The Minority Matters: A Diversity-Promoting Collaborative Metric Learning Algorithm

Shilong Bao[1,2] **Qianqian Xu**[3*] **Zhiyong Yang**[4] **Yuan He**[5]
Xiaochun Cao[6] **Qingming Huang**[3,4,7,8*]
[1] State Key Laboratory of Information Security, Institute of Information Engineering, CAS
[2] School of Cyber Security, University of Chinese Academy of Sciences
[3] Key Lab. of Intelligent Information Processing, Institute of Computing Technology, CAS
[4] School of Computer Science and Tech., University of Chinese Academy of Sciences
[5] Alibaba Group
[6] School of Cyber Science and Technology, Shenzhen Campus, Sun Yat-sen University
[7] Key Laboratory of Big Data Mining and Knowledge Management, CAS
[8] Peng Cheng Laboratory
baoshilong@iie.ac.cn, xuqianqian@ict.ac.cn, heyuan.hy@alibaba-inc.com,
caoxiaochun@mail.sysu.edu.cn, {yangzhiyong21,qmhuang}@ucas.ac.cn

Abstract

Collaborative Metric Learning (CML) has recently emerged as a popular method in recommendation systems (RS), closing the gap between metric learning and Collaborative Filtering. Following the convention of RS, existing methods exploit unique user representation in their model design. This paper focuses on a challenging scenario where a user has multiple categories of interests. Under this setting, we argue that the unique user representation might induce preference bias, especially when the item category distribution is imbalanced. To address this issue, we propose a novel method called *Diversity-Promoting Collaborative Metric Learning* (DPCML), with the hope of considering the commonly ignored minority interest of the user. The key idea behind DPCML is to include a multiple set of representations for each user in the system. Based on this embedding paradigm, user preference toward an item is aggregated from different embeddings by taking the minimum item-user distance among the user embedding set. Furthermore, we observe that the diversity of the embeddings for the same user also plays an essential role in the model. To this end, we propose a *Diversity Control Regularization Scheme* (DCRS) to accommodate the multi-vector representation strategy better. Theoretically, we show that DPCML could generalize well to unseen test data by tackling the challenge of the annoying operation that comes from the minimum value. Experiments over a range of benchmark datasets speak to the efficacy of DPCML.

1 Introduction

Recommender system (RS) is a well-known building block in eCommerce, which can assist buyers to find products they wish to purchase by giving them the relevant recommendations. The key recipe behind RS is to learn from user-item interaction records [45, 30, 31, 29, 19]. In practice, since user preferences are hard to collect, such records often exist as implicit feedback [48, 1, 42] where only indirect actions are provided (say clicks, collections, reposts, and etc.). Such a property of implicit

*Corresponding authors.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

https://doi.org/10.52202/068431-0178

feedback raises a great challenge to RS-targeted machine learning methods and thus stimulates a wave of relevant studies along this course [56, 63, 51].

Over the past two decades, most literature follows a typical paradigm known as the One-Class Collaborative Filtering (OCCF) [35], where the items not being observed are usually assumed to be of less interest for the user and labeled as negative instances. In the early days, the vast majority of studies in the OCCF community focus on Matrix Factorization (MF) based algorithms, where the preference of a specific user to an item is conveyed by the inner product between their embeddings [60, 6]. Recently, a milestone study known as *Collaborative Metric Learning* (CML) [18] pointed out that the inner product involved in MF violates the triangle inequality, resulting in a sub-optimal topological embedding space. To fix this, CML proposes a novel framework to overcome such a problem by borrowing the strength from metric learning. Practically, CML has achieved promising performance over a series of RS benchmark datasets. Hereafter, many efforts have been made along the research direction to improve CML [44, 41, 36, 2, 54, 47, 65, 62, 43]. More discussions of the related work are presented in Appendix.A.

