IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 6 February 2023, accepted 12 February 2023, date of publication 16 February 2023, date of current version 22 March 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3246090

RESEARCH ARTICLE

Pairwise Metric Learning With Angular Margin for Tag Recommendation

ZHENGSHUN FEI, (Member, IEEE), JINGLONG WANG, BINGQIANG HUANG, AND XINJIAN XIANG

School of Automation and Electrical Engineering, Zhejiang University of Science and Technology, Hangzhou 310023, China

Corresponding author: Zhengshun Fei (zsfei@zju.edu.cn)

This work was supported in part by the Key Research and Development Program of Zhejiang Province under Grant 2018C01085, in part by the Natural Science Foundation of Zhejiang Province under Grant LQ15F030006, in part by the Research Project of Zhejiang Provincial Department of Education under Grant Y202249418, and in part by the Key Project of Zhejiang University of Science and Technology Postgraduate Innovation Fund under Grant 2021yjskc04.

ABSTRACT The users on the Internet have been growing exponentially, and tag recommendation can automatically provide users with a selection of tags of interest to meet their personalized needs. Users can utilize these tags to freely annotate their favorite resources, making them efficient and fast in retrieving related resources. Tensor factorization methods are commonly used in tag recommendation at present. These methods model the user $\times$ item $\times$ tag interactions, transform the latent feature representations of users, items and tags into low-rank matrices and use inner products for prediction. However, the problem of using inner product is that it does not satisfy the triangle inequality, it ignores the distance relationship among entity pairs and cannot capture the fine-grained preference information. Metric learning in recommendation domains focus on using pairwise loss, which assumes that different categories (such as users, items and tags) have fixed margins. Different categories often have different intra-class variations. With fixed margins, it is often difficult to accurately distinguish between positive and negative samples, thus reducing recommendation performance and limiting the expression ability of the model. In this study, the metric learning method is used to explore the distance relationship among user $\times$ item $\times$ tag triplet, and the existing metric learning based methods (namely LRML, CML, SML) are applied to the tag recommendation. A pairwise metric learning method with angular margin is proposed, named PMLT. The pairwise distance relationship between user-tag and item-tag is modeled for the information of different entities. And an extra angular margin regularizer is added to the original pairwise loss to control the size of angular margin for user-tag and item-tag respectively. The strength of the constrained angular margin regularizer is controlled to dynamically adjust the distance changes of entity. This method constrains the fixed margin and also the angular margin of user-tag and item-tag. Compared with the traditional metric learning method, this method can capture additional relationship structure and has good recommendation performance. Finally, we conducted extensive experiments on two datasets, LastFm and Movielens, and the experimental results showed that the proposed method PMLT outperform the state-of-the-art baseline in the evaluation metrics Recall@N and NDCG@N, and obtain better prediction quality. We also analyze the influence of different parameters and internal components on the performance of the proposed method, which improves the interpretability of the proposed method.

INDEX TERMS Tag recommendation, tensor factorization, metric learning, angular margin.

I. INTRODUCTION

Tag recommendation [1] is a popular feature in many web applications (LastFm, Movielens, Flick and Bibsonomy) that can suggest several relevant tags for users and simplify the tagging process to help users annotate web resources of interest (e.g., songs, images and videos). Through the deployment of tag recommendation, the platform can improve user experience, promote product dissemination, and improve the quality of information retrieval services. Since different users

The associate editor coordinating the review of this manuscript and approving it for publication was Chong Leong Gan.

tend to provide different tags for the same item, it is quite important to provide individual users with tags that match their personalities. Tag recommendation [2], [3], [4], [5] infers the probability that a given user is likely to annotate tags for a specific item from the given historical interactive data, and also needs to satisfy the user to describe an item using keywords (tags) based on their own understanding to predict a personalized Top-N tag list. For example, Movielens platform recommends relevant tags for users according to their individual needs to describe a movie of interest. Users can quickly query the movie they are interested in by using these tags. LastFm assigns users tags that highlight an artist's style of music, and users find favorite songs by looking up certain tags to find a group of artists of interest. Flick users can utilize tags to annotate images of interest, and users are able to describe scenes or objects in a particular image using tags. Currently, tag recommendation is becoming more and more important in various application scenarios. Different from the traditional item recommendation task, which only focuses on the user × item relationship, tag recommendation considers the relationship among user × item × tag triplet at the same time, which is more in line with the complex recommendation scenarios in reality.

Recently, matrix factorization (MF) methods [6], [7], [8] have been successfully applied in various item recommendations. These methods learn latent representations of users and items, approximate user-item interaction matrix as two low-rank matrices, and calculate the similarity scores of users and items by the inner product of their corresponding vectors. However, the limitation of using inner product is that the result does not satisfy the triangle inequality and it is difficult to capture the users preference information. As a generalization of matrix factorization, tensor factorization methods (TF) [9], [10], [11] are widely used in tag recommendation. These methods learn the latent representations of users, items, and tags and compute the similarity scores of entity pairs based on the inner product among the user × item × tag triplet, which makes the use of TF in tag recommendation face the same problem as MF. Assume that each user u and each item i interacts with two tags t_1 and t_2. In this case, the tensor factorization based approach attempts to keep with u and t_1 or t_2 (i and t_1 or t_2) close to each other in the latent space, but does not try to keep the two vectors of t_1 and t_2 close to each other. It cannot accurately capture the similarity between users (between items or between tags) without satisfying the triangle inequality, resulting in the difficulty of accurately predicting a Top-N tag list.

Among the various existing tag recommendation methods, researchers have developed various interaction modes for user × item × tag triples that describe the contents of the item through tags and give the user a recommendation of preferred item. To model the triplet of entity, tucker decomposition (TD) methods encode the user × item × tag triplet relationship into three decomposition dimensions. HOSVD [12] and RTF [13] modeled the three interactions for entity pair vectors. The time complexity of this method is $O(k^3)$, and it is not applicable to medium and large datasets because of the large amount of computation. As a special case of TD, canonical decomposition (CD) [14] also has the same problems as TD. In order to improve the performance of tag recommendation, an effective tensor factorization method PITF [10] is proposed for tag recommendation by modeling the pairwise interaction of user × item × tag triplet and factorization dimension is $O(k)$. As PITF is a Tucker decomposition method with linear time complexity and good recommendation quality, many improved methods have been proposed on the basis of PITF. For example, NITF [15], in contrast, uses Gaussian kernel functions to improve the capacity of the model. ABNT [16] models the nonlinear relationship among users, items, and tags by a multilayer perceptron. ATF [17] uses adversarial techniques [18] to enhance the robustness of the PITF method. In addition, latent Dirichlet Allocation (LDA) based tag recommendation methods [19], [20] can further model the underlying topics of documents and users as a distribution of tags. These methods jointly model users, items and metadata at the topic level by extending the LDA model to describe different items by their content features. Researchers also use graph-based approaches to interact different entities with graphs [21], [22], [23], and the relationship of entities is modeled as a graph network structure. However, in the above methods, the inner product among user × item × tag triplets is used to predict the similarity score, and little attention is paid to the distance relationship, which affects the results of Top-N tag recommendation to some extent. Different from the existing work, the proposed pairwise metric learning method is able to learn both user-tag and item-tag distances, making the tags associated with users and items closer together.

Metric learning (ML) is widely used in the field of recommendation, in which data that are near each other tend to be more similar, and data that are far apart are as different as possible. In recent studies, researchers have proposed a number of metric learning recommendation methods to compute the similarity of distance between users × items. Typical methods include CML [24], LRML [25], TransCF [26], SML [27], and PMLAM [28], aiming to bringing the user closer to the items he is interested in, while keeping the user away from items he is not interested in. These methods recommend the top N items that the user is most interested in. CML uses pairwise loss to ensure that the distance between the user and the negative items (items the user does not like) is greater than the distance between the current user and the positive items (items the user likes). The rationale of this is based on the assumption that the user is likely to be closer to the items it likes than to the items it does not like. Although the CML recommendation method is effective, it suffers from two aspects [29]. First, CML attempts to embed the user and the positive items as identical points in Euclidean space. Therefore, when there is a large number of user-item interactions, the loss function of CML leads to instability during the training process. When the distance between the user and the negative item is too close, the item that the user is not interested in may be incorrectly recommended to the user. Secondly, CML optimizes the pairwise loss with a fixed margin, which makes CML unable to understand the user-item distance variation and also requires tedious hyperparameter adjustment for different margins. This may mislead other users in predicting the outcome of the item. And the latent relationship metric learning approach [25], [26] improves the geometric inflexibility of ML. In order to alleviate the limitations of fixed margins, adaptive margin approaches are proposed [27], [28], which allows adaptive adjustment of the distances among user, positive item and negative item.

However, the above method based on metric learning assumes that there is a fixed margin between different categories, through which positive and negative samples can be distinguished. Different entity pairs often have different intra-class variations, and the fixed margin does not change with the change of entity. It is difficult to dynamically adjust the change of entity distance, which limits the expressiveness of the model. Therefore, it is necessary to select the value of the appropriate fixed margin. Generally, a larger fixed margin often leads to difficulty in training convergence, and a smaller fixed margin makes it difficult to correctly distinguish between positive and negative samples, thus producing an unstable recommendation result. Different from previous work, we constrain the angular margin of user-tag and item-tag. On the basis of pairwise loss, an angular margin regularizer is considered, which can dynamically adjust the distance variation of different entity pairs. Compared with existing metric learning recommendation methods, the proposed angular margin pairwise metric learning method can produce more stable prediction results.

In this paper, we apply existing metric learning methods (i.e. LRML, CML, and SML) for tag recommendation systems and propose a pairwise metric learning method with angular margin for tag recommendation, named PMLT. For the similarity relationships of users, items and tags, a pairwise metric learning method is proposed to construct the Euclidean distance of user-tag and item-tag respectively. This method learns the distance measurement of user-tag and item-tag at the same time, which can reduce the distance between similar data of user-tag and item-tag, and expand the distance between different data. Aiming at the problem that the fixed margin is difficult to distinguish between positive and negative samples, an entity pair relationship based on the angular margin constraint is proposed to dynamically adjust the user-tag and item-tag distance changes. Based on the original pairwise loss, a new objective function combining angular margin is established, and an angular margin regularization is added, which can control the angular margin of user-tag and item-tag respectively, and improve the stability of the model by adjusting the angular margin. Finally, extensive experiments are conducted on two real datasets, LastFm and Movielens, and the influence of different parameters and components on tag recommendation performance is also analyzed. The results showed that PMLT has better than other baselines in terms of Recall@N and NDCG@N, which verified the effectiveness of the proposed approach. The main contributions of this paper are as follows.

1) The effectiveness of metric learning in tag recommendation is analyzed, and existing metric learning methods (CML, LRML and SML) are applied to tag recommendation. Aiming at the relationship among user $\times$ item $\times$ tag triplet, a pairwise metric learning method is proposed to construct the Euclidean distances of user-tag and item-tag separately, which can simultaneously reduce the distance between similar data of user-tag and item-tag and expand the distance between different data.
2) For the problem that it is difficult to distinguish positive and negative samples with fixed margins in the presence of a large number of user-tag and item-tag interactions, an entity pair relationship with angular margin constraints is given to dynamically adjust the distance variation of user-tags and item-tags by constraining the angular margins.
3) For typical pairwise loss function, a new objective function combining the angular margin is established. The angular margin regularizer is added on the basis of the original pairwise loss, and the stability of the model is effectively improved by controlling the angular margin of user-tag and item-tag respectively and adjusting the angular margin strength.
4) Extensive experiments are conducted on two datasets, LastFm and Movielens and the effects of different parameters and components on tag recommendation performance is analyzed. The results show that the proposed method PMLT outperforms the existing baselines.

The remainder of this paper is organized as follows. In Section II, we briefly review the existing recommendation models for tag recommendation and metric learning. In Section III, we propose pairwise metric learning with angular margins for tag recommendation algorithms to predict Top-N tag recommendation lists. In Section IV, we conduct extensive experiments to demonstrate the effectiveness of the proposed approach. In Section V, we conclude the paper.

II. RELATED WORK

A. TAG RECOMMENDATION

Tag recommendation has become an important technique of numerous applications to solve the information overload problem. The aim is to help users describe the items of their interest and provide personalized tag suggestion. Tensor factorization methods have been proposed to perform prediction considering user $\times$ item $\times$ tag interactions, in which the pairwise tensor factorization method PITF has been illustrated to have good recommendation performance. PITF represents each entity pair (e.g., users, items, tags) as a low-dimensional feature vector, and then learn entity relationships using matrix factorization of user-tag and item-tag pairwise interactions. Specifically, the PITF approach represents the scoring function $d(u, i, t)$ through the inner product for different latent vectors.

$$d(u, i, t) = \sum_k \alpha_u t_u + \sum_k \beta_i t_i \tag{1}$$

where k is the dimension of the latent vector and the user, item and tag inputs are embedded in a low-dimensional space. And α_u, t_u, β_i, t_i denote the user features, item features, tag features corresponding to the user and tag features corresponding to the item, respectively. Then, PITF employs the Bayesian personalized ranking (BPR) [30] to estimate the model parameters, and its objective function can be expressed as.

$$J_{BPR}(\Theta) = \sum_{u,i,t,t^- \in V_s} -\ln \delta\left(J_{u,i,t,t^-}(\Theta)\right) + \lambda_\Theta \|\Theta\|^2 \tag{2}$$

where $\Theta = \{\alpha_u, \beta_i, t_u, t_i, t_u^-, t_i^-\}$, $\delta(x)$ denotes the sigmoid function, $J_{u,i,t,t^-}(\Theta) = d(u, i, t) - d\left(u, i, t^-\right)$ and λ_Θ denotes the regularization factor. The gradient of PITF for different parameters is Equation 3.

$$\frac{\partial J_{BPR}(\Theta)}{\partial(\Theta)} = -(1 - \delta(J_{u,i,t,t^-}(\Theta)))\frac{\partial J_{u,i,t,t^-}(\Theta)}{\partial(\Theta)} + 2\lambda_\theta \Theta \tag{3}$$

where the derivative of the potential factor corresponding to the equation $J_{u,i,t,t^-}(\Theta)$ can be expressed as.

$$\frac{\partial J_{u,i,t,t^-}(\Theta)}{\partial \Theta} = \begin{cases} t_u - t_u^-, & \text{if } \Theta = \alpha_u \\ t_i - t_i^-, & \text{if } \Theta = \beta_i \\ \alpha_u, & \text{if } \Theta = t_u \\ \beta_i, & \text{if } \Theta = t_i \\ -\alpha_u, & \text{if } \Theta = t_u^- \\ -\beta_i, & \text{if } \Theta = t_i^- \end{cases} \tag{4}$$

Tensor factorization has been widely applied to tag recommendation and PITF [10] proposed by Randle et al is an effective method. Compared with HOSVD [12] and RTF [13], PITF has linear time complexity and capture pairwise interactions among users, tags, and items. Different from its linear complexity, Fang et al. proposed the NITF method [15] to extend the capacity of the model using Gaussian kernel functions, which is regarded as a nonlinear extension of typical methods and also can work well with a small number of features. Cai et al. proposed the LOTD method [31], using point-by-point regression methods to learn from observed tagging data, and it enhances the interaction among users, items, and tags using low-order polynomials. Lu et al. developed a post-based collaborative filtering method [32] based on a ternary social tag network. To capture higher-order collaborative signals in entity interactions, Yu et al. applied graph networks to PITF to aggregate neighbor information from multiple layers to generate the final representation of entity pairs [33]. Wang et al. extended PITF by adding weights to user-tag interactions and item-tag interactions separately considering both temporal factors and personalization [34]. Tensor factorization methods have good decomposition effect and can handle user-tag and item-tag interactions separately, predicting tag-ranked lists based on users' historical behaviors. In addition, deep neural networks (DNN) have been used for tag recommendation [5], [35], [36] to mine the entity relationships of hidden information in data, and these methods improve the performance of traditional tag recommendation algorithms due to the effective learning capability of DNN. Although tensor factorization and deep neural network methods have been shown to be effective for modeling entity information, they do not satisfy the triangle inequality and cannot capture the fine-grained preference information.

B. METRIC LEARNING FOR RECOMMENDATION MODELS

Metric learning methods have been successfully applied to Top-N recommendation tasks. Prior researchers concentrated on studying the user-item distance, and these methods compute the similarity scores of users and items in terms of Euclidean distance and use pairwise loss to distinguish positive items from negative items in terms of the difference in distance. For instance, CML [24] proposes the mechanism using pairwise loss, which can overcome the constraint that dot product in MF methods could not satisfy triangle inequality. It reduces the distance between users and positive items and keeps users away from those negative items that are not of interest. In item recommendation, as a typical ML method, CML scoring function can be expressed as Equation 5.

$$d(u, i) = \|\alpha_u - \beta_i\|_2^2 \tag{5}$$

where $\| * \|_2$ denotes the L2 norm and CML uses pairwise loss as the objective function to ensure that the distance between the user α_u and the negative term β_i^- is greater than the distance between the current user α_u and the positive term β_i.

$$J(\Theta) = \sum_{(u,i)\in D_s} \sum_{(u,i^-)\notin D_s} w_{u,i}[d(u, i) - d(u, i^-) + m]_+ \tag{6}$$

where $[x]_+ = \max(x, 0)$ denotes the pairwise loss, also known as the Relu function, D_s is the trained sample, $w_{u,i}$ denotes the weight of the pairwise ranking loss of penalized users and positive items in lower ranks, and m is the fixed margin. Then CML performs the derivative of the parameter $\Theta = \left\{\alpha_u, \beta_i, \beta_i^-\right\}$ update as

$$\frac{\partial J(\Theta)}{\partial \Theta} = \begin{cases} 2\left(\beta_i^- - \beta_i\right), & \text{if } \Theta = \alpha_u \\ 2\left(\beta_i - \alpha_u\right), & \text{if } \Theta = \beta_i \\ 2\left(\alpha_u - \beta_i^-\right), & \text{if } \Theta = \beta_i^- \end{cases} \tag{7}$$

To alleviate the problems of geometric inflexibility and fixed margins of CML that make it difficult to distinguish between positive and negative samples, researchers have proposed a number of ML variants of these methods that improve recommendation performance. Inspired by the translation mechanism [37], Tay et al used a memory-based attention network to generate a user-item latent relationship vector

LRML [25]. Park et al proposed the TransCF method [26] to construct a user-item specific transformation vector using the neighborhood information of users and items, and then transform each user into an item based on the users' relationship to those items. TransCF and LRML aim to address the problem of metric learning geometric inflexibility. Compared with CML, the latent relationship metric learning approach improves the user-item distance measurements. In order to alleviate the limitation of fixed margin, Li et al. proposed an adaptive strategy SML [27], which can effectively adjust the size of user-centric and item-centric fixed margins. Then, the SML was optimized for pairwise loss with different margin sizes. In the optimization process, SML adaptively determines the user-item fixed margin size and also adjusts the distance between positive items and negative items. Ma et al. proposed an adaptive margin generation scheme PMLAM [28], which utilizes a neural network based on the Wasserstein distance [38], where both users and items are parameterized using a Gaussian distribution. SML and PMLAM aim to find an adaptive strategy that adaptively learns the size of the different margins between users, positive items and negative items. However, SML and PMLAM still use a pairwise loss paradigm to measure the relative distance between a given user and item, and these methods continue to be tuned on a fixed margin basis. In addition, existing ML methods focus mainly on item recommendation tasks and less attention are paied to distance relations involving user, item, and tag relationships. In contrast, our work provides a new attempt for tag recommendation by reconstructing the ternary relationship of entities using metric learning to improve the performance of tag recommendation.

III. THE PROPOSED METHOD

A. PROBLEM STATEMENTS

The function of the tag recommendation is to recommend a series of tags for users to mark items of interest. The historical tagging information of the user × item × tag triplet can be represented by a set of ternary relations.

$$S = (U, I, T) \tag{8}$$

where U, I, and T are the set of users $U = \{u_1, \cdots, u_i, \cdots, u_{|U|}\}$, the set of items $I = \{i_1, \cdots, i_i, \cdots, i_{|I|}\}$, and the set of tags $T = \{t_1, \cdots, t_i, \cdots, t_{|T|}\}$, respectively. $|U|$ represents the number of users, $|I|$ represents the number of items, and $|T|$ represents the number of tags.

From the set of ternary relations S, a third-order tensor $D \in \mathbb{R}^{|U|\times|I|\times|T|}$ can usually be defined, where the elements in the third-order tensor D can be defined as follows.

$$d(u, i, t) = \begin{cases} 1, & (u, i, t) \in S \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

If a user tagging an item with a tag indicates a positive instance, the element of D is $d(u, i, t) = 1$. The rest of the data is a mixture of negative instances and missing values, and the element of D is $d_{u,i,t} = 0$.

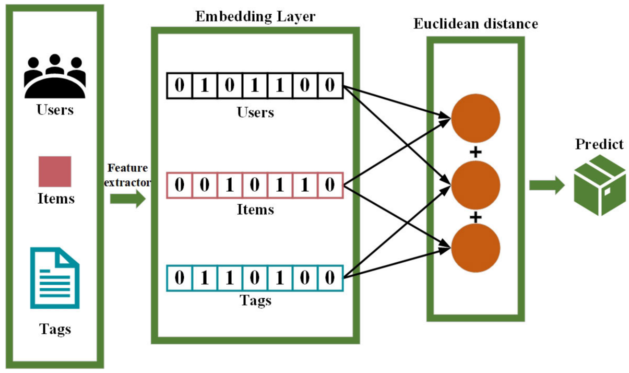


FIGURE 1. An illustration of pairwise distance relations for the user × item × tag triplet.

Tag recommendation is intended to present the top N tags for users to mark favorite items, and after predicting the scores $d_{u,i,t}$ of all candidate tags t, the top N tags ranked list can be represented as Equation 10.

$$\hat{T}_{u,i} = \underset{t\in T}{\arg\max} d(u, i, t) \tag{10}$$

where N denotes the number of recommended tags in the tag list.

B. PAIRED METRIC LEARNING LABEL RECOMMENDATION MODEL BASED ON ANGULAR MARGIN

1) PAIRWISE DISTANCES FOR USERS, ITEMS AND TAGS

As shown in Figure 1, for tag recommendation task, we propose a pairwise metric learning method to construct pairwise distance relations for the user × item × tag triplet. The key point involves projecting users, items and tags into a low-dimensional space and calculating similarity scores between user-item, user-tag and item-tag using Euclidean distances. The scoring functions can be defined as Equation 11.

$$d(u, i, t) = \|\alpha_u - \beta_i\|_2^2 + \|\alpha_u - t_u\|_2^2 + \|\beta_i - t_i\|_2^2 \tag{11}$$

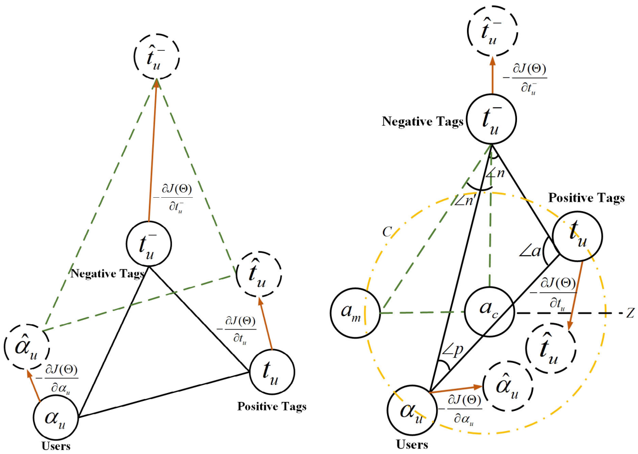
Notably, due to the optimization using pairwise loss, which eliminates the user-item distance metric, the final result can be expressed as.

$$d(u, i, t) = \|\alpha_u - t_u\|_2^2 + \|\beta_i - t_i\|_2^2 \tag{12}$$

Given $m > 0$, the distance among user u, item i and negative tag t^- needs to be ensured to exceed the distance among current user u, item i and positive tag t.

$$d(u, i, t) + m \leq d\left(u, i, t^-\right) \tag{13}$$

where $d\left(u, i, t^-\right) = \left\|\alpha_u - t_u^-\right\|_2^2 + \left\|\beta_i - t_i^-\right\|_2^2$, t_u^- and t_i^- denote the unobserved negative tags. Then, we learn the pairwise distances of user-tag and item-tag simultaneously, using pairwise loss to reduce the distance between user-tag and item-tag similar data and to expand the distance from different data, and the final objective function can be



(a) (b)

FIGURE 2. **(a): Illustration of the pairwise loss and its gradient. (b): Illustration of the angular constraint.**

expressed as Equation 14.

$$J(\Theta)=\sum_{(u,i,t)\in V_s}\sum_{(u,i,t^-)\notin V_s}\left[d(u,i,t)-d\left(u,i,t^-\right)+m\right]_+ \tag{14}$$

where V_s denotes the training data, and since the user prefers to observe (u, i, t) rather than (u, i, t^-), the training data V_s can be defined as

$$V_s=\left\{\left(u,i,t,t^-\right)\mid (u,i,t)\in S\wedge\left(u,i,t^-\right)\notin S\right\} \tag{15}$$

where (u, i, t) denotes the observed tags and (u, i, t^-) denotes the tags that are not observed.

Then the gradients of the different parameters $\Theta = \left\{\alpha_u, \beta_i, t_u, t_i, t_u^-, t_i^-\right\}$ are calculated separately. If $d(u, i, t) + m \leq d\left(u, i, t^-\right)$, their derivative values are equal to zero. Otherwise, the derivatives of the different parameters are updated as.

$$\frac{\partial J(\Theta)}{\partial\Theta}=\begin{cases}2\left(t_u^- - t_u\right), & \text{if}\Theta=\alpha_u\\ 2\left(t_i^- - t_i\right), & \text{if }\Theta=\beta_i\\ -2\left(\alpha_u-t_u\right), & \text{if }\Theta=t_u\\ -2\left(\beta_i-t_i\right), & \text{if}\Theta=t_i\\ 2\left(\alpha_u-t_u^-\right), & \text{if}\Theta=t_u^-\\ 2\left(\beta_i-t_i^-\right), & \text{if }\Theta=t_i^-\end{cases} \tag{16}$$

The pairwise distance relationship between user-tag and item-tag is modeled by using a metric learning approach. Finally, the optimization iterations are performed using stochastic gradient descent. $\Theta \leftarrow \Theta - \eta \times \frac{\partial J(\Theta)}{\partial \Theta}$, where η denotes the step size.

2) ANGULAR MARGIN CONSTRAINTS FOR USERS, ITEMS AND TAGS

Metric learning uses pairwise loss to distinguish positive and negative samples with a fixed margin, as shown in Figure 2(a). However, when there are a large number of user-tag and item-tag interactions, the user and the item may wrongly recommend tags that are not of interest to the user at a distance close to each other, generating inaccurate results. Inspired by machine vision studies on angular constraint [39], [40], [41], as shown in Figure 2(b), we encode the relationship among user α_u, positive tag t_u and negative tag t_u^- according to the angle at the negative tag, which together form a triangle.

$$\angle n+\angle a+\angle p=180^\circ \tag{17}$$

In order to keep the negative tag t_u^- as far away from the user α_u as possible, according to the cosine rule, $\angle n$ has to be the smallest angle, $\angle n \leq \min(\angle a, \angle p)$, and $\angle n$ has to be less than 60°. The upper bound of $\angle n$ ensures that the user and the positive tag are close.

However, when $\angle a > 90°$, although the margin of $\angle n$ is reduced, the negative tag t_u^- may be dragged to α_u, resulting in the distance of the negative tag t_u^- being closer to the user α_u. Inspired by the literature [39], we plan to define a new triangular relation about the user, the positive tag and the negative tag. Based on the geometrically symmetric structure of metric learning, we first define a local sample distribution using an approximate circle C passing through α_u and t_u, with a_c in the middle as

$$a_c=\frac{\alpha_u+t_u}{2} \tag{18}$$

A new triangular relationship is defined by introducing a hyperplane Z that intersects the circle C at two nodes, one node is denoted as a_m. Based on these auxiliary structures, we define a new triangular relationship by shifting the user α_u and the positive tag t_u to a_c and a_m separately. Given a predefined angular margin φ, the user-tag angular margin as follows.

$$\tan\angle n'=\frac{\|a_m-a_c\|}{\|t_u^- - a_c\|}=\frac{\|\alpha_u-t_u\|}{2\|t_u^- - (\alpha_u+t_u)/2\|}\leq\tan\varphi \tag{19}$$

where $a_m - a_c$ is the radius of the circle C, it is equal to $(\alpha_u - t_u)/2$. By applying constraints on the angular margin we can achieve dynamic adjustment of the user and tag distances. For the item-tag, we also make the same angular margin constraint can be expressed as Equation 20.

$$\frac{\|\beta_i-t_i\|}{2\|t_i^- - (\beta_i+t_i)/2\|}\leq\tan\varphi \tag{20}$$

Finally, the angular margin constraint relationship between user-tag and item-tag is considered on the basis of pairwise loss, and the size of the angular margin is defined to satisfy the following relationship.

$$\|\alpha_u-t_u\|_2^2\leq 4\tan^2\varphi\|t_u^- - \frac{\alpha_u+t_u}{2}\|_2^2 \tag{21}$$

$$\|\beta_i-t_i\|_2^2\leq 4\tan^2\varphi\|t_i^- - \frac{\beta_i+t_i}{2}\|_2^2 \tag{22}$$

where $\alpha_u - t_u$ and $\beta_i - t_i$ are required to satisfy the constraint of angular margin φ simultaneously. Thus, the objective function of the angular margin considering user-tag and item-tag based on pairwise loss can be expressed as Equation 23.

$$J(\Theta) = \sum_{(u,i,t)\in V_s} \sum_{(u,i,t^-)\notin V_s} [d(u,i,t) - 4\tan^2\varphi \| t_u^- - \frac{\alpha_u + t_u}{2} \|_2^2 - 4\tan^2\varphi \| t_i^- - \frac{\beta_i + t_i}{2} \|_2^2]_+ \tag{23}$$

Compared to the original gradient with pairwise loss, the gradient constrained by the angular margin is more robust, because the distance variation of different entity pairs is considered simultaneously and produces a positive effect. The parameter updating process of angular margin relative to user, item and tag can be expressed as Equation 24.

$$\frac{\partial J(\Theta)}{\partial \Theta} = \begin{cases} 2(\alpha_u - t_u) - 2\tan^2\varphi \left(\alpha_u + t_u - 2t_u^-\right), \\ \text{if } \Theta = \alpha_u \\ 2(\beta_i - t_i) - 2\tan^2\varphi \left(\beta_i + t_i - 2t_i^-\right), \\ \text{if } \Theta = \beta_i \\ -2(\alpha_u - t_u) - 2\tan^2\varphi \left(\alpha_u + t_u - 2t_u^-\right), \\ \text{if } \Theta = t_u \\ -2(\beta_i - t_i) - 2\tan^2\varphi \left(\beta_i + t_i - 2t_i^-\right), \\ \text{if } \Theta = t_i \\ 4\tan^2\varphi \left(\alpha_u + t_u - 2t_u^-\right), \\ \text{if } \Theta = t_u^- \\ 4\tan^2\varphi \left(\beta_i + t_i - 2t_i^-\right), \\ \text{if } \Theta = t_i^- \end{cases} \tag{24}$$

This method improves the limitations imposed by fixed margin, and the distance variation of different entity pairs can be dynamically adjusted by the constraint of angular margins.

3) OBJECTIVE FUNCTION BASED ON PAIRWISE METRIC LEARNING WITH ANGULAR MARGIN FOR TAG RECOMMENDATION

The fixed margin limits the expressiveness of the model, especially when the data distribution is complex. In Equation 19 and Equation 20, we define constraints on the angular margin of user-tag and item-tag, enabling the exclusion of tags that are not relevant to the user and item when learning the pairwise distance relationship between user-tag and item-tag. Ultimately, the objective function for pairwise metric learning with the angular margin is determined as.

$$J(\Theta) = \sum_{(u,i,t)\in V_s} \sum_{(u,i,t^-)\notin V_s} [d(u,i,t) - d(u,i,t^-) + m]_+ + \gamma \sum_{(u,i,t)\in V_s} \sum_{(u,i,t^-)\notin V_s} [d(u,i,t) - 4\tan^2\varphi \| t_u^- - \frac{\alpha_u + t_u}{2} \|_2^2 - 4\tan^2\varphi \| t_i^- - \frac{\beta_i + t_i}{2} \|_2^2]_+ \tag{25}$$

where γ controls the strength of the angular margin regularizer and φ is the size of the angular margin. In the following Algorithm 1, we introduce the whole program procedure.

In Algorithm 1, we describe the whole working process of the tag recommendation method. Given embeddings of users, items and tags, they are multiplied by the embedding weights U, I, T^u and T^i, respectively, in which the embedding weights satisfy a normal distribution (line 1). Next, positive samples (u, i, t) are sampled from the training instances V_s (line 3), and negative samples (u, i, t^-) are randomly sampled in the remaining data $(u, i, t)\backslash V_s$ (line 4). Then, the gradients of the model parameters $\Theta = \{\alpha_u, \beta_i, t_u, t_i, t_u^-, t_i^-\}$ are calculated separately (lines 5-12) until convergence and the updated parameters are returned (line 14). The similarity score of the entities is calculated according to Equation 12. Finally, a Top-N list of tag recommendations is returned according to Equation 10.

Algorithm 1 Pairwise Metric Learning With Angular Margin for Tag Recommendation

Input: Training dataset V_s, the embedding matrices U, I, T_u, T_i, the hyperparameters γ and φ, the number of latent feature dimensions k

Output: Model parameters $\Theta = \{\alpha_u, \beta_i, t_u, t_i, t_u^- t_i^-\}$

1: Initialize U, I, T^u, T^i from $N(\mu, \sigma^2)$
2: **repeat**
3: **for** (u, i, t) in V_s **do**
4: Randomly draw (u, i, t^-) from $(u, i, t)\backslash V_s$
5: Updating $\alpha_u, \beta_i, t_u, t_i, t_u^-, t_i^-$
6: $\alpha_u \leftarrow \alpha_u - \eta(2(t_u^- - t_u) + \gamma(2(\alpha_u - t_u) - 2\tan^2\varphi(\alpha_u + t_u - 2t_u^-)))$
7: $\beta_i \leftarrow \beta_i - \eta(2(t_i^- - t_i) + \gamma(2(\beta_i - t_i) - 2\tan^2\varphi(\beta_i + t_i - 2t_i^-)))$
8: $t_u \leftarrow t_u - \eta(-2(\alpha_u - t_u) + \gamma(-2(\alpha_u - t_u) - 2\tan^2\varphi(\alpha_u + t_u - 2t_u^-)))$
9: $t_i \leftarrow t_i - \eta(-2(\beta_i - t_i) + \gamma(2(\beta_i - t_i) - 2\tan^2\varphi(\beta_i + t_i - 2t_i^-)))$
10: $t_u^- \leftarrow t_u^- - \eta(2(\alpha_u - t_u^-) + \gamma(4\tan^2\varphi(\alpha_u + t_u - 2t_u^-)))$
11: $t_i^- \leftarrow \beta_i^- - \eta(2(\beta_i - t_i^-) + \gamma(4\tan^2\varphi(\beta_i + t_i - 2t_i^-)))$
12: **end for**
13: **until** converge
14: **return** $\Theta = \{\alpha_u, \beta_i, t_u, t_i, t_u^-, t_i^-\}$

IV. EXPERTMENTS

In this section, we first describe the settings of the experiment, including datasets, experimental details, evaluation methods, and baselines. Then, we conducted extensive experiments to answer the following research questions.

RQ1: What is the recommendation performance of the proposed PMLT method? How competitive is it compared to baselines and the adjusted state-of-the-art metric learning methods?

RQ2: What is the impact of different embedding dimensions k on recommendation performance?

RQ3: How do hyperparameters affect recommendation performance, and how to choose the optimal value?

RQ4: What are the contributions of different components of PMLT?

TABLE 1. Details of datasets used in our experiments.

Dataset	Users	Items	Tags	Train samples	Test samples
LastFm	966	3870	1204	105056	28889
Movielens	990	3247	2566	50384	11304

RQ5: What are the effects of different fixed margins and different iterations?

A. DATASETS

To demonstrate the effectiveness of the proposed method, we conducted extensive experiments on two publicly available datasets, LastFm dataset (http://www.last.fm) and Movielens dataset (http://www.grouplens.org). The statistics of the different datasets are summarized in Table 1. The datasets are preprocessed to obtain its p-core, which is the probability of p occurrences per user per item and per tag. The Movielens dataset is 5-core and for LastFm is the denser dataset 10-core.

B. EXPERIMENTAL DETAILS AND EVALUATION METHODS

1) EXPERIMENTAL DETAILS

We implement all baseline methods (CF, PITF, NITF, ABNT, CML, LRML, SMLT, PMLT) utilizing a Tensorflow framework in a Linux environment, and all the methods are optimized using the Adam optimizer [42] with learning rates tuned in {0.001, 0.05, 0.01}. To maintain comparison consistency, the dimension of embedding is fixed at 64 and the batch size is 1024. The fixed margin is 0.3, the regularization factor is 0.001, and all latent vectors (e.g. $\alpha_u, \beta_i, t_u, t_i, t_u^-, t_i^-$) are randomly initialized using a normal distribution (the mean is 0 and standard variance is 0.01).

2) EVALUATION METHODS

For each user, we randomly select a post (u, i) and move the triple relevant to the selected post from S to S_{test}. The remaining observed user-item tagged triples are treated as the training set $S_{\text{train}} = S \backslash S_{\text{test}}$. Similar to the classical item recommendation task, tag recommendation recommends a list of N top-ranked tags for user-item (u, i). To evaluate the accuracy and quality of the ranking, in our experiments, we use two widely used evaluation metrics for ranking to measure the tag recommendation performance of all comparison methods, Recall@N and NDCG@N.

$$\operatorname{Re} c@N = \frac{1}{|P_{S_{\text{test}}}|} \sum_{(u,i)\in P_{S_{\text{test}}}} \frac{|\operatorname{Top}(u, i, N) \cap (u, i, t) \in S_{\text{test}}|}{|\{t \mid (u, i, t) \in S_{\text{test}}\}|} \tag{26}$$

$$NDCG@N = \frac{DCG@N}{IDCG@N} \tag{27}$$

where N denotes the length of the ranked tag recommendation list, $DCG@N = \text{rel}(1) + \sum_{i=1}^{N} \frac{\text{rel}(i)}{\log_2(i)}$, P_{test} denotes the number of posts in the test set, and IDCG is the ideal discounted cumulative gain.

C. BASELINE ALGORITHM

We evaluate the proposed method PMLT in comparison with the following baselines. Table 2. summarizes the scoring functions and loss functions used in all baselines.

CF [7]: it factorizes the training matrix into two low-rank matrices and recovers the original matrix by the inner product of these matrices, which uses only the matrix information of the user and the tag.

PITF [10]: it explicitly models the pairwise interactions between the user $\times$ item $\times$ tag triplet and aims to return a list of the top N ranked tags for a particular user-item pair.

NITF [15]: it is a nonlinear tensor factorization model that uses Gaussian radial basis functions to enhance the capacity of the model.

ABNT [16]: it is a nonlinear relationship using a multilayer perceptron to model the interaction among users, items and tags.

CMLT [24]: it uses Euclidean distance to calculate the distance relationship among users, items and tags, considering both user-tag and item-tag Euclidean distances.

LRMLT [25]: it is a latent relationship metric learning method that uses a memory attention network to induce latent relationship vectors for user-tag and item-tag.

SMLT [27]: it is a metric learning algorithm with adaptive margin that measures both positive and negative tag distances.

PMLT: The method proposed in this paper, it is a pairwise metric learning method based on angular margin. Pairwise metric learning can learn both user-tag and item-tag distances, and the distance variation of user-tag and item-tag can be dynamically adjusted by angular margin constraints.

D. COMPARISON OF RESULTS

We conduct extensive experiments on the LastFm and Movielens datasets. For the baselines, we follow the hyperparameter settings reported in the corresponding literature, and if hyperparameters are not reported in the literature, we perform fine-tuning to obtain the results for all baselines under the best hyperparameters. Among them, CF, PITF, NITF, and ABNT are tensor factorization methods, and CMLT, LRMLT, SMLT, and PMLT are metric learning methods. Table 3. and Table 4. shows the comparison results of proposed methods and baselines.

From the results, we can find that PMLT outperforms CF, PITF, NITF and ABNT. This indicates that the metric learning based approach overcomes the inherent limitation of tensor factorization methods using inner products that do not satisfy the triangle inequality. The tensor factorization methods CF, PITF, NITF and ABNT predict scores by the inner product of two interactions of user, item and tag and employ Bayesian personalized ranking (BPR) for optimization. The proposed

TABLE 2. Scoring functions and loss function comparison with baselines. $\omega = 1/\sigma^2$, σ is the standard deviation of the Gaussian distribution. ϕ_x denotes the activation function. $r_{\alpha u, t_u}$ and r_{β_i, t_i} denote the latent relationship vectors for user-tag and item-tag, respectively. m_u, m_i, n_{tu}, and n_{ti} denote the adaptive margin. For user-tag, $d(t, t^-) = \|t_u - t_u^-\|_2^2$. For item-tag, $d(t, t^-) = \|t_i - t_i^-\|_2^2$.

Baseline	Scoring Function	Loss function
CF	$\mathrm{d(u,i,t)} = \alpha_\mathrm{u}\mathrm{t_u}$ or $\mathrm{d(u,i,t)} = \beta_\mathrm{i}\mathrm{t_i}$	
PITF	$\mathrm{d(u,i,t)} = \alpha_\mathrm{u}\mathrm{t_u} + \beta_\mathrm{i}\mathrm{t_i}$	
NITF	$\mathrm{d(u,i,t)} = \exp(-\frac{1}{2}\omega\|\alpha_\mathrm{u} - \mathrm{t_u}\|^2) + \exp(-\frac{1}{2}\omega\|\beta_\mathrm{i} - \mathrm{t_i}\|^2)$	$\mathrm{J}(\Theta) = \sum_{\mathrm{(u,i,t)\in V_s}} \sum_{\mathrm{(u,i,t^-)\notin V_s}} -\ln\delta(\mathrm{J_{u,i,t,t^-}}(\Theta)) + \lambda_\Theta\|\Theta\|^2$
ABNT	$\mathrm{d(u,i,t)} = \phi_{\mathrm{last}}^{\mathrm{u,t}}(\cdots\phi_2^{\mathrm{u,t}}(\phi_1^{\mathrm{u,t}}(\alpha_\mathrm{u},\mathrm{t_u}))\cdots) + \phi_{\mathrm{last}}^{\mathrm{i,t}}(\cdots\phi_2^{\mathrm{i,t}}(\phi_1^{\mathrm{i,t}}(\beta_\mathrm{i},\mathrm{t_i}))\cdots)$	
CMLT	$\mathrm{d(u,i,t)} = \|\alpha_\mathrm{u} - \mathrm{t_u}\|_2^2 + \|\beta_\mathrm{i} - \mathrm{t_i}\|_2^2$	$\mathrm{J}(\Theta) = \sum_{\mathrm{(u,i,t)\in V_s}} \sum_{\mathrm{(u,i,t^-)\notin V_s}} [\mathrm{d(u,i,t)} - \mathrm{d(u,i,t^-)} + \mathrm{m}]_+$
LRMLT	$\mathrm{d(u,i,t)} = \|\alpha_\mathrm{u} + \mathrm{r}_{\alpha_\mathrm{u},\mathrm{t_u}} - \mathrm{t_u}\|_2^2 + \|\beta_\mathrm{i} + \mathrm{r}_{\beta_\mathrm{i},\mathrm{t_i}} - \mathrm{t_i}\|_2^2$	
SMLT	$\mathrm{d(u,i,t)} = \|\alpha_\mathrm{u} - \mathrm{t_u}\|_2^2 + \|\beta_\mathrm{i} - \mathrm{t_i}\|_2^2$	$\mathrm{J}(\Theta) = \sum_{\mathrm{(u,i,t)\in V_s}} \sum_{\mathrm{(u,i,t^-)\notin V_s}} ([\mathrm{d(u,t)} - \mathrm{d(u,t^-)} + \mathrm{m_u}]_+ + [\mathrm{d(i,t)} - \mathrm{d}(\mathrm{i,t^-}) + \mathrm{m_i}]_+ + \lambda_1([\mathrm{d(u,t)} - \mathrm{d(t,t^-)} + \mathrm{n_{t^u}}]_+ + [\mathrm{d(i,t)} - \mathrm{d(t,t^-)} + \mathrm{n_{t^i}}]_+) - \lambda_2(\frac{1}{\mathrm{\|U\|}}\sum_\mathrm{u}\mathrm{m_u} + \frac{1}{\mathrm{\|T\|}}\sum_\mathrm{t}\mathrm{n_{t^u}} + \frac{1}{\mathrm{\|I\|}}\sum_\mathrm{i}\mathrm{m_i} + \frac{1}{\mathrm{\|T\|}}\sum_\mathrm{t}\mathrm{n_{t^i}}))$
PMLT	$\mathrm{d(u,i,t)} = \|\alpha_\mathrm{u} - \mathrm{t_u}\|_2^2 + \|\beta_\mathrm{i} - \mathrm{t_i}\|_2^2$	$\mathrm{J}(\Theta) = \sum_{\mathrm{(u,i,t)\in V_s}} \sum_{\mathrm{(u,i,t^-)\notin V_s}} [\mathrm{d(u,i,t)} - \mathrm{d(u,i,t^-)} + \mathrm{m}]_+ + \gamma\sum_{\mathrm{(u,i,t)\in V_s}} \sum_{\mathrm{(u,i,t^-)\notin V_s}} [\mathrm{d(u,i,t)} - 4\tan^2\varphi\|\mathrm{t_u^-} - \frac{\alpha_\mathrm{u}+\mathrm{t_u}}{2}\|_2^2 - 4\tan^2\varphi\|\mathrm{t_i^-} - \frac{\beta_1+\mathrm{t_i}}{2}\|_2^2]_+$

method PMLT is based on the metric learning method, which constructs the pairwise distance relationship between user-tag and item-tag to calculate the similarity score, and uses pairwise loss to reduce the distance between the same categories and expand the distance between different categories. The proposed method can effectively capture the distance relationship among entity pairs and has better recommendation quality.

Our proposed method significantly outperforms all baseline methods in terms of Recall@N and NDCG@N evaluation metrics on both datasets. There is a significant performance improvement over the most competitive CMLT method. These results illustrate that CMLT distinguishes positive and negative samples with a fixed margin. When there are a large number of user-tag and item-tag interactions, CMLT may incorrectly recommend tags that are not of interest to users, reducing the recommendation quality. The proposed method PMLT adds an extra angular margin regularizer, which can adjust the size of the angular margin and the strength of the angular margin regularizer, and helps to improve the flexibility of the recommendation model.

Compared with latent relationship metric learning LRMLT and adaptive margin method SMLT, we can see that PMLT still have significant improvement. This indicates that PMLT provides important potential for metric learning approaches as it alleviates the problems of geometric inflexibility of metric learning and the difficulty of distinguishing between positive and negative samples with fixed margins to some extent.

E. DIFFERENT EMBEDDING DIMENSIONS AND EFFECT OF HYPERPARAMETERS

1) EFFECT OF DIFFERENT EMBEDDING DIMENSIONS

The embedding dimension k is an important parameter that affects the performance of the proposed method PMLT. In this section, we define the value of k as $k = \{8, 16, 32, 64, 128, 256, 512, 1024\}$ and also study the impact of the embedding dimension k on the quality of tag recommendation.

As can be seen from Table 5. and Table 6., the recommendation performance of PMLT on the LastFm and Movielens datasets improves with continuous increase in k values first, and the best performance is achieved at $k = 512$ and $k = 256$ for Recall@5 and NDCG@5 on the Movielens and LastFm datasets, respectively. However, larger k values (e.g., $k = 1024$) do not guarantee improved recommendation performance, and this observation suggests that a continuous increase in the latent dimension k does not improve the recommendation quality. The embedding representation will become sparse, resulting in overfitting phenomena and lead to further performance degradation. Despite the fact that 512 and 256 will have positive results, we chose 64 as our final embedding dimension as larger embedding size would increase training time and space.

TABLE 3. Performance comparisons on tag ranking task in terms of Recall@N and NDCG@N on LastFm dataset.

Evaluation Metric	CF	PITF	NITF	ABNT	CMLT	LRMLT	SMLT	PMLT	Improve
Recall@3	0.21032	0.33289	0.35263	0.19218	<u>0.37893*</u>	0.37741	0.29755	**0.38793**	2.3%
Recall@5	0.29243	0.45453	0.49164	0.27412	<u>0.50992*</u>	0.507424	0.39749	**0.52594**	3.1%
Recall@10	0.41680	0.61342	0.67628	0.41434	<u>0.69456*</u>	0.68961	0.52823	**0.70799**	1.9%
Recall@20	0.57285	0.75326	0.83364	0.57070	<u>0.84357*</u>	0.83918	0.64142	**0.85223**	1%
NDCG@3	0.15865	0.24923	0.26006	0.14239	<u>0.28337*</u>	0.28267	0.22680	**0.29130**	2.7%
NDCG@5	0.19224	0.29920	0.31725	0.17606	<u>0.33747*</u>	0.33626	0.26775	**0.34803**	3.1%
NDCG@10	0.23232	0.35085	0.37695	0.22099	<u>0.39710*</u>	0.39600	0.31043	**0.40602**	2.2%
NDCG@20	0.26654	0.38623	0.41695	0.25972	<u>0.43481*</u>	0.43386	0.33890	**0.44197**	1.6%

TABLE 4. Performance comparisons on tag ranking task in terms of Recall@N and NDCG@N on Movielens dataset.

Evaluation Metric	CF	PITF	NITF	ABNT	CMLT	LRMLT	SMLT	PMLT	Improve
Recall@3	0.16065	<u>0.31060*</u>	0.30034	0.14782	0.30679	0.30706	0.23045	**0.32475**	4.5%
Recall@5	0.20656	0.38217	0.38588	0.19542	<u>0.39853*</u>	0.39340	0.29140	**0.41109**	3.1%
Recall@10	0.28901	0.48549	0.49531	0.26637	<u>0.51327*</u>	0.50672	0.36377	**0.52123**	1.5%
Recall@20	0.37491	0.58227	0.60244	0.35448	<u>0.61916*</u>	0.61553	0.44170	**0.62358**	0.7%
NDCG@3	0.12547	<u>0.24262*</u>	0.22552	0.11403	0.23314	0.23178	0.18111	**0.24628**	1.5%
NDCG@5	0.14406	<u>0.27200*</u>	0.26062	0.13346	0.27155	0.26714	0.20611	**0.28243**	3.8%
NDCG@10	0.17049	0.30561	0.29630	0.15503	<u>0.30815*</u>	0.30409	0.22937	**0.31821**	3.2%
NDCG@20	0.19217	0.33008	0.32341	0.17665	<u>0.33488*</u>	0.33165	0.24936	**0.34404**	2.7%

2) EFFECT OF HYPERPARAMETERS

PMLT introduces an additional two hyperparameters γ and φ that control the magnitude of the angular margin and the strength of the angular margin regularizer respectively. We demonstrate here how these two hyperparameters affect the performance and illustrate to set them. Due to space constraints, we report the display results of Recall@5 and NDCG@5 on Movielens and LastFm.

First, we fix φ at 40 on the Movielens dataset and 50 on the LastFm dataset. Subsequently, we change the value of γ in different datasets, which is shown in Figure 3(a) and Figure 3(b), and the optimal values are around $\gamma = 0.6$ and $\gamma = 2$ for the Movielens and LastFm datasets respectively. Furthermore, we observe that continuing to increase γ, after $\gamma > 2$ leads to a gradually decreasing performance for both datasets, which indicates that the larger the value of γ will disrupt the learning process. Therefore, it is suggested that γ should be set to a value of 2 or lower.

Secondly, we fix γ to the default values of 0.6 and 2 in the Movielens and LastFm datasets respectively, and change φ. According to the results of Recall@5 and NDCG@5 shown in Figure 3(c) and Figure 3(d), we observe that the optimal values for Movielens and LastFm are approximately 40 and 50, respectively. As the value of φ continues to increase, the performance of PMLT gradually improves. The results indicate that the angular margin provides additional constraints on the distance of entity pairs to ensure that dissimilar tags can be separated from users and items. Furthermore, the angular margin will not be able to distinguish the distance relationship of entity pairs when φ is too large (e.g., $\varphi > 50$), and the performance starts to degrade. Therefore, we recommend that φ takes values between $30 < \varphi < 50$.

F. ABLATION EXPERIMENTS

To clarify the impact of different components of PMLT on performance, we analyzed the contribution of each component of PMLT, including considering separately the relationship between user and tag UT, the relationship between item and tag IT, and the pairwise distance relationship among user, item and tag CMLT. And AugularMLT represents the model that only considers the constraint relation of angular margin, while PMLT is the total model. While keeping all hyperparameters are optimally set, the results on different data sets are as follows.

As shown in Figure 4, UT and IT perform relatively poorly on both datasets, which indicates that the optimal performance cannot be obtained by considering only 2-dimensional relationships for user-tag or item-tag. However, CMLT and PMLT consider the pairwise distance relationship of user-tag and item-tag at the same time, which is conducive to capture the relationship among user, item and tag. Comparison of AugularMLT and CMLT, PMLT shows that combining pairwise metric learning and angular margin is more advantageous than using one or the other alone.

TABLE 5. The different embedding dimensions in terms of Recall@N and NDCG@N on LastFm dataset.

Embedding size	Recall@3	Recall@5	Recall@10	Recall@20	NDCG@3	NDCG@5	NDCG@10	NDCG@20
8	0.29925	0.40971	0.56530	0.70217	0.22365	0.26901	0.31941	0.35407
16	0.34383	0.46751	0.63377	0.77774	0.25647	0.30718	0.36086	0.39759
32	0.37319	0.50040	0.68061	0.82197	0.27953	0.33126	0.38957	0.42555
64	0.38793	0.52594	0.70799	0.85223	0.29130	0.34803	0.40602	0.44197
128	0.39953	0.54516	0.73256	0.87037	0.29895	0.35883	0.41852	0.45364
256	**0.41524**	0.55911	0.74485	0.87822	**0.31114**	**0.37025**	**0.42938**	**0.46345**
512	0.41203	**0.56201**	**0.75167**	**0.88407**	0.30575	0.36747	0.42916	0.46301
1024	0.40614	0.55696	0.74862	0.88324	0.30172	0.36304	0.42446	0.45969

TABLE 6. The different embedding dimensions in terms of Recall@N and NDCG@N on Movielens dataset.

Embedding size	Recall@3	Recall@5	Recall@10	Recall@20	NDCG@3	NDCG@5	NDCG@10	NDCG@20
8	0.21046	0.27203	0.36023	0.45055	0.16167	0.18695	0.21492	0.23820
16	0.26760	0.33705	0.43312	0.52716	0.20484	0.23269	0.26473	0.28842
32	0.29689	0.37774	0.48010	0.58723	0.22902	0.26107	0.29428	0.32046
64	0.32475	0.41109	0.52123	0.62358	0.24628	0.28243	0.31821	0.34404
128	0.33404	0.42507	0.54414	0.64835	0.25402	0.29049	0.33048	0.35613
256	**0.34457**	**0.44135**	**0.56201**	**0.66658**	**0.26010**	**0.30005**	**0.33879**	**0.36554**
512	0.33519	0.43613	0.56104	0.66357	0.25416	0.29424	0.33476	0.36104
1024	0.33289	0.43215	0.55662	0.66030	0.25382	0.29440	0.33442	0.36085

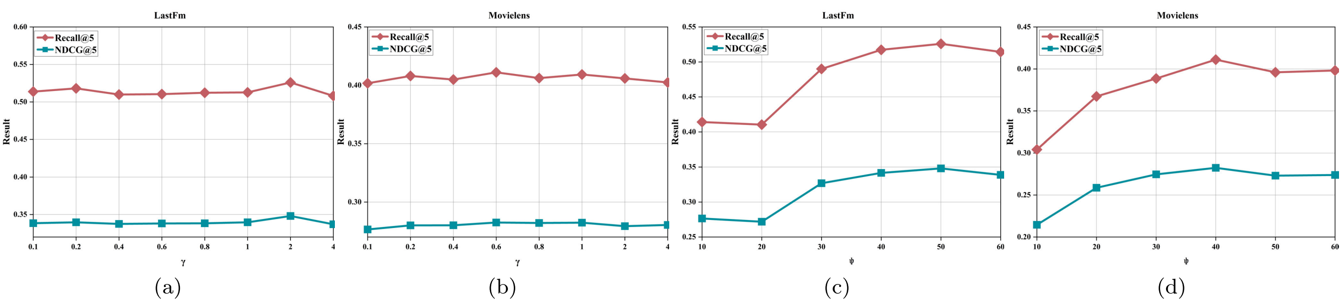


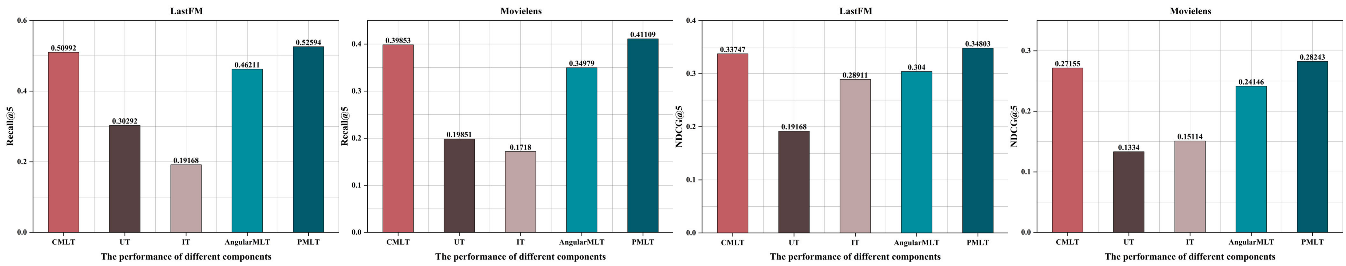
FIGURE 3. The performance of PMLT with respect to different values of γ and φ on two datasets.

G. RECOMMENDATION PERFORMANCE WITH DIFFERENT FIXED MARGINS AND DIFFERENT ITERATIONS

1) RECOMMENDATION PERFORMANCE WITH DIFFERENT FIXED MARGINS

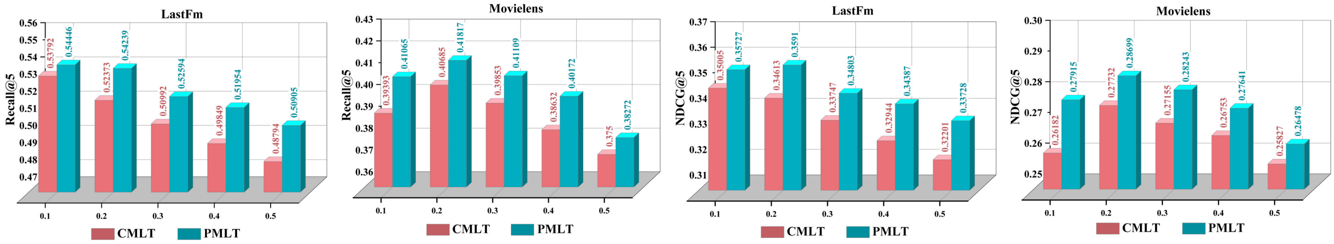
To explore the effect of fixed margins on the recommended performance, our experiments on real datasets quantitatively demonstrate the recommendation results on LastFm and Movielens for different fixed margins.

As shown in the Figure 5, we investigate the performance of different values of the fixed margin m in CMLT and PMLT on Movielens and LastFm under the previous 1000 iterations, where the value of m increases by the step of 0.1. It can be seen that as the value of m changes, the value of CMLT at Recall@5 and NDCG@5 reaches the optimal value near $m = 0.2$, and then drops significantly after it is greater than 0.2. This indicates that if the fixed margin m is too small or too large, it will be difficult to distinguish the distance between user-tag and item-tag, so that the tags close to the user and item are unlikely to be included in the Top-N tag recommendation list, which has an adverse impact on the accuracy of recommendation. Then PMLT makes the user-tag distance and item-tag distance with high similarity closer by applying the constraint of Angle φ to contrast with CMLT in this way. Comparing PMLT and CMLT under different values of m, the accuracy of PMLT is increased by 4.2%, 2.7%, 3.1%, 3.9%, 2% and 6.6%, 3.4%, 4%, 3.3%, 2.5% on the Movielens dataset accordingly for Recall@5 and NDCG@5. The corresponding improvements in Recall@5 and NDCG@5 on



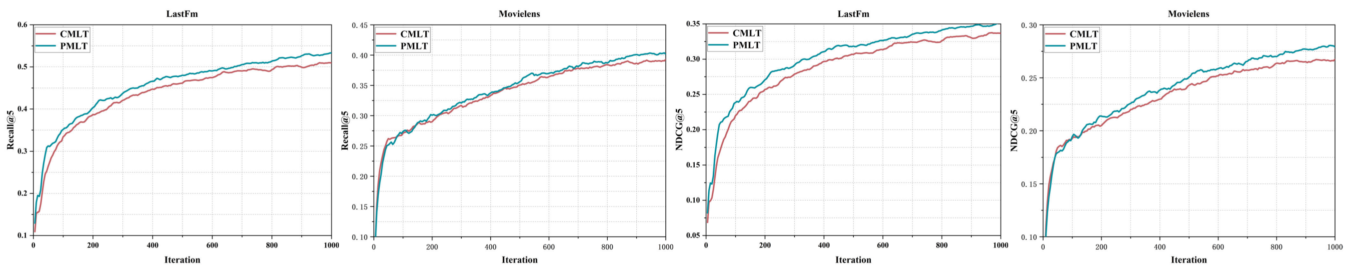
(a) (b) (c) (d)

FIGURE 4. The different components performance of PMLT for Recall@5 and NDCG@5 on two datasets.



(a) (b) (c) (d)

FIGURE 5. The impacts of the m in terms of Recall@5 and NDCG@5 on the two datasets.



(a) (b) (c) (d)

FIGURE 6. The iterative performance of PMLT and CMLT for Recall@5 and NDCG@5 on two datasets.

the LastFm dataset are 1.2%, 3.5%, 3.1%, 4.2%, 4.3% and 2%, 3.7%, 3.1%, 4.3%, 4.7%. The results indicate that the proposed PMLT method improves the performance of recommendation by dynamically adjusting the distance of different entity pairs to a certain extent through the angular margin approach, which can effectively distinguish the irrelevant tags of users and items, compared with the CMLT using only the metric learning method.

2) RECOMMENDATION PERFORMANCE WITH DIFFERENT ITERATIONS

To further illustrate the effectiveness of the PMLT method, we investigated the comparative performance of CMLT and PMLT in 1000 iterations under optimal parameter settings.

As shown in Figure 6, according to the recommendation performance on LastFm and Movielens datasets, it can be found that with the increase of iterations, the recommendation performance of PMLT in different iterations is much higher than that of CMLT in terms of Recall@5 and NDCG@5. This indicates that on the basis of CMLT, by imposing user-tag and item-tag angular margin constraints, it has a significant positive effect on adjusting user-tag and item-tag distances. PMLT can determine more stable prediction results, and the method can enhance the computational advantage of metric learning methods, which has significant performance improvement in tag recommendation.

V. CONCLUSION

Tensor factorization based approaches have received wide attention in tag recommendation, and these methods predict scores by the inner product among the user × item × tag triplet. And the main problem of inner product is that it does not satisfy the triangle inequality and cannot capture the fine-grained preference information, which reduces the recommendation performance. In addition, metric learning methods tend to distinguish positive and negative samples with a fixed margin. However, different entity pairs have different intra-class variations and the fixed margin does not change with the entities, which limits the expressiveness of the model. In this work, a pairwise metric learning method with angular margin is proposed for tag recommendation. Firstly, the distance relationship among user $\times$ item $\times$ tag triplets is modeled using metric learning method. In the proposed method, Euclidean distances regarding user-tag and item-tag can be considered at the same time, which can reduce the distance between similar data of user-tag and item-tag through pairwise loss, and expand the distance between different data. All the tags can be included in the Top-N tag recommendation list. Secondly, aiming at the problem that the fixed margin is difficult to distinguish between positive and negative samples, an entity pair relationship based on the angular margin constraint is proposed to dynamically adjust the user-tag and item-tag distance changes. In addition, based on the original pairwise loss, a new objective function combining angular margin is established, and an angular margin regularization is added, which can control the angular margin of user-tag and item-tag respectively, and improve the stability of the model by adjusting the angular margin. In the end, extensive experiments are conducted on real datasets, LastFm and Movielens. The results show that the proposed PMLT is better than other baselines in terms of Recall@N and NDCG@N. And the influence of different parameters and components on tag recommendation performance is also analyzed. It can be found that the proposed pairwise metric learning method with angular margin has competitive performance.

REFERENCES

[1] F. M. Belém, J. M. Almeida, and M. A. Gonçalves, "A survey on tag recommendation methods," *J. Assoc. Inf. Sci. Technol.*, vol. 68, no. 4, pp. 830–844, Apr. 2017.

[2] Z. Guan, J. Bu, Q. Mei, C. Chen, and C. Wang, "Personalized tag recommendation using graph-based ranking on multi-type interrelated objects," in *Proc. 32nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2009, pp. 540–547.

[3] X. Liu, X. Qian, D. Lu, X. Hou, and L. Wang, "Personalized tag recommendation for Flickr users," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2014, pp. 1–6.

[4] R. Jäschke, L. Marinho, A. Hotho, L. Schmidt-Thieme, and G. Stumme, "Tag recommendations in folksonomies," in *Knowledge Discovery in Databases: PKDD 2007*. Warsaw, Poland: Springer, Sep. 2007, pp. 506–514.

[5] E. Quintanilla, Y. Rawat, A. Sakryukin, M. Shah, and M. Kankanhalli, "Adversarial learning for personalized tag recommendation," *IEEE Trans. Multimedia*, vol. 23, pp. 1083–1094, 2021.

[6] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Adv. Artif. Intell.*, vol. 2009, no. 4, Jan. 2009, Art. no. 421425.

[7] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *IEEE Comput.*, vol. 42, no. 8, pp. 30–37, Aug. 2009.

[8] X. Luo, M. Zhou, Y. Xia, and Q. Zhu, "An efficient non-negative matrix-factorization-based approach to collaborative filtering for recommender systems," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1273–1284, May 2014.

[9] A. Karatzoglou, X. Amatriain, L. Baltrunas, and N. Oliver, "Multiverse recommendation: N-dimensional tensor factorization for context-aware collaborative filtering," in *Proc. 4th ACM Conf. Recommender Syst.*, Sep. 2010, pp. 79–86.

[10] S. Rendle and L. Schmidt-Thieme, "Pairwise interaction tensor factorization for personalized tag recommendation," in *Proc. 3rd ACM Int. Conf. Web Search Data Mining*, Feb. 2010, pp. 81–90.

[11] L. Yu, J. Huang, G. Zhou, C. Liu, and Z.-K. Zhang, "TIIREC: A tensor approach for tag-driven item recommendation with sparse user generated content," *Inf. Sci.*, vol. 411, pp. 122–135, Oct. 2017.

[12] P. Symeonidis, A. Nanopoulos, and Y. Manolopoulos, "Tag recommendations based on tensor dimensionality reduction," in *Proc. ACM Conf. Recommender Syst.*, Oct. 2008, pp. 43–50.

[13] S. Rendle, L. B. Marinho, A. Nanopoulos, and L. Schmidt-Thieme, "Learning optimal ranking with tensor factorization for tag recommendation," in *Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jun. 2009, pp. 727–736.

[14] J. D. Carroll and J.-J. Chang, "Analysis of individual differences in multidimensional scaling via an n-way generalization of 'Eckart–Young' decomposition," *Psychometrika*, vol. 35, no. 3, pp. 283–319, Sep. 1970.

[15] X. Fang, R. Pan, G. Cao, X. He, and W. Dai, "Personalized tag recommendation through nonlinear tensor factorization using Gaussian kernel," in *Proc. AAAI Conf. Artif. Intell.*, 2015, vol. 29, no. 1, pp. 1–7.

[16] J. Yuan, Y. Jin, W. Liu, and X. Wang, "Attention-based neural tag recommendation," in *Proc. Int. Conf. Database Syst. Adv. Appl.* Chiang Mai, Thailand: Springer, 2019, pp. 350–365.

[17] H. Chen and J. Li, "Adversarial tensor factorization for context-aware recommendation," in *Proc. 13th ACM Conf. Recommender Syst.*, Sep. 2019, pp. 363–367.

[18] X. He, Z. He, X. Du, and T.-S. Chua, "Adversarial personalized ranking for recommendation," in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jun. 2018, pp. 355–364.

[19] R. Krestel, P. Fankhauser, and W. Nejdl, "Latent Dirichlet allocation for tag recommendation," in *Proc. 3rd ACM Conf. Recommender Syst.*, Oct. 2009, pp. 61–68.

[20] J. Yao, Y. Wang, Y. Zhang, J. Sun, and J. Zhou, "Joint latent Dirichlet allocation for social tags," *IEEE Trans. Multimedia*, vol. 20, no. 1, pp. 224–237, Jan. 2018.

[21] Y. Zhang, N. Zhang, and J. Tang, "A collaborative filtering tag recommendation system based on graph," in *Proc. ECML PKDD Discovery Challenge*, 2009, pp. 297–306.

[22] M. Ramezani, "Improving graph-based approaches for personalized tag recommendation," *J. Emerg. Technol. Web Intell.*, vol. 3, no. 2, pp. 168–176, May 2011.

[23] R. Huang, C. Han, and L. Cui, "Tag-aware attentional graph neural networks for personalized tag recommendation," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2021, pp. 1–8.

[24] C.-K. Hsieh, L. Yang, Y. Cui, T.-Y. Lin, S. Belongie, and D. Estrin, "Collaborative metric learning," in *Proc. 26th Int. Conf. World Wide Web*, Apr. 2017, pp. 193–201.

[25] Y. Tay, L. Anh Tuan, and S. C. Hui, "Latent relational metric learning via memory-based attention for collaborative ranking," in *Proc. World Wide Web Conf. World Wide Web*, 2018, pp. 729–739.

[26] C. Park, D. Kim, X. Xie, and H. Yu, "Collaborative translational metric learning," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2018, pp. 367–376.

[27] M. Li, S. Zhang, F. Zhu, W. Qian, L. Zang, J. Han, and S. Hu, "Symmetric metric learning with adaptive margin for recommendation," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 4, pp. 4634–4641.

[28] C. Ma, L. Ma, Y. Zhang, R. Tang, X. Liu, and M. Coates, "Probabilistic metric learning with adaptive margin for top-K recommendation," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 1036–1044.

[29] J. Park, Y.-C. Lee, and S.-W. Kim, "Effective and efficient negative sampling in metric learning based recommendation," *Inf. Sci.*, vol. 605, pp. 351–365, Aug. 2022.

[30] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," 2012, *arXiv:1205.2618*.

[31] Y. Cai, M. Zhang, D. Luo, C. Ding, and S. Chakravarthy, "Low-order tensor decompositions for social tagging recommendation," in *Proc. 4th ACM Int. Conf. Web Search Data Mining*, Feb. 2011, pp. 695–704.

[32] C. Lu, X. Hu, J.-R. Park, and J. Huang, "Post-based collaborative filtering for personalized tag recommendation," in *Proc. iConference*, Feb. 2011, pp. 561–568.

[33] Y. Yu, X. Chen, L. Zhang, R. Gao, and H. Gao, "Neural graph for personalized tag recommendation," *IEEE Intell. Syst.*, vol. 37, no. 1, pp. 51–59, Jan. 2022.

[34] K. Wang, Y. Jin, H. Wang, H. Peng, and X. Wang, "Personalized time-aware tag recommendation," in *Proc. AAAI Conf. Artif. Intell.*, 2018, vol. 32, no. 1, pp. 1–8.

[35] H. T. Nguyen, M. Wistuba, J. Grabocka, L. R. Drumond, and L. Schmidt-Thieme, "Personalized deep learning for tag recommendation," in *Proc. Pacific–Asia Conf. Knowl. Discovery Data Mining*. Springer, 2017, pp. 186–197.

[36] P. Zhou, J. Liu, X. Liu, Z. Yang, and J. Grundy, "Is deep learning better than traditional approaches in tag recommendation for software information sites?" *Inf. Softw. Technol.*, vol. 109, pp. 1–13, May 2019.

[37] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 26, 2013, pp. 1–9.

[38] C. R. Givens and R. M. Shortt, "A class of Wasserstein metrics for probability distributions," *Michigan Math. J.*, vol. 31, no. 2, pp. 231–240, 1984.

[39] J. Wang, F. Zhou, S. Wen, X. Liu, and Y. Lin, "Deep metric learning with angular loss," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2593–2601.

[40] B. Yu and D. Tao, "Deep metric learning with tuplet margin loss," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6490–6499.

[41] D. Zhang, Y. Li, and Z. Zhang, "Deep metric learning with spherical embedding," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 18772–18783.

[42] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

ZHENGSHUN FEI (Member, IEEE) received the degrees from the Department of Control Science and Engineering, Zhejiang University, Hangzhou, China, in 2007 and 2012, respectively.

From September 2018 to September 2019, he was a Visiting Scholar with the University of Notre Dame, Notre Dame, IN, USA. He is currently an Associate Professor with the School of Automation and Electrical Engineering, Zhejiang University of Science and Technology. His research interests include system modeling, machine learning, optimal control, and their applications.

JINGLONG WANG is currently pursuing the master's degree with the School of Automation and Electrical Engineering, Zhejiang University of Science and Technology, Hangzhou, China. His research interests include recommendation systems, metric learning, and hyperbolic spaces.

BINGQIANG HUANG received the Ph.D. degree in control science and engineering from Shanghai Jiao Tong University, China, in 2007. From 2008 to 2011, he was a Researcher with the PSE Group, Sumitomo Chemical, Japan. From 2011 to 2014, he was a Postdoctoral Researcher with the Department of Control Science and Engineering, Zhejiang University, China. He is currently a Lecturer with the School of Automation and Electrical Engineering, Zhejiang University of Science and Technology. His research interests include computer vision, image recognition, and intelligent control.

XINJIAN XIANG received the M.S. degree in automation from Donghua University, in 1988.

He is currently a Professor with the School of Automation and Electrical Engineering, Zhejiang University of Science and Technology. His research interests include intelligence manufacturing, robotics engineering, the Internet of Things, and artificial intelligence.