, those methods ignored that the latent information of different latent spaces may lack the relevance for promoting the adaptability of transfer learning models, even may lead to negative knowledge transfer when there exists a glaring discrepancy among the different latent spaces. Additionally, since those methods learn the latent space distributions using a strategy of direct-promotion, their computational complexity increases exponentially as the number of latent spaces increases. To tackle this challenge, this paper proposes a Multiple Groups Transfer Learning (MGTL) method. MGTL first constructs multiple different latent feature spaces and then integrates the adjacent ones that have a similar latent feature dimension into one latent space group. Along this way, multiple latent space groups can be obtained. To enhance the relevance among these latent space groups, MGTL makes the adjacent groups contain one same latent space at least. Then, different groups will have more relevance than raw latent spaces. Second, MGTL utilizes an indirect-promotion strategy to connect different latent space groups. The computational complexity of MGTL increases linearly as the number of latent space groups increases and is superior to those multiple latent space methods based on direct-promotion. In addition, an iterative algorithm is proposed to solve the optimization problem. Finally, a set of systematic experiments demonstrate that MGTL outperforms all the compared existing methods.

INDEX TERMS Transfer learning, non-negative matrix tri-factorization, multi-group, cross-domain classification.

I. INTRODUCTION

Traditional classification algorithms can achieve satisfying performance since they have a common assumption that both training and test data come from the same distribution. However, this assumption cannot hold in many practical applications. To tackle the challenge, many transfer learning methods have been proposed recently [3]–[7], [11]–[15], [23]–[26], [28], [30]–[32], [36]. Transfer learning is designed to model a better classifier using examples with tags in the source domain to predict the categories of test instances with fewer or without tags in the target domain. Some previous studies have shown that the latent information, which has a close relationship with the high-level concepts, is more suitable for cross-domain text classification than learning raw features [7]. To obtain more latent information that exists in the latent feature space, some previous methods such as MBTL [33] and MLTL [35] constructed multiple latent feature spaces and then learn the corresponding distribution on each latent space. We represent such methods as the multiple latent spaces transfer learning. The limitation of these approaches is two-fold:

The associate editor coordinating the review of this manuscript and approving it for publication was Mohamad Forouzanfar.



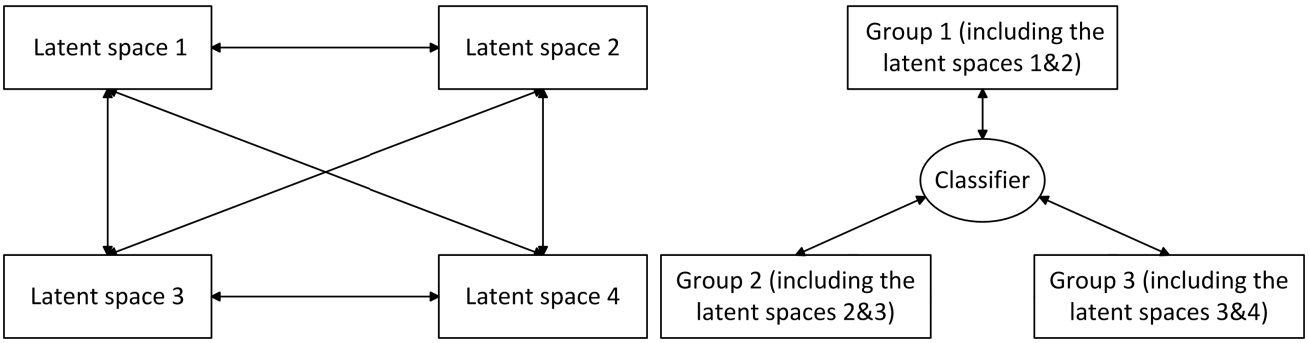
(a) multi-latent spaces (b) multi-latent space groups

FIGURE 1. **The relevance comparison between the traditional multi-latent spaces methods and MGTL.**

First, it can be empirically considered that the correlation of different latent spaces is inversely proportional to their differences. With the increase of the number of latent spaces, the average dimensional difference among latent spaces will increase, and the corresponding average relevance among them will decrease accordingly. Although more latent information can be obtained as the number of latent spaces increase, this latent information of different latent spaces may lack the relevance for promoting the adaptability of transfer learning models, even may lead to negative knowledge transfer when there exists a glaring discrepancy among the different latent spaces. For example, although words like ''CPU'', ''PCI'', and ''RISC'', which are drawn from the latent space about *hardware*, as well as ''Twitter'', ''WhatsApp'' and ''Facebook'', which are drawn from the latent space about *application*, are all related to *computer*, the relevance between *hardware* and *application* is insufficient. In Figure 1, we can find that although multiple shared bridges can be built on these different latent spaces respectively, they can not effectively promote each other to establish a more adaptive structure for knowledge transfer across domains.

Second, to construct a more effective structure across domains, those transfer learning methods based on multiple latent spaces usually learn the latent space distributions by using the strategy of direct-promotion [5]–[7], [33]–[35]. The key idea of this strategy is that learning the distributions on one latent feature space can directly promote the distribution learning on the others. Therefore, the computational complexity of those algorithms increases exponentially as the number of latent spaces increases. In Figure 2, we can find that learning of each distribution is dependent on the learning of the others. With the increasing of latent spaces, the computational complexity will be intolerable.

In this paper, we propose Multi-Group Transfer Learning (MGTL) based on non-negative matrix tri-factorization (NMTF) techniques, which groups multiple latent feature spaces and learns the corresponding distributions in the different latent space groups simultaneously. The key idea of MGTL is as follows: First, to obtain more latent information that can be used to learn the shared structure across domains, MGTL constructs multiple different latent feature spaces and then integrates the adjacent latent spaces that have the similar latent feature dimension into one latent space group. Along this way, multiple latent spaces groups can be obtained. To enhance the relevance among these latent space groups, MGTL makes the adjacent latent space groups contain one same latent space at least. For example, as shown in Figure 1, three different latent spaces, which can be indicated to *hardware*, *OS*, and *application*, respectively, are integrated into two groups of latent spaces overlapping partially. These two latent space groups can be indicated to *embedded* and *software*, respectively. Obviously, *embedded* and *software* exhibit more relevance than the three raw latent spaces do for establishing a more adaptive structure for knowledge transfer across domains. Second, to decrease the computational complexity of learning multiple latent space groups, MGTL utilizes an indirect-promotion strategy to connect different latent space groups [35]. Specifically, it exploits the label information in the source domains and the latent shared information on one latent space group to learn the corresponding distributions. Then, a shared classification model can be obtained to promote learning distributions on the other latent space groups. In other words, learning the distributions and modeling the classifier can promote each other directly. Therefore, learning distributions on different latent space groups can facilitate each other indirectly by the classification model that is shared on different latent space groups. For example, as Figure 2 shows, four latent spaces are integrated into three latent space groups, and the corresponding number of connections (the direct promotion



(a) direct-promotion strategy (b) indirect-promotion strategy

FIGURE 2. The computational complexity comparison between the direct-promotion and indirect-promotion strategy.

is represented as the connection line among different spaces or groups) is reduced from 8 to 3. Obviously, MGTL has a linear increase in the computational complexity as the number of latent space groups increases, which is superior to the traditional transfer learning methods based on multiple latent spaces.

The main contributions of this paper are three-fold:

1) Motivated by a significant observation that different latent spaces may not promote each other effectively to build a shared bridge, we propose a novel method MGTL which can construct multiple relevant groups for knowledge transfer.
2) To solve the optimization of MGTL, we present a non-negative matrix tri-factorization based iterative algorithm and utilize an indirect-promotion strategy to decrease the computational complexity.
3) In addition, we conducted extensive experiments, which demonstrates that the proposed MGTL is superior to the state-of-the-art transfer learning methods.

The remainder of this paper is organized as follows: Section 2 briefly review the related studies. Section 3 presents some preliminary knowledge. Section 4 presents the proposed MGTL model. Section 5 shows the experiments and discusses the experimental results. Finally, Section 6 concludes the paper.

II. RELATED WORK

According to the homogeneity of feature spaces, transfer learning approaches can be categorized into the homogeneous ones and the heterogeneous ones. Both of these two kinds of transfer learning methods are widely used in real-world applications, such as image classification [12], [30]–[32], computational biology [11], [13], [14], [28], and text classification [7], [15], [23]–[26], [36]. For the heterogeneous approaches, which can construct a shared bridge on different feature spaces, the key idea is to map different feature spaces to a same latent one. Reference [37] developed a semi-supervised approach to match the examples and preserve the semantic consistency between heterogeneous domains. Reference [38] proposed a novel HTL approach (Deep-MCA) based on a structure with adversarial kernel training to obtain an end-to-end solution. Reference [39] proposed a new TDML framework for heterogeneous tasks, which learns the metric in the target domain by extracting the knowledge fragments from the source domain. Reference [40] developed a novel framework (HHTL) and two architectures to transfer knowledge across heterogeneous domains via the feature transformation cross domains. For the homogeneous approaches, which can construct a shared bridge on the same feature spaces, the key idea is to learn a consistent distribution on these feature spaces. In this paper, we focus on the homogeneous transfer learning tasks.

According to the literature survey [1], our method is more closely related to the feature representation-based methods, which can be further divided into feature selection-based ones and feature mapping based-ones. Then, we will first review these approaches in brief. Dai *et al.* [3] developed a co-clustering based approach to identify feature clusters of different domains, by spreading class information from one domain to another. Jiang and Zhai [21] proposed a two-step framework to transfer knowledge across domains. The first step is to generalize features, and the second one is to select the specific features in the target domains for domain adaptation. Uguroglu and Carbonell [22] proposed a new approach to distinguish variant and invariant original features among datasets for knowledge transfer and transformed a distribution problem to a convex optimization one. Blitzer *et al.* [2] proposed a feature correspondence based method by using unlabeled data and pivots raw features from different domains for knowledge transfer. Zhuang *et al.* [4] proposed a method for cross-domain learning using the association between feature clustering and example clustering. Reference [17] proposed a domain adaptation method TCL, which leverages both the common original features to construct a shared bridge, and uses the specific ones to discriminate domains. Additionally, Pan *et al.* [34] proposed QTL to integrates all kinds of high-level concepts for fitting different distributions.

Our work belongs to the feature representation-based approaches in which some methods utilize a multiple latent spaces strategy. The key idea of these methods is to obtain

more latent information by constructing and learning multiple latent spaces [33], [35]. From the view of multiple latent spaces, Hu *et al.* [33] developed a multi-bridge approach (MBTL) to build multiple shared bridges to transfer knowledge. Pan *et al.* [35] presented an expanded version of MBTL, which constructs one common space and two specific spaces as one latent space layer. Then, along this line, multiple layers are built and used to learn the corresponding distributions simultaneously. However, these multi-latent transfer learning approaches ignored that the latent information of different latent spaces may lack the relevance that can promote the adaptability of the transfer learning model and even may lead to negative knowledge transfer when there exists a glaring discrepancy among the different latent spaces. Moreover, since these methods learn the latent space distributions using a strategy of direct-promotion, the computational complexity of these algorithms increases exponentially as the number of latent spaces increases. To tackle these problems, we propose the Multi-Group Transfer Learning (MGTL) method.

III. PRELIMINARY KNOWLEDGE

In this section, we first list the mathematical notations used in this paper, then briefly introduce the high-level concepts and non-negative matrix tri-factorization (NMTF) model.

A. NOTATIONS

We use an uppercase letter (such as X and Y) to represent a matrix, and denote the element at the i-th row and j-th column of matrix X as $X_{[i,j]}$. The sets of real numbers and non-negative real numbers are denoted by R and R_+, respectively. Let $\mathcal{D} = (\mathcal{D}_1, \cdots, \mathcal{D}_s, \mathcal{D}_{s+1}, \cdots, \mathcal{D}_{s+t})$ be a set of domains, including s source domains and t target ones. $X_r = [X^r_*], \ldots, X^r_{*n_r}]$ indicate the feature-instance matrix of domain $\mathcal{D}_r$, $(1 \le r \le s+t)$. Labels of the instances in source domain $\mathcal{D}_r$ are given as Y_r, $(1 \le r \le s)$. The frequently-used notations in MGTL are summarized in Table 1.

TABLE 1. Notations and descriptions.

Notations	Descriptions
$\mathcal{D}_r$	Domain r
g	Index of a latent space group, $1 \le g \le NOG$
$G_{(g)}$	Latent space group g
NOG	Number of latent space groups
e	Index of a latent space $1 \le e \le NOL$
l_e	Latent space e
NOL	Number of latent spaces
NOO	Number of overlapped latent spaces
NPG	Number of latent spaces per group
T	Transposition of matrix
k	Dimension of latent feature space

B. HIGH-LEVEL CONCEPTS AND NMTF

Since MGTL utilizes the high-level concepts, which have a close relationship with the latent information, to construct the shared structure across domains, we will introduce the high-level concepts at first. Specifically, the high-level concepts consider two sides of a concept, namely concept *extension* and *intension*. The association between original features and the high-level concepts is represented as the Concept Extension (CE), and the association between these high-level concepts and the example classes is represented as the Concept Intension (CI) [6], [7]. We list two kinds of high-level concepts used in this paper in Table 2.

TABLE 2. High-level concepts learned in the latent feature spaces.

Notation	Description
Identical concepts	An high-level concept which has the same CE and the same CI in different domains.
Synonymous concepts	An high-level concept which has the different CE and the same CI in different domains.

In addition, as mentioned above, we utilize NMTF, which is widely used for text classification [3]–[7], [13], [14], to implement the proposed MGTL. The key formula of NMTF is as follows:

$$X_{m\times n} = U_{m\times k} H_{k\times c} V^T_{n\times c} \tag{1}$$

where $X \in R^{m\times n}$, $U \in R^{m\times k}$, $H \in R^{k\times c}$ and $V \in R^{n\times c}$ represent the feature-instance, feature-concept, concept-class, and instance-class matrices, respectively. Here, m, n, k, and c represent the numbers of original features, instances, high-level concepts, and instance classes, respectively. Additionally, $U \in R^{m\times k}$ and $H \in R^{k\times c}$ also represent concept CE and CI, respectively, and $V \in R^{n\times c}$ can be used as a classifier. Actually, the multiplication of these matrices forms a mapping from one dimension to another.

Additionally, NMTF is an optimization problem as follows:

$$\begin{aligned} &\min_{U,H,V\ge 0} \| X - UHV^T \|^2 \\ &s.t. \sum_{i=1}^{m} U_{[i,j]} = 1, \quad \sum_{j=1}^{c} V_{[i,j]} = 1 \end{aligned} \tag{2}$$

To deal with a transfer learning problem, NMTF is developed to adapt to different domains. The above formula can be rewritten as follows:

$$\begin{aligned} &\min_{U_rH,V_r\ge 0} \sum_{r=1}^{s+t} ||X_r - U_r H V_r^T||^2 \\ &s.t. \sum_{i=1}^{m} U_{r[i,j]} = 1, \quad \sum_{j=1}^{c} V_{r[i,j]} = 1 \end{aligned} \tag{3}$$

where s and t represent the number of source and target domains, respectively, and r represents the index of a domain.

IV. MULTI-GROUP TRANSFER LEARNING

In this section, we present our MGTL method for cross-domain classification. Meanwhile, we formulate MGTL as an optimization problem and propose an iterative algorithm to solve it.

A. PROBLEM DEFINITION

Since MGTL groups multiple latent feature spaces, the high-level concepts should be learned in the corresponding

latent space groups, respectively. Then, the concept extension U and the concept intension H can be represented as $U_{m\times k} = [U^{G(1)}_{m\times k^{G(1)}}, \ldots, U^{G(g)}_{m\times k^{G(g)}}, \ldots, U^{G(NOG)}_{m\times k^{G(NOG)}}]$ and $H_{k\times c} = [H^{G(1)}_{k^{G(1)}\times c}; \ldots; H^{G(g)}_{k^{G(g)}\times c}; \ldots; H^{G(NOG)}_{k^{G(NOG)}\times c}]$, $(k^{G(1)} + \ldots + k^{G(g)} + \ldots + k^{G(NOG)} = k)$, respectively, where $U^{G(g)}_{m\times k^{G(g)}}$ and $H^{G(g)}_{k^{G(g)}\times c}$ represent the CE and the CI in latent space group $G(g)$, respectively. In addition, since MGTL learns two kinds of shared high-level concepts including the identical concept and the synonymous concept together, we divide $U^{G(g)}_{m\times k^{G(g)}}$ and $H^{G(g)}_{k^{G(g)}\times c}$ into two parts. That is, $U^{G(g)}_{m\times k^{G(g)}} = [U^{1G(g)}_{m\times k^{G(g)}_1}, U^{2G(g)}_{m\times k^{G(g)}_2}]$, where $U^{1G(g)}_{m\times k^{G(g)}_1}$ and $U^{2G(g)}_{m\times k^{G(g)}_2}$ represent the CEs of identical concepts and synonymous concepts, respectively. Accordingly, $H^{G(g)}_{k^{G(g)}\times c}$ can be represented as $H^{G(g)}_{k^{G(g)}\times c} = \begin{bmatrix} H^{1G(g)}_{k^{G(g)}_1\times c} \\ H^{2G(g)}_{k^{G(g)}_2\times c} \end{bmatrix}$, where $H^{1G(g)}_{k^{G(g)}_1\times c}$ and $H^{2G(g)}_{k^{G(g)}_2\times c}$ represent the CIs of identical concepts and synonymous concepts, respectively. For all above equations, we have $k^{G(g)}_1 + k^{G(g)}_2 = k^{G(g)}$.

Therefore, in the latent feature space group $G(g)$, Eq. (1) can be rewritten as:

$$\begin{aligned} X_{m\times n} &= U^{G(g)}_{m\times(k^{G(g)})} H^{G(g)}_{(k^{G(g)})\times c} V^T_{n\times c} \\ &= [U^{1G(g)}_{m\times k^{G(g)}_1}, U^{2G(g)}_{m\times k^{G(g)}_2}] \begin{bmatrix} H^{1G(g)}_{k^{G(g)}_1\times c} \\ H^{2G(g)}_{k^{G(g)}_2\times c} \end{bmatrix} V^T_{n\times c} \end{aligned} \tag{4}$$

Then, the objective function can be formulated as follows:

$$\mathcal{L} = \sum_{g=1}^{NOG}\sum_{r=1}^{s+t} \| X_r - U^{G(g)}_r H^{G(g)}_r V^T_r \|^2 \tag{5}$$

where $X_r \in R^{m\times n}_+$, $U^{G(g)}_r \in R^{m\times k^{G(g)}}_+$, $H^{G(g)}_r \in R^{k^{G(g)}\times c}_+$ and $V^T_r \in R^{n^r\times c}_+$.

As described in previous sections, we divide the CE and CI into two parts on latent space group $G(g)$, respectively. Therefore, Eq. (5) can be rewritten as follows:

$$\mathcal{L} = \sum_{g=1}^{NOG}\sum_{r=1}^{s+t} \| X_r - \left[U^{1G(g)}, U^{2G(g)}_r\right] \begin{bmatrix} H^{1G(g)} \\ H^{2G(g)} \end{bmatrix} V^T_r \|^2 \tag{6}$$

For Eq. (6), we add the constraint condition to CE, CI, and V_r simultaneously to quantify the relevance among high-level concepts, raw features and instance classes. Then, the optimization problem is deduced as follows:

$$\begin{aligned} &\min_{U^{G(g)}_r, H^{G(g)}_r, V_r} \mathcal{L} \\ &s.t. \sum_{j=1}^{k^{G(g)}_1} U^{1G(g)}_{[i,j]} = 1, \sum_{j=1}^{k^{G(g)}_2} U^{2G(g)}_{r[i,j]} = 1, \sum_{j=1}^{c} H^{1G(g)}_{[i,j]} = 1, \\ &\sum_{j=1}^{c} H^{2G(g)}_{[i,j]} = 1, \sum_{j=1}^{c} V_{r[i,j]} = 1. \end{aligned} \tag{7}$$

Here, the constraint conditions for CE, CI, and V_r represent the high-level concepts distribution of original features, the class distribution of high-level concepts, and the class distribution of examples, respectively.

B. SOLUTION OF MGTL

We first elaborate on the objective function and deduce the corresponding update formulas, then propose an iterative algorithm. According to the attributes of the Frobenius norm and the trace of matrices, the objective function can be formulated as follows:

$$\begin{aligned} \mathcal{L} = \sum_{g=1}^{NOG}\sum_{r=1}^{s+t} tr\Big(& X_r{}^T X_r - 2\cdot X_r{}^T A^{G(g)}_r - 2\cdot X_r{}^T B^{G(g)}_r \\ & + A^{G(g)T}_r A^{G(g)}_r + B^{G(g)T}_r B^{G(g)}_r + 2\cdot A^{G(g)T}_r B^{G(g)}_r \Big) \end{aligned} \tag{8}$$

where $A^{G(g)}_r = U^{1G(g)} H^{1G(g)} V^T_r$ and $B^{G(g)}_r = U^{2G(g)}_r H^{2G(g)} V^T_r$. Then, the corresponding variables are updated as follows:

$$U^{1G(g)}_{[i,j]} \leftarrow U^{1G(g)}_{[i,j]} \cdot \sqrt{\frac{[\sum_{r=1}^{s+t} X_r V_r H^{1G(g)T}]_{[i,j]}}{[\sum_{r=1}^{s+t}(A^{G(g)}_r V_r H^{1G(g)T} + B^{G(g)}_r V_r H^{1G(g)T})]_{[i,j]}}} \tag{9}$$

$$U^{2G(g)}_{r[i,j]} \leftarrow U^{2G(g)}_{r[i,j]} \cdot \sqrt{\frac{[X_r V_r H^{2G(g)T}]_{[i,j]}}{[A^{G(g)}_r V_r H^{2G(g)T} + B^{G(g)}_r V_r H^{2G(g)T}]_{[i,j]}}} \tag{10}$$

$$H^{1G(g)}_{[i,j]} \leftarrow H^{1G(g)}_{[i,j]} \cdot \sqrt{\frac{[\sum_{r=1}^{s+t} U^{1G(g)T} X_r V_r]_{[i,j]}}{[\sum_{r=1}^{s+t}(U^{1G(g)T} A^{G(g)}_r V_r + U^{1G(g)T} B^{G(g)}_r V_r)]_{[i,j]}}} \tag{11}$$

$$H^{2G(g)}_{[i,j]} \leftarrow H^{2G(g)}_{[i,j]} \cdot \sqrt{\frac{[\sum_{r=1}^{s+t} U^{2G(g)T}_r X_r V_r]_{[i,j]}}{[\sum_{r=1}^{s+t}(U^{2G(g)T}_r A^{G(g)}_r V_r + U^{2G(g)T}_r B^{G(g)}_r V_r)]_{[i,j]}}} \tag{12}$$

$$V_{r[i,j]} \leftarrow V_{r[i,j]} \cdot \sqrt{[\sum_{g=1}^{NOG} Mn_{G(g)}]_{[i,j]} / [\sum_{g=1}^{NOG} Md_{G(g)}]_{[i,j]}} \tag{13}$$

where $Mn_{G(g)} = X_r{}^T U^{G(g)}_r H^{G(g)}_r$ and $Md_{G(g)} = V_r H^{G(g)T}_r U^{G(g)T}_r U^{G(g)}_r H^{G(g)}_r$. In each iteration, we calculate all the variables according to the updating rules and use Eq. (14) to normalize $U^{1G(g)}$, $U^{2G(g)}_r$, $H^{1G(g)}$, $H^{2G(g)}$, and V_r as follows:

$$U^{1G(g)}_{[i,j]} \leftarrow \frac{U^{1G(g)}_{[i,j]}}{\sum_{j=1}^{k^{G(g)}_1} U^{1G(g)}_{[i,j]}}, H^{1G(g)}_{[i,j]} \leftarrow \frac{H^{1G(g)}_{[i,j]}}{\sum_{j=1}^{c} H^{1G(g)}_{[i,j]}},$$

$$U^{2G(g)}_{r[i,j]} \leftarrow \frac{U^{2G(g)}_{r[i,j]}}{\sum_{j=1}^{k_2^{G(g)}} U^{2G(g)}_{r[i,j]}}, H^{2G(g)}_{[i,j]} \leftarrow \frac{H^{2G(g)}_{[i,j]}}{\sum_{j=1}^{c} H^{2G(g)}_{[i,j]}},$$
$$V_{r[i,j]} \leftarrow \frac{V_{r[i,j]}}{\sum_{j=1}^{c} V_{r[i,j]}} \tag{14}$$

Based on Eqs. (9)-(14), an iterative algorithm is proposed and described in Algorithm 1. We normalize the data matrices such that ${X_r}^T 1_m = 1_n$. The CE of the high-level concepts are initialized with the matrices obtained by implemented PLSA [9]. For instance, we set the numbers of the shared concept in the latent space group $G(g)$ as $(k_1^{G(g)}+k_2^{G(g)})$. Then, we obtain feature information $W \in R_+^{m\times(k_1^{G(g)}+k_2^{G(g)})}$ through conducting PLSA on the data from the source to the target domains. W is divided into two parts $W = [W_1, W_2](W_1 \in R_+^{m\times k_1^{G(g)}}$ and $W_2 \in R_+^{m\times k_2^{G(g)}})$. Finally, $U^{1G(g)}$ is initialized as W_1 and $U_r^{2G(g)}(1 \le r \le s{+}t)$ is initialized as W_2, respectively.

Algorithm 1 Multi-Group Transfer Learning

Input: $\{X_r\}_{r=1}^{s+t}$, $\{V_r\}_{r=1}^{s}$, parameters $k_1^{G(g)}$, $k_2^{G(g)}$, and the number of iterations *maxIter*.
Output: $U^{1G(g)}$, $U_r^{2G(g)}(1 \le r \le s{+}t)$, $H^{1G(g)}$, $H^{2G(g)}$, and $V_r(1{+}s \le r \le s{+}t)$.

```
Normalize the data matrices by
X_{r[i,j]} ← X_{r[i,j]} / Σ_{i=1}^{m} X_{r[i,j]}, (1 ≤ r ≤ s+t);
U^{1G(g)(0)} and U_r^{2G(g)(0)} are initialized according to
Section 4.2, and V_r^{(0)} is initialized by Logistic regression;
for k ← 1 to maxIter do
    Update U_r^{2G(g)(k)} by Eq. (10);
    for r ← 1 to s+t do
        Update U^{1G(g)(k)}, H^{1G(g)(k)}, H^{2G(g)(k)} by Eqs. (9),
        (11), and (12), respectively;
    end
    for r ← s+1 to s+t do
        Update V_r^{(k)} by Eq. (13);
    end
    Normalize these CE, CI and V_r^{(k)} by Eq. (14);
end
return U^{1G(g)}, U_r^{2G(g)}, H^{1G(g)}, and H^{2G(g)}.
```

C. COMPUTATIONAL COMPLEXITY OF THE ITERATIVE ALGORITHM

For each round of iteration in Algorithm 1, the computational complexity of Eq. (9) that calculates $U^{1G(g)}$ is $\mathcal{O}(5mn_rc{+}3mck_1^{G(g)}{+}m(k_1^{G(g)}{+}k_2^{G(g)})c{+}mk_1^{G(g)})$. Since we have $c \ll (k_1^{G(g)}{+}k_2^{G(g)})$ and $(k_1^{G(g)}{+}k_2^{G(g)}) \ll n$, the computational complexity of Eq. (9) can be rewritten as $\mathcal{O}(\sum_{r=1}^{s+t} mn_rc)$. Similarly, the computational complexity of Eqs. (10), (11), (12), and (13) are $\mathcal{O}(mn_rc)$, $\mathcal{O}(\sum_{r=1}^{s+t} mk_1^{G(g)}n_r)$, $\mathcal{O}(\sum_{r=1}^{s+t} mk_2^{G(g)}n_r)$, and $\mathcal{O}(\sum_{r=1}^{s+t} mn_rk)$, respectively. Then, the maximal computational intensity in each round of iteration is $\mathcal{O}(\sum_{r=1}^{s+t} mn_rk)$. In summary, the computational complexity of Algorithm 1 is $\mathcal{O}(\sum_{r=1}^{s+t} maxIter{\cdot}mn_rk)$.

V. EXPERIMENTAL EVALUATION

In this section, we use 20Newsgroups and Sentiment as benchmark datasets to compare MGTL with other state-of-the-art transfer learning methods.

A. DATA PREPARATION

20-Newsgroups[1] includes a large number of newsgroup examples that are distributed across twenty different newsgroups [19], [20]. Some similar ones can be grouped into one top-category, e.g., the top category *rec* includes four subcategories such as *autos*, *motorcycles*, *sport.baseball* and *sport.hockey*. The corresponding tasks are constructed as follows:

First, two top categories *rec* and *sci* are chosen as positive and negative classes, respectively. Then we randomly selected two subcategories from the above top categories respectively as the source domain, and then generated the target domain in a similar way. Thus, 144 ($P_4^2{\times}P_4^2$) traditional transfer learning classification tasks are constructed in this way. Second, we replace one subcategory of *sci* as another subcategory which is selected from *comp* or *talk* to construct a new target domain. Then, 384 ($P_4^2{\times}P_4^1{\times}8$) tasks are produced. Since a new top category that does not exist in the source domain is used to generate the new target domain, more specific information on different perspectives exists in this new kind of task. To avoid negative transfer, we choose 334 new classification tasks according to their initial accuracies, which are higher than 50% set by Logistic regression. In summary, we have 144 traditional transfer learning tasks and 334 new ones on the 20-Newsgroups dataset.

Sentiment Data[2] contains reviews from four fields *books*, *dvd*, *electronics*, and *kitchen*. To verify the adaptability of MGTL, we generate multi-source and high-dimension sentiment classification tasks. We first randomly choose two fields, which includes 400 positive and 400 negative examples, as source domain and one rest field, which includes 200 positive and 200 negative examples, as the target domain. To show that MGTL can be applied to the sentiment data set with different examples, we select 800 examples in each domain. Then, 24 sentiment tasks are generated.

B. EXPERIMENTAL SETTING

1) ALGORITHMS IN COMPARISON

(1) Traditional machine learning algorithm Logistic Regression (LR) [8]. We use the data in the source domain to train

[1] http://people.csail.mit.edu/jrennie/20Newsgroups/
[2] http://www.cs.jhu.edu/mdredze/datasets/sentiment/

TABLE 3. Parameter settings of latent space groups.

Latent Feature Space Groups (G_g)	$G_{(1)}$			$G_{(2)}$			$G_{(3)}$			$G_{(4)}$		
Latent Feature Spaces (l_e)	l_1	l_2	l_3	l_3	l_4	l_5	l_5	l_6	l_7	l_7	l_8	l_9
Dimension of Latent Feature Space (k)	6	7	8	8	9	10	10	11	12	12	13	14

TABLE 4. Performances (%) on 20-Newsgroups (10 repeated experiments).

		LR	NMTF	DTL	TCL	Tri-TL	MBTL	MGTL
Average Performance of Total 144 Tasks	Accuracy	65.57	70.86	82.23	89.92	94.65	97.79±0.01	**98.02±0.00**
	F_1-Measure	63.45	64.47	80.94	89.67	94.37	97.76±0.01	**98.01±0.00**
	Precision	68.40	64.12	79.33	88.91	93.59	97.09±0.01	**97.35±0.00**
	NumNT	-	-	21	13	1	**0**	**0**
Average Performance of Total 334 Tasks	Accuracy	66.6	78.09	86.69	87.23	92.06	96.96±0.01	**97.41±0.00**
	F_1-Measure	65.4	74.03	83.78	86.91	91.79	96.94±0.01	**97.39±0.00**
	Precision	68.9	72.81	81.34	86.26	91.14	96.46±0.01	**97.30±0.00**
	NumNT	-	-	31	32	13	3	**0**

the classifier and use the data in the target domain to test. (2) Traditional transfer learning model Non-negative Matrix Tri-Factorization (NMTF). We use NMTF in [17] as a baseline transfer learning method. (3) Transfer learning methods, including DTL [5], TCL [17], Tri-TL [6], and MBTL [33].

2) PARAMETER SETTINGS

Since it is extremely difficult to formalize the latent information or quantify the relationships among different latent feature spaces and the groups, MGTL can not automatically tune the optimal number of latent spaces and the corresponding groups. Therefore, we evaluate the MGTL on the dataset by empirically searching the parameter space. The parameter settings for latent space groups are shown in Table 3. We set $NOG = 4$, $NOL = 9$, $NOO = 1$, $NPG = 3$, and $maxIter = 200$. Additionally, we implement LR using Matlab.[3] NMTF is can be obtained from [17]. We set the parameters of the above methods as their default ones.

3) EVALUATION METRICS

To check out the classification results comprehensively, we use three widely used evaluation metrics:

a: ACCURACY

$$Accuracy = \frac{|\{d : d \in D \wedge f(d) = y(d)\}|}{n}$$

where $y(d)$ is the true label of example d, $f(d)$ is the label predicted by the classification model and n is the number of the examples.

b: F_1-MEASURE

$$F_1\text{-}Measure = (NF_1 + PF_1)/2$$

where NF_1(F_1 on negative extractions) $= (2{\cdot}NP{\cdot}NR)/(NP+NR)$, PF_1(F_1 on positive extractions) $= (2{\cdot}PP{\cdot}PR)/(PP+PR)$, NR(recall on negative extractions) $= d/(d+c)$, NP(precision on negative extractions) $= d/(d+b)$, PR(recall on positive extractions) $= a/(a+b)$, PP(precision on positive extractions) $= a/(a+c)$.

[3]http://www.kyb.tuebingen.mpg.de/bs/people/pgehler/code/index.html

c: PRECISION

$$Precision = a/(a+c)$$

In addition, we also use "Number of Negative Transfer" (*NumNT*) as another evaluation metric for transfer learning [34].

	Predicted Positive Pairs	Predicted Negative Pairs
Positive Pairs	a	b
Negative Pairs	c	d

C. EXPERIMENTAL RESULTS

We compare our MGTL with LR, NMTF, DTL, TCL, Tri-TL and MBTL on the 20-Newsgroups and sentiment tasks respectively.

1) COMPARISON ON 20-NEWSGROUPS

From the results in Table 4, we observe that MGTL obtains the best average performance on 144 traditional transfer learning tasks and 334 new tasks. In addition, we find that all compared transfer methods result in a negative transfer. Only MGTL can successfully avoid negative transfers on all tasks. There are two reasons why MGTL can achieve satisfactory performance. First, MGTL constructs multiple different latent feature spaces, which contain more latent information from different perspectives, to build a shared bridge. Then, MGTL, which can utilize more latent information, may avoid negative transfer when domain distributions are dominated by some latent information that is easy to be ignored. Second, since many latent feature space groups are constructed by integrating the adjacent latent spaces that have a similar latent feature dimension into one latent space group, MGTL enhances the relevance among these latent space groups by keeping the adjacent latent space groups contain one same latent space at least. Consequently, the latent information obtained in different latent spaces can be utilized effectively. In addition, DTL, TCL, and Tri-TL are better than

TABLE 5. Performances (%) on sentiment (10 repeated experiments).

		LR	NMTF	DTL	TCL	Tri-TL	MBTL	MGTL
Average Performance (400 examples in each domain)	**Accuracy**	74.23	58.75	72.0	58.26	58.27	75.87±0.05	**77.52±0.03**
	F_1-Measure	74.14	58.52	71.94	58.16	57.26	75.75±0.04	**77.43±0.03**
	Precision	74.54	60.38	73.24	60.11	59.52	76.45±0.04	**78.83±0.03**
	NumNT	-	-	9	12	12	3	**0**
Average Performance (800 examples in each domain)	**Accuracy**	76.05	59.02	75.63	59.86	58.64	76.25±0.04	**78.04±0.03**
	F_1-Measure	75.98	58.92	75.54	59.61	57.78	76.18±0.04	**78.02±0.03**
	Precision	76.41	60.89	76.95	61.16	58.98	77.51±0.04	**79.31±0.03**
	NumNT	-	-	7	12	12	5	**0**

TABLE 6. Average performances (%) comparison between MGTL and the variants (10 repeated experiments).

		MGTL-$G_{(1)}$	MGTL-$G_{(2)}$	MGTL-$G_{(3)}$	MGTL-$G_{(4)}$	MGTL-NoOverlap	MGTL	MGTL-Direct
Performance of Total 144 Tasks on 20-Newsgroups	Accuracies	97.42±0.00	97.14±0.01	97.27±0.01	97.52±0.00	97.84±0.00	98.02±0.00	**98.07±0.00**
	F_1-Measure	97.42±0.00	97.12±0.01	97.27±0.01	97.51±0.00	97.82±0.00	98.01±0.00	**98.06±0.00**
	Precision	96.71±0.00	96.46±0.01	96.58±0.01	96.83±0.00	97.11±0.00	97.35±0.00	**97.46±0.00**
	NumNT	**0**	1	**0**	**0**	**0**	**0**	**0**
Performance of Total 334 Tasks on 20-Newsgroups	Accuracies	96.84±0.00	96.35±0.01	96.79±0.00	96.79±0.01	96.62±0.01	97.41±0.00	**97.51±0.00**
	F_1-Measure	96.82±0.00	96.32±0.01	96.76±0.00	96.77±0.00	96.62±0.01	97.39±0.00	**97.50±0.00**
	Precision	96.72±0.00	96.21±0.01	96.67±0.00	96.67±0.00	96.51±0.01	97.30±0.00	**97.41±0.00**
	NumNT	**0**	1	1	**0**	2	**0**	**0**
Performance (400 examples in each domain) on Sentiment	Accuracies	76.83±0.03	76.54±0.04	76.82±0.03	76.75±0.04	76.64±0.04	**77.52±0.03**	77.39±0.03
	F_1-Measure	76.81±0.03	76.52±0.04	76.81±0.03	76.75±0.04	76.64±0.04	**77.43±0.03**	77.37±0.03
	Precision	78.01±0.03	77.73±0.04	78.03±0.03	77.95±0.04	77.82±0.04	**78.83±0.03**	78.79±0.03
	NumNT	1	2	1	1	1	**0**	**0**
Performance (800 examples in each domain) on Sentiment	Accuracies	77.54±0.03	77.02±0.04	77.21±0.03	77.37±0.03	77.16±0.04	**78.04±0.03**	77.92±0.03
	F_1-Measure	77.53±0.03	77.02±0.04	77.21±0.03	77.35±0.03	77.15±0.04	**78.02±0.03**	77.89±0.03
	Precision	78.74±0.03	78.32±0.04	78.52±0.03	78.64±0.03	78.41±0.04	**79.31±0.03**	79.22±0.03
	NumNT	1	1	1	1	1	**0**	**0**

LR and NMTF, which means traditional learning methods may fail in transfer learning tasks. On the other hand, MBTL outperforms DTL, TCL, and Tri-TL, whose reason may be that MBTL which builds multiple transfer bridges can obtain more useful latent information to fit domain distribution. Overall, MGTL achieves the best performance regardless of running on the 144 traditional transfer learning tasks which contain more common features or on the 334 new tasks which contain more specific features.

2) COMPARISON ON SENTIMENT

To further verify the adaptability of MGTL, we construct the multi-source and high-dimension classification tasks on sentiment dataset with two source and one target domains. In Table 5, we can find that MGTL obtains the best experimental results once more and outperforms all the compared methods. Notably, the performance of MGTL is very stable on these challenging sentiment tasks with fewer examples. As shown in Table 5, we can find that all the compared algorithms occur negative transfer on these multi-source and high-dimension classification tasks that are more challenging than traditional sentiment tasks. The reason that all the compared transfer methods that can deal with topic classification tasks fail in the more challenging tasks is that these methods cannot utilize the latent information effectively. Only MGTL avoids negative transfer successfully and exhibits the best performance.

In summary, these results not only prove the effectiveness of MGTL on cross-domain text classification, but also prove the adaptability of MGTL for the multi-source transfer learning tasks.

D. EFFECTIVENESS OF MGTL

To verify the effect of MGTL, we construct four single group approaches, including MGTL-$G_{(1)}$, MGTL-$G_{(2)}$, MGTL-$G_{(3)}$, and MGTL-$G_{(4)}$. Actually, these methods are trained in the four different latent space groups constructed in MGTL, respectively. The parameter settings of these approaches are shown in Table 3. In addition, we construct two variants of MGTL including MGTL-Direct and MGTL-NoOverlap. MGTL-Direct learns the high-level concepts in different latent groups with direct-promotion strategy. MGTL-NoOverlap regroups these latent spaces (including l_1, l_2, l_3, l_4, l_5, l_6, l_7, l_8 and, l_9) into three groups including $G_{(1,2,3)}$, $G_{(4,5,6)}$ and $G_{(7,8,9)}$ (each group contains three different latent spaces, e.g. group $G_{(1,2,3)}$) contains latent spaces l_1, l_2 and l_3, and learns the high-level concepts in these latent groups with indirect-promotion strategy. The experimental results are shown in Table 6.

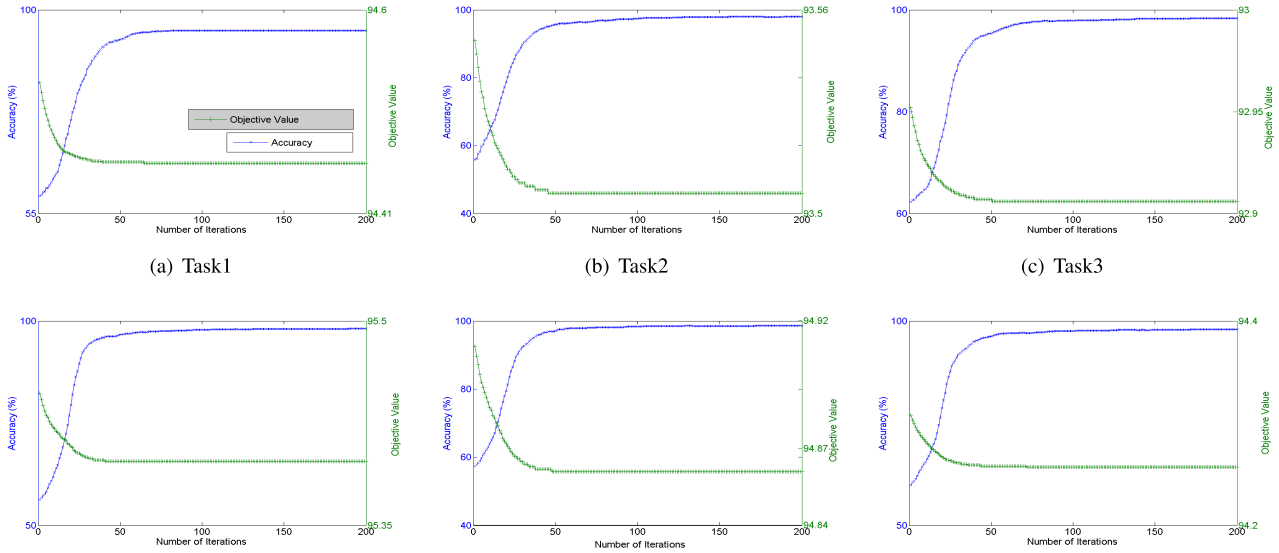
First, we find that MGTL and MGTL-Direct outperform all the single latent space group approaches on all the tasks, which means that the strategy of MGTL can improve the performance of classification effectively. Second, we find that MGTL outperforms MGTL-Direct on sentiment tasks, and MGTL-Direct outperforms MGTL on 20-Newsgroups tasks, which means that the indirect-promotion strategy is more suitable for handling challenging tasks and the direct-promotion strategy is suitable for dealing with traditional tasks. Third, considering the computational complexity, which will be analyzed in the next subsection, we adopt the indirect-promotion strategy for MGTL. Fourth, MGTL and MGTL-Direct outperform MGTL-NoOverlap on all the tasks. This is because the neighbouring two groups

TABLE 7. Running time of MGTL and other compared methods (s).

	LR	NMTF	DTL	TCL	Tri-TL	MBTL	MGTL-NoOverlap	MGTL	MGTL-Direct
Traditional 20-Newsgroups Task	11.4	14.1	21.1	24.3	45.2	75.6	118.6	124.5	225.9
New 20-Newsgroups Task	12.3	15.2	22.4	25.8	47.5	78.5	122.5	133.7	253.1
Sentiment Task with 400 examples	17.9	24.5	32.3	36.7	95.1	127.4	249.6	279.6	412.3
Sentiment Task with 800 examples	23.1	31.2	39.2	44.8	117.4	153.1	310.2	356.4	533.8

TABLE 8. The parameter influence on performance (%) of algorithm MGTL.

Sampling ID	$G_{(1)}$			$G_{(2)}$			$G_{(3)}$			$G_{(4)}$			Problem ID			
	$l_{(1)}$	$l_{(2)}$	$l_{(3)}$	$l_{(3)}$	$l_{(4)}$	$l_{(5)}$	$l_{(5)}$	$l_{(6)}$	$l_{(7)}$	$l_{(7)}$	$l_{(8)}$	$l_{(9)}$	1	2	3	4
1	5	9	10	10	7	11	11	9	13	13	14	16	98.71	97.81	98.01	98.31
2	7	8	7	7	11	12	12	13	14	14	15	14	98.26	97.76	98.06	98.21
3	4	5	9	9	9	10	10	11	11	11	13	16	98.21	97.41	97.63	98.43
4	6	7	6	6	8	8	8	12	13	13	11	12	98.66	97.96	98.21	98.52
5	7	6	8	8	10	9	9	10	10	10	12	13	98.61	97.91	98.11	98.17
6	8	9	7	7	8	12	12	9	12	12	15	15	98.36	97.45	97.74	98.21
7	5	5	10	10	7	11	11	11	13	13	14	16	98.46	97.36	97.86	98.26
8	8	7	6	6	11	8	8	13	12	12	13	15	98.41	97.45	97.95	98.45
9	4	6	9	9	10	9	9	12	14	14	12	14	98.51	97.31	98.16	98.17
Mean													98.46	97.60	97.97	98.30
Variance													0.027	0.058	0.034	0.015
This paper	6	7	8	8	9	10	10	11	12	12	13	14	98.51	97.45	97.95	98.31



(a) Task1 (b) Task2 (c) Task3

(d) Task4 (e) Task5 (f) Task6

FIGURE 3. The Performance of MGTL and objective value vs. the number of iterations.

in Table 4 contain the same one latent space at least, then the correlation among these groups obviously overtops the one among $G_{(1,2,3)}$, $G_{(4,5,6)}$ and $G_{(7,8,9)}$. These results not only prove our empirical hypothesis in section 1, but also prove the validity of the grouping strategy in MGTL.

E. RUNNING TIME

We randomly choose four tasks from 20-Newsgroups and sentiment respectively, and check the running time empirically. From the experimental results in Table 7, we find that MGTL-NoOverlap, MGTL and MGTL-Direct run longer than the other transfer learning algorithms. This is because the running time of the algorithm based on high-level concept is proportional to the number of high-level concepts, and these MGTL methods construct more latent spaces and learn more high-level concepts to obtain and utilize more latent information. Nevertheless, the running time of MGTL is within acceptable limits.[4] Since the interrelation complexities among high-level concepts learned on different groups using indirect-promotion strategy are less than the ones using the direct-promotion strategy, MGTL-NoOverlap and MGTL based on indirect-promotion strategy run faster than MGTL-Direct based on direct-promotion strategy. Additionally, since the difference in computational complexity

[4]The configuration of computing platform: Intel Core i5-3470s CPU 2.9GHz, RAM 8.0GB.

between the two strategies is exponential, the difference in running time between methods using these two strategies will be more significant as the number of groups increases.

F. PARAMETER SENSITIVITY

The parameter sensitivity of MGTL is investigated with changing parameters l_1, l_2, l_3, l_4, l_5, l_6, l_7, l_8, and l_9, which represent the numbers of high-level concepts in nine latent spaces, respectively. We elaborate these parameters in Table 3. We randomly choose nine combinations of these parameters when $l_1 \in [4, 8]$, $l_2 \in [5, 9]$, $l_3 \in [6, 10]$, $l_4 \in [7, 11]$, $l_5 \in [8, 12]$, $l_6 \in [9, 13]$, $l_7 \in [10, 14]$, $l_8 \in [11, 15]$, and $l_9 \in [12, 16]$, and then investigate the performance of MGTL on 4 randomly chosen traditionally transfer learning tasks, which can verify the stableness of MGTL when the parameters vary in a widely range. From the results shown in Table 8, we find that the average accuracy of the nine parameter combinations for each selected task is almost the same as the accuracy of using the default parameters, and the variance is small. Therefore, MGTL is generally insensitive to parameters selected from a predefined range.

G. ALGORITHM CONVERGENCE

We randomly choose six tasks on *rec vs. sci* to check the convergence of MGTL. In Figure 3, the left and right y-axis and the x-axis indicate the prediction accuracy, the objective value, and the number of iterations, respectively. From the results in Figure 3, we observe that as the number of iterations increases, the objective value decreases and the accuracy of MGTL increases conversely, and both of them converge within 200 iterations.

VI. CONCLUSION

In this paper, to utilize more latent information effectively for knowledge transfer, we proposed a novel approach MGTL. Our method first constructs multiple latent feature spaces and groups them. To enhance the relevance among these latent space groups, MGTL makes the adjacent groups contain one same latent space at least. Then, different groups will have more relevance than raw latent spaces. Second, MGTL learns the shared high-level concepts on different groups utilizing an indirect-promotion strategy to establish a bridge across domains. Then, the computational complexity of MGTL increases linearly as the number of latent space groups increases and is superior to the methods based on direct-promotion. In addition, an effective algorithm is proposed to derive the solution to the optimization problem. Finally, we conduct comprehensive experiments to show that MGTL outperforms all the comparison methods.

It should be noted that although MGTL has achieved excellent results on text classification tasks cross domain, some parameters of MGTL are set empirically. In the future, we will optimize the proposed algorithm to try to adjust some parameters adaptively. Additionally, we intend to adapt MGTL for other applications, such as query expansion, machine translation, recognizing textual entailment.

APPENDIX

The partial differentials of $\mathcal{L}$ are as follows:

$$\frac{\partial \mathcal{L}}{\partial U^{1G(g)}} = \sum_{r=1}^{s+t} 2\cdot(A_r^{G(g)} V_r H^{1G(g)^T} + B_r^{G(g)} V_r H^{1G(g)^T} - X_r V_r H^{1G(g)^T}) \tag{15}$$

$$\frac{\partial \mathcal{L}}{\partial U_r^{2G(g)}} = 2\cdot(A_r^{G(g)} V_r H^{2G(g)^T} + B_r^{G(g)} V_r H^{2G(g)^T} - X_r V_r H^{2G(g)^T}) \tag{16}$$

$$\frac{\partial \mathcal{L}}{\partial H^{1G(g)}} = \sum_{r=1}^{s+t} 2\cdot(U^{1G(g)^T} A_r^{G(g)} V_r + U^{1G(g)^T} B_r^{G(g)} V_r - U^{1G(g)^T} X_r V_r) \tag{17}$$

$$\frac{\partial \mathcal{L}}{\partial H^{2G(g)}} = \sum_{r=1}^{s+t} 2\cdot(U_r^{2G(g)^T} A_r^{G(g)} V_r + U_r^{2G(g)^T} B_r^{G(g)} V_r - U_r^{2G(g)^T} X_r V_r) \tag{18}$$

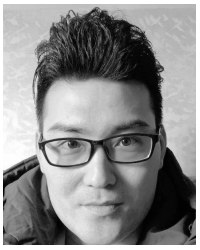
$$\frac{\partial \mathcal{L}}{\partial V_r} = 2\cdot \sum_{g=1}^{NOG} (V_r H_r^{G(g)^T} U_r^{G(g)^T} U_r^{G(g)} H_r^{G(g)} - X_r^T U_r^{G(g)} H_r^{G(g)}) \tag{19}$$

where $U_r^{G(g)} = [U^{1G(g)}, U_r^{2G(g)}]$ and $H_r^{G(g)} = \begin{bmatrix} H^{1G(g)} \\ H^{2G(g)} \end{bmatrix}$.

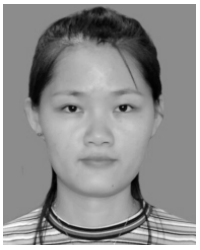
REFERENCES

[1] S. Jialin Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
[2] J. Blitzer, R. McDonald, and F. Pereira, "Domain adaptation with structural correspondence learning," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2006, pp. 120–128.
[3] W. Dai, G.-R. Xue, Q. Yang, and Y. Yu, "Co-clustering based classification for out-of-domain documents," in *Proc. 13th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2007, pp. 210–219.
[4] F. Z. Zhuang, P. Luo, H. Xiong, Q. He, Y. H. Xiong, and Z. Z. Shi, "Exploiting associations between word clusters and document classes for cross-domain text categorization," *Stat. Anal. Data Mining, ASA Data Sci. J.*, vol. 4, no. 1, pp. 100–114, 2011.
[5] M. Long, J. Wang, G. Ding, W. Cheng, X. Zhang, and W. Wang, "Dual transfer learning," in *Proc. 12th SIAM SDM*, 2012, pp. 540–551.
[6] F. Zhuang, P. Luo, C. Du, Q. He, Z. Shi, and H. Xiong, "Triplex transfer learning: Exploiting both shared and distinct concepts for text classification," *IEEE Trans. Cybern.*, vol. 44, no. 7, pp. 1191–1203, Jul. 2014.
[7] F. Z. Zhuang, P. Luo, P. F. Yin, Q. He, and Z. Z. Shi, "Concept learning for cross-domain text classification: A general probabilistic framework," in *Proc. 23th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2013, pp. 1960–1966.
[8] D. Hosmer and S. Lemeshow, *Applied Logistic Regression*. Hoboken, NJ, USA: Wiley, 2004.
[9] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," *Mach. Learn.*, vol. 42, nos. 1–2, pp. 177–196, Jan. 2001.
[10] P. Wei, Y. Ke, and C. K. Goh, "A general domain specific feature transfer framework for hybrid domain adaptation," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 8, pp. 1440–1451, Aug. 2019.
[11] Q. Liu, A. J. Mackey, D. S. Roos, and F. C. N. Pereira, "Evigan: A hidden variable model for integrating gene evidence for eukaryotic gene prediction," *Bioinformatics*, vol. 24, no. 5, pp. 597–605, Mar. 2008.
[12] Y. Zhu, Y. Chen, Z. Lu, S. J. Pan, G. R. Xue, Y. Yu, and Q. Yang, "Heterogeneous transfer learning for image classification," in *Proc. 25th AAAI*, 2011, pp. 1304–1309.
[13] J. Wiens, J. Guttag, and E. Horvitz, "A study in transfer learning: Leveraging data from multiple hospitals to enhance hospital-specific predictions," *J. Amer. Med. Inform. Assoc.*, vol. 21, no. 4, pp. 699–706, Jul. 2014.

[14] P.-J. Kindermans, M. Tangermann, K.-R. Müller, and B. Schrauwen, "Integrating dynamic stopping, transfer learning and language models in an adaptive zero-training ERP speller," *J. Neural Eng.*, vol. 11, no. 3, Jun. 2014, Art. no. 035005.
[15] R. Xia, J. Jiang, and H. He, "Distantly supervised lifelong learning for large-scale social media sentiment analysis," *IEEE Trans. Affect. Comput.*, vol. 8, no. 4, pp. 480–491, Oct. 2017.
[16] C. Ding, T. Li, W. Peng, and H. Park, "Orthogonal nonnegative matrix tri-factorizations for clustering," in *Proc. 12th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2006, pp. 126–135.
[17] Z. Chen and W. X. Zhang, "Domain adaptation with topical correspondence learning," in *Proc. 23th Int. Joint Conf. Artif. Intell. (IJCAI)*, 2013, pp. 1280–1286.
[18] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2000, pp. 556–562.
[19] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu, "Boosting for transfer learning," in *Proc. 24th Int. Conf. Mach. Learn. (ICML)*, 2007, pp. 193–200.
[20] J. Gao, W. Fan, J. Jiang, and J. Han, "Knowledge transfer via multiple model local structure mapping," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2008, pp. 283–291.
[21] J. Jiang and C. Zhai, "A two-stage approach to domain adaptation for statistical classifiers," in *Proc. 16th ACM Conf. Conf. Inf. Knowl. Manage. (CIKM)*, 2007, pp. 401–410.
[22] S. Uguroglu and J. Carbonell, "Feature selection for transfer learning," in *Proc. Mach. Learn. Knowl. Discovery Databases (ECML PKDD)*, 2011, pp. 430–442.
[23] S. J. Pan, J. T. Kwok, and Q. Yang, "Transfer learning via dimensionality reduction," in *Proc. 23rd AAAI*, 2008, pp. 677–682.
[24] M. Long, J. Wang, G. Ding, D. Shen, and Q. Yang, "Transfer learning with graph co-regularization," in *Proc. 26th AAAI*, 2012, pp. 1033–1039.
[25] J. Jiang and C. X. Zhai, "Instance weighting for domain adaptation in NLP," in *Proc. 45th ACL*, 2007, pp. 264–271.
[26] Z. Yang, R. Salakhutdinov, and W. W. Cohen, "Transfer learning for sequence tagging with hierarchical recurrent networks," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–10.
[27] H. Wang, H. Huang, F. Nie, and C. Ding, "Cross-language Web page classification via dual knowledge transfer using nonnegative matrix tri-factorization," in *Proc. 34th Int. ACM SIGIR Conf. Res. Develop. Inf. (SIGIR)*, 2011, pp. 933–942.
[28] D. Oyen and T. Lane, "Bayesian discovery of multiple Bayesian networks via transfer learning," in *Proc. IEEE 13th Int. Conf. Data Mining (ICDM)*, Dec. 2013, pp. 577–586.
[29] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[30] Z. Ding and Y. Fu, "Robust transfer metric learning for image classification," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 660–670, Feb. 2017.
[31] F. Liu, X. Xu, S. Qiu, C. Qing, and D. Tao, "Simple to complex transfer learning for action recognition," *IEEE Trans. Image Process.*, vol. 25, no. 2, pp. 949–960, Feb. 2016.
[32] J. Gao, H. Ling, W. Hu, and J. Xing, "Transfer learning based visual tracking with Gaussian processes regression," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2014, pp. 188–203.
[33] X. Hu, J. Pan, P. Li, H. Li, W. He, and Y. Zhang, "Multi-bridge transfer learning," *Knowl.-Based Syst.*, vol. 97, pp. 60–74, Apr. 2016.
[34] J. Pan, X. Hu, Y. Zhang, P. Li, Y. Lin, H. Li, W. He, and L. Li, "Quadruple transfer learning: Exploiting both shared and non-shared concepts for text classification," *Knowl.-Based Syst.*, vol. 90, pp. 199–210, Dec. 2015.
[35] J. Pan, X. Hu, P. Li, H. Li, W. He, Y. Zhang, and Y. Lin, "Domain adaptation via multi-layer transfer learning," *Neurocomputing*, vol. 190, pp. 10–24, May 2016.
[36] W. Pengfei, K. Yiping, and G. C. Keong, "Domain specific feature transfer for hybrid domain adaptation," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2017, pp. 1027–1032.
[37] Y. Yan, W. Li, H. Wu, H. Min, M. Tan, and Q. Wu, "Semi-supervised optimal transport for heterogeneous domain adaptation," in *Proc. 27th Int. Joint Conf. Artif. Intell. (IJCAI)*, Jul. 2018, pp. 2969–2975.
[38] H. Li, S. J. Pan, R. Wan, and A. C. Kot, "Heterogeneous transfer learning via deep matrix completion with adversarial kernel embedding," in *Proc. AAAI Conf. Artif. Intell.*, vol. 33, Jul. 2019, pp. 8602–8609.
[39] Y. Luo, Y. Wen, T. Liu, and D. Tao, "Transferring knowledge fragments for learning distance metric from a heterogeneous domain," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 4, pp. 1013–1026, Apr. 2019.
[40] J. T. Zhou, S. J. Pan, and I. W. Tsang, "A deep learning framework for hybrid heterogeneous transfer learning," *Artif. Intell.*, vol. 275, pp. 310–328, Oct. 2019.

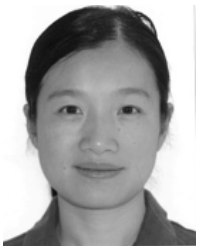


JIANHAN PAN received the Ph.D. degree in computer science from the Hefei University of Technology, China, in 2016. He is currently an Associate Professor with the School of Computer Science and Technology, Jiangsu Normal University. His research interests include data mining, machine learning, and transfer learning.



TENG CUI received the B.S. degree in the Internet of Things engineering from the Binjiang College, Nanjing University of Information Science and Technology, Nanjing, China. She is currently pursuing the M.S. degree in software engineering with Jiangsu Normal University, Xuzhou, China. Her research interests include machine learning and transfer learning.

THUC DUY LE is currently a Senior Lecturer with the School of Information Technology and Mathematical Sciences, University of South Australia. His research interest includes causal discovery methods and their applications in bioinformatics. He is also a DECRA Fellow, for the period of 2020–2022. He has served as an Academic Editor for *PLOS One* and a reviewer for many top conferences and journals in data mining and bioinformatics.



XIAOMEI LI received the M.S. degree in computer science from Fuzhou University, China, in 2011. She is currently pursuing the Ph.D. degree with the University of South Australia. Her research focuses on the development of computational methods for cancer subtype and prognosis based on bulk data and single-cell sequencing data.

JING ZHANG (Senior Member, IEEE) received the M.S. degree in computer science from the Graduate University of Chinese Academy of Sciences, Beijing, China, in 2006, and the Ph.D. degree in computer science from the Hefei University of Technology, Hefei, China, in 2015. He was a Visiting Research Scholar with the University of Pittsburgh, from March 2019 to March 2020, under the sponsorship of 2018 Jiangsu Overseas Visiting Scholar Program for University Prominent Young and Middle-Aged Teachers and Presidents. He is currently an Associate Professor with the School of Computer Science and Engineering, Nanjing University of Science and Technology. His research interests include data mining and machine learning.