However, through the lens of a critical example in the practical scenarios (shown in Sec.2.2), we notice that users usually have multiple categories of preferences in real-world RS. Moreover, such interest groups are often not equally distributed, where the amount of some groups dominates the others. Unfortunately, as shown in Fig.3, in this case, the existing studies might induce preference bias since they tend to meet the majority interest while missing the other potential preference. Therefore, in this paper, we ask:

How to develop an effective CML-based algorithm to accommodate the diversity of user preferences?

Contributions. In search of an answer, we propose a novel algorithm called *Diversity-Promoting Collaborative Metric Learning (DPCML)*. The key idea is to explore the diversity of user interest which spans multiple groups of items. To this end, we propose a multi-vector user representation strategy, where each user has a set of C embeddings. To find out the score of a given item embedding $\boldsymbol{g}_v$, we aggregate the results from the user embeddings $\boldsymbol{g}_u^1, \boldsymbol{g}_u^2, \cdots, \boldsymbol{g}_u^C$ by taking the minimum distance $s(u, v) = \min_c \|\boldsymbol{g}_u^c - \boldsymbol{g}_v\|^2$. Then we will recommend the item with the smallest s value. In this way, we can focus on all potential items that fit one of the users' interests well, both for the majority and the minority interests. Meanwhile, we observe that the diversity of the embeddings among the same user representation set also plays an important role in better achieving our goal. Therefore, we further present a novel diversity control regularization scheme.

Taking a step further, we continue to ask the following question:

Could CML generalize well under the multi-vector representation strategy?

To the best of our knowledge, such a problem remains barely explored in the existing literature. To solve the problem, we then proceed to explore the generalization bound for DPCML algorithm. Here the major challenges fall into two aspects: 1) The risk of DPCML could not be expressed as a sum of independently identically distributed (i.i.d.) loss terms, making the standard Rademacher Complexity-based [3, 33] theoretical arguments unavailable; 2) The annoying minimum operation are not continuous, which cannot be analyzed easily in the Rademacher complexity framework. Facing these challenges, we employ the covering number and ϵ-net arguments to derive the generalization bound. On top of this, we show that DPCML could induce a small generalization error with high probability. This supports the effectiveness of DPCML from a theoretical perspective.

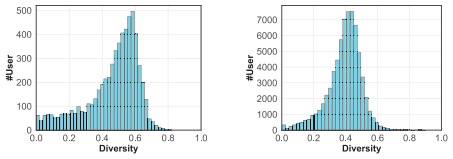
Finally, we conduct empirical studies over a range of RS benchmark datasets that demonstrate the superiority of DPCML.

2 Methodology

2.1 Preliminary

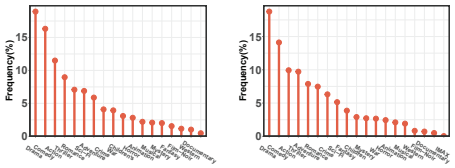
In this paper, we focus on how to develop an effective CML-based recommendation system on top of the implicit feedback signals (say clicks, browses, and bookmarks). Assume there are a pool of users and items in the system, denoted by $\mathcal{U} = \{u_1, u_2, \ldots, u_{|\mathcal{U}|}\}$ and $\mathcal{I} = \{v_1, v_2 \ldots, v_{|\mathcal{I}|}\}$,

https://doi.org/10.52202/068431-0178



(a) MovieLens-1M (b) MovieLens-10M

Figure 1: Statistics of preference diversity.



(a) MovieLens-1M (b) MovieLens-10M

Figure 2: The item category distribution.

respectively. For each user $u_i \in \mathcal{U}, i = 1, 2, \ldots, |\mathcal{U}|$, let $\mathcal{D}^+_{u_i} = \{v^+_1, v^+_2, \ldots, v^+_{n^+_i}\}$ denote the set of items that user u_i has interacted with (i.e., observed user-item interactions) and the rest of the items (i.e., unobserved interactions) are denoted by $\mathcal{D}^-_{u_i} = \{v^-_1, v^-_2, \ldots, v^-_{n^-_i}\}$, where n^+_i, n^-_i are the number of observed/unobserved interactions of user u_i. We have $\mathcal{I} = \mathcal{D}_{u_i} = \mathcal{D}^+_{u_i} \cup \mathcal{D}^-_{u_i}$ and $|\mathcal{I}| = n^+_i + n^-_i$. In the standard settings of OCCF, one usually assumes that users tend to have a higher preference for the items contained in $\mathcal{D}^+_{u_i}$ than the items in $\mathcal{D}^-_{u_i}$. Therefore, given a target user $u_i \in \mathcal{U}$ and his/her historical interaction records, the goal of RS is to discover the most interested N items by recommending the items with the top-N (bottom-N) score. The top-N item list is denoted as $\mathcal{I}^{u_i}_N$.

2.2 Motivating Example

We start by a definition of the preference diversity of users.

Definition 1 (Preference Diversity)**.** Assume that there exists an attribute set $\mathcal{T} = \{\mathcal{T}(v_1), \mathcal{T}(v_2), \ldots, \mathcal{T}(v_{|\mathcal{I}|})\}$ in a typical RS, where $\mathcal{T}(v_j) = \{t_1, t_2, \ldots, t_{T_j}\}$ contains the attribute information of item v_j (e.g., the genres of a movie) and T_j is the number of attributes. Given a user u_i and interaction records $\mathcal{D}^+_{u_i}$, the preference diversity is defined as follows:

$$\mathsf{Div}(u_i) = \frac{\sum_{v_j, v_k \in \mathcal{D}^+_{u_i}, v_j \neq v_k} \mathbb{I}\left[\mathcal{T}(v_j) \cap \mathcal{T}(v_k) = \varnothing\right]}{|\mathcal{D}^+_{u_i}|(|\mathcal{D}^+_{u_i}| - 1)},$$

where $\mathbb{I}(x)$ is an indicator function, i.e., returns 1 if the condition x holds, otherwise 0 is returned.

Remark 1. Intuitively, the range of $\mathsf{Div}(u_i)$ is among $[0, 1]$, and its value measures the diversity of u_i's preference to a certain extent. That is to say, if items among the historical interaction records of users are irrelevant, there should induce a large value (e.g., $\mathsf{Div}(u_i) = 1$), implying the diversity of their preferences. If the opposite is the case, the value is small. This means users may have narrow interests where only some unique attributes appeal to them.

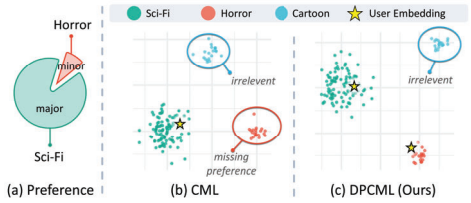


Figure 3: An illustration shows the benefit of our proposed algorithm when a user has multiple categories of preferences. Taking movies as an example, we assume that Sci-Fi/Horror is the majority/minority interest of the user while Cartoon is an irrelevant movie type. It is easy to see that if the item embeddings are distributed as shown in the figure, we can hardly find a single user embedding that simultaneously captures both interests.

Based on Def.1, we visualize the user preferences on two real-world benchmark datasets, including **MovieLens-1m** and **MovieLens-10m**. The detailed information of datasets is listed in Tab.1 in Appendix.C. Here we adopt the movie genres as the attribute set $\mathcal{T}$ because such information is easy to obtain. The results are shown in Fig.1. From the results, we can make the following observations. First, only a few users have limited interest. Moreover, most of the users have a diversity value spaning $(0, 0.8]$, suggesting that they have multiple categories of interests. Finally, one can notice that there are very few users with high preference diversity (at the lower-right corner) in both figures. This is a convincing case in the real-world recommendation since most users usually have interests in a couple of movie genres but not all.

https://doi.org/10.52202/068431-0178

Motivation and Discussion. Through the above example, the key information is that users usually have multiple categories of preference in real-world recommendations. This poses a critical challenge to the current CML framework. Specifically, following the convention of RS, the existing CML-based methods leverage unique representations of users to model their preferences. Facing the multiplicity of user intentions, such a paradigm may induce preference bias due to the limited expressiveness, especially when the item category distribution is imbalanced. Fig.2 visualizes the item distribution on MovieLens-1m and MovieLens-10m datasets. We see that both of them are imbalanced. In this case, as shown in Fig.3-(b), CML would pay more attention to the **majority** interest of users making the unique user embedding close to the items with the science fiction (Sci-Fi) category. In this way, the **minority** interest of the user (i.e., Horror movies) would be ignored by the method, inducing performance degradation. This motivates us to explore diversity-promoting strategies on top of CML.

2.3 Diversity-Promoting Collaborative Metric Learning

Recall that the critical recipe behind CML-based algorithms is to seek a metric space (usually adopting the Euclidean space) such that user preferences could be naturally specified by their distance toward different items. To do this, the traditional CML-based methods usually represent each user and each item as a vector, respectively. Different from them, taking the preference diversity of users into account, we propose to adopt C ($C > 1$) different embeddings for each user and represent each item as one single vector in a joint Euclidean space.

Concretely, each user u_i is projected into the metric space via the following lookup transformations [49, 50, 55]:

$$\boldsymbol{g}_{u_i}^c = \boldsymbol{P}_c^\top \boldsymbol{u}_i, \ \ \forall c, u_i, \ c \in [C], \ \ u_i \in \mathcal{U}, \tag{1}$$

where $\boldsymbol{g}_{u_i}^c \in \mathbb{R}^d$ is a representation vector of user u_i; $[C]$ is the set $\{1, 2, \ldots, C\}$; $\boldsymbol{P}_c \in \mathbb{R}^{|\mathcal{U}| \times d}$ is a learned transformation weight; d is the dimension of space and $\boldsymbol{u}_i \in \mathbb{R}^{|\mathcal{U}|}$ is a one-hot encoding that the nonzero elements correspond to its index of a particular user u_i.

Similarly, we apply the following transformation to each item v_j:

$$\boldsymbol{g}_{v_j} = \boldsymbol{Q}^\top \boldsymbol{v}_j, \ \ \forall v_j \in \mathcal{I}, \tag{2}$$

where $\boldsymbol{g}_{v_j} \in \mathbb{R}^d$ is the embedding of item v_j; $\boldsymbol{Q} \in \mathbb{R}^{|\mathcal{I}| \times d}$ is the learned transformation weight and $\boldsymbol{v}_j \in \mathbb{R}^{|\mathcal{I}|}$ is a one-hot embedding of item v_j.

In what follows, given a target user u_i, we need to find out a score function to express the user preference toward an item in the context of multiple representations of users. Here we define the score function by taking the minimum item-user Euclidean distance among the user embedding set:

$$s(u_i, v_j) = \min_{c \in [C]} \|\boldsymbol{g}_{u_i}^c - \boldsymbol{g}_{v_j}\|^2, \forall v_j \in \mathcal{I}\,. \tag{3}$$

Equipped with this formulation, we focus on the potential items that fit one of the user preferences. If user u_i has interacted with item v_j, there should be a small value with respect to $s(u_i, v_j)$. If the opposite is the case, we then expect to see a large $s(u_i, v_j)$. Mathematically, the following inequality should be satisfied to reflect the relative preference of u_i in the learned Euclidean space:

$$s(u_i, v_j^+) < s(u_i, v_k^-), \forall v_j^+ \in \mathcal{D}_{u_i}^+, \ \forall v_k^- \in \mathcal{D}_{u_i}^-. \tag{4}$$

Therefore, given the whole sample set $\mathcal{D} = \bigcup_{u_i \in \mathcal{U}} \mathcal{D}_{u_i}$, we adopt the following pairwise learning problems [18, 44, 24, 57] to achieve such goal:

$$\min_{\boldsymbol{g}} \ \ \hat{\mathcal{R}}_{\mathcal{D},\boldsymbol{g}}, \tag{5}$$

where, $\forall v_j^+ \in \mathcal{D}_{u_i}^+, \ \forall v_k^- \in \mathcal{D}_{u_i}^-$, we have

$$\hat{\mathcal{R}}_{\mathcal{D},\boldsymbol{g}} = \frac{1}{|\mathcal{U}|} \sum_{u_i \in \mathcal{U}} \frac{1}{n_i^+ n_i^-} \sum_{j=1}^{n_i^+} \sum_{k=1}^{n_i^-} \ell_g^{(i)}(v_j^+, v_k^-),$$

$$\ell_g^{(i)}(v_j^+, v_k^-) = \max(0, \lambda + s(u_i, v_j^+) - s(u_i, v_k^-)). \tag{6}$$

and $\lambda > 0$ is a safe margin.

According to (5), we have the following explanations. At first, optimizing the above problem could pull the observed items close to the users and push the unobserved items apart from the observed items. This achieves our goal of preserving user preferences in the Euclidean space. Then, as shown in Fig.3-(c), equipped with a multiple set of representations for each user, DPCML would exploit different user vectors to focus on different interest groups. In this sense, the minority interest groups can also be modeled well. Last but not least, one appealing property is that, DPCML also preserves the triangle inequality for the items falling into the same interest group.

2.4 Diversity Control Regularization Scheme

In practice, we note that a proper regularization scheme is crucial to accommodate the multi-vector representation strategy. Here we focus on the diversity within the embedding sets of a given user. Such diversity is defined as the average pairwise distance among the C user embeddings for user u_i, i.e.,

$$\delta_{\boldsymbol{g},u_i} = \frac{1}{2C(C-1)} \sum_{c_1,c_2 \in [C]} \|\boldsymbol{g}_{u_i}^{c_1} - \boldsymbol{g}_{u_i}^{c_2}\|^2.$$

Based on the definition, we argue that one should attain a proper $\delta_{\boldsymbol{g},u_i}$ to get a good performance since extremely large/small values of $\delta_{\boldsymbol{g},u_i}$ might be harmful for the generalization error. It is easy to see that if $\delta_{\boldsymbol{g},u_i}$ is extremely small, the embeddings for a given user are very close to each other such that the multi-vector representation strategy degenerates to the original single-vector representation. This increases the model complexity with few performance gains and obviously will induce overfitting. On the other hand, a too large diversity might also induce overfitting. It might be a bit confusing at first glance. But, imagine that when some noise observations or extremely rare interests far away from the normal patterns exist in the data, having a large diversity will make it easier to overfit such data. Moreover, it is also a natural assumption that a user's interests should not be too different, as validated in Fig.1. In this sense, the distance across different user embeddings should remain at a moderate magnitude.

Therefore, controlling a proper diversity is essential for the multi-vector representation. To do this, we put forward the following diversity control regularization scheme (DCRS):

$$\hat{\Omega}_{\mathcal{D},\boldsymbol{g}} = \frac{1}{|\mathcal{U}|} \sum_{u_i \in \mathcal{U}} \psi_{\boldsymbol{g}}(u_i), \tag{7}$$

where, we have

$$\psi_{\boldsymbol{g}}(u_i) = \max\left(0, \delta_1 - \delta_{\boldsymbol{g},u_i}\right) + \max\left(0, \delta_{\boldsymbol{g},u_i} - \delta_2\right),$$

and δ_1, δ_2 are two threshold parameters with $\delta_1 \leq \delta_2$. Intuitively, optimizing (7) ensures that the diversity of user's vectors lies between δ_1 and δ_2.

2.5 Optimization

Finally, we arrive at the following optimization problem for our proposed DPCML:

$$\min_{\boldsymbol{g}} \hat{\mathcal{L}}_{\mathcal{D}}(\boldsymbol{g}) := \hat{\mathcal{R}}_{\mathcal{D},\boldsymbol{g}} + \eta \cdot \hat{\Omega}_{\mathcal{D},\boldsymbol{g}}, \tag{8}$$

where η is a trade-off hyper-parameter.

When the training is completed, one can easily carry out recommendations by choosing the items with the smallest $s(u_i, v_j), \forall v_j, v_j \in \mathcal{I}$.

2.6 General Framework of Joint Accessibility

Now, we expect to provide another intriguing perspective of our proposed method. As we discussed in Sec.A.2, equipped with a multiple set of representations for each user, our proposed algorithm could be treated as a generalized framework against the joint accessibility issue. To see this, if we

https://doi.org/10.52202/068431-0178

restrict the user and item embeddings within a unit sphere, then the score function (3) degenerates to :

$$\begin{aligned} s(u_i, v_j) &= \min_{c\in[C]} \left(1 - \hat{\boldsymbol{g}}_{u_i}^c \boldsymbol{g}_{v_j}\right), \\ s.t.\ \ \|\boldsymbol{g}_{u_i}^c\| &= 1, \forall u_i \in \mathcal{U}, \\ \|\boldsymbol{g}_{v_j}\| &= 1, \forall v_j \in \mathcal{I}, \end{aligned} \tag{9}$$

where $\hat{\boldsymbol{g}}_{u_i}^c \in \mathbb{R}^{1\times d}$ represents the transpose vector of $\boldsymbol{g}_{u_i}^c \in \mathbb{R}^d$. Therefore, to minimize (9), one only needs to maximize the following equivalent problem:

$$\begin{aligned} \hat{s}(u_i, v_j) &= \max_{c\in[C]} \hat{\boldsymbol{g}}_{u_i}^c \boldsymbol{g}_{v_j}, \\ s.t.\ \ \|\hat{\boldsymbol{g}}_{u_i}^c\| &= 1, \forall u_i \in \mathcal{U}, \\ \|\boldsymbol{g}_{v_j}\| &= 1, \forall v_j \in \mathcal{I}, \end{aligned} \tag{10}$$

which is exactly the original form of the joint accessibility model.

3 Generalization Analysis

In this section, we present a systematic theoretical analysis of the generalization ability of our proposed algorithm. Following the standard learning theory, deriving a uniform upper bound of the generalization error relies on the proper measure of its complexity over the given hypothesis space $\mathcal{H}$. The most common complexity to achieve this is the Rademacher complexity [3, 33, 23], which is derived from the symmetrization technique as an upper bound for the largest deviation over a given hypothesis space $\mathcal{H}$:

$$\mathbb{E}_{\mathcal{D}}\left[\sup_{f\in\mathcal{H}} \mathbb{E}_{\mathcal{D}}(\hat{\mathcal{R}}_{\mathcal{D}}) - \hat{\mathcal{R}}_{\mathcal{D}}\right].$$

However, the standard symmetrization technique requires the empirical risk $\hat{\mathcal{R}}_{\mathcal{D}}$ to be a sum of independent terms, which is not applicable for the CML-based methods since they usually involve a sum of pairwise terms in (5). For instance, with respect to (5), the terms $\ell_g^{(i)}(v_j^+, v_k^-)$ and $\ell_g^{(i)}(\tilde{v}_j^+, \tilde{v}_k^-)$ are interdependent as long as one of them is the same (i.e., $v_j^+ = \tilde{v}_j^+$ or $v_k^- = \tilde{v}_k^-$).

Therefore, we turn to leverage another complexity measure, i.e., covering number, to overcome this difficulty. The necessary notations are summarized as follows.

Definition 2 (ϵ-Covering)**.** [21] Let $(\mathcal{F}, \rho)$ be a (pseudo) metric space, and $\mathcal{G} \subseteq \mathcal{F}$. $\{f_1, \ldots, f_K\}$ is said to be an ϵ-covering of $\mathcal{G}$ if $\mathcal{G} \subseteq \bigcup_{i=1}^{K} \mathcal{B}(f_i, \epsilon)$, i.e., $\forall g \in \mathcal{G}$, $\exists i$ such that $\rho(g, f_i) \leq \epsilon$.

Definition 3 (Covering Number)**.** [21] According to the notations in Def.2, the covering number of $\mathcal{G}$ with radius ϵ is defined as:

$$\mathcal{N}(\epsilon; \mathcal{G}, \rho) = \min\{n : \exists \epsilon - covering\ over\ \mathcal{G}\ with\ size\ n\}$$

With the above definitions, we further have the following assumption and lemma to help us derive the generalization bound.

Assumption 1 (Basic Assumptions)**.** We assume that all the embeddings of users and items are chosen from the following embedding hypothesis space:

$$\mathcal{H}_R = \left\{\boldsymbol{g} : \boldsymbol{g} \in \mathbb{R}^d, \|\boldsymbol{g}\| \leq r\right\}, \tag{11}$$

where $\boldsymbol{g}_{u_i}^c \in \mathcal{H}_R, u_i \in \mathcal{U}, c \in [C]$ and $\boldsymbol{g}_{v_j} \in \mathcal{H}_R, v_j \in \mathcal{I}$.

Lemma 1. *[28, 25, 64] The covering number of the hypothesis class $\mathcal{H}_R$ has the following upper bound:*

$$\log \mathcal{N}(\epsilon; \mathcal{H}_R, \rho) \leq d \log\left(\frac{3r}{\epsilon}\right), \tag{12}$$

where d is the dimension of embedding space.

Based on the above introductions, we have the following results. ***Due to space limitations, please refer to Appendix.B for all proofs in detail.***

Theorem 1 (Generalization Upper Bound of DPCML)**.** *Let* $\mathbb{E}[\hat{\mathcal{L}}_{\mathcal{D}}(\boldsymbol{g})]$ *be the population risk of* $\hat{\mathcal{L}}_{\mathcal{D}}(\boldsymbol{g})$*. Then,* $\forall\ \boldsymbol{g} \in \mathcal{H}_R$*, with high probability, the following inequation holds:*

$$\left|\hat{\mathcal{L}}_{\mathcal{D}}(\boldsymbol{g}) - \mathbb{E}[\hat{\mathcal{L}}_{\mathcal{D}}(\boldsymbol{g})]\right| \leq \sqrt{\frac{2d\log\left(3r\tilde{N}\right)}{\tilde{N}}}, \tag{13}$$

where we have

$$\tilde{N} = \left(4r^2\sqrt{\left(\frac{(4+\eta)^2}{|\mathcal{U}|} + \frac{2}{|\mathcal{U}|^2}\sum_{u_i\in\mathcal{U}}\left(\frac{1}{n_i^+} + \frac{1}{n_i^-}\right)\right)}\right)^{-2}$$

Intriguingly, we see that our derived bound does not depend on C. This is consistent with the over-parameterization phenomenon [8, 34]. On top of Thm.1, we have the following corollary.

Corollary 1. *DPCML could enjoy a smaller generalization error than CML.*

Therefore, we can conclude that DPCML generalizes to unseen data better than single-vector CML and thus improves the recommendation performance. This supports the superiority of our proposed DPCML from a theoretical perspective. In addition, we also empirically demonstrate this in the experiment Sec.4.3 and Appendix.C.6.4.

Table 1: Basic Information of the Datasets. %Density is defined as $\frac{\#Ratings}{\#Users\times\#Items}\times 100\%$.

Datasets	MovieLens-1M	Steam-200k	CiteULike-T	MovieLens-10M
Domain	Movie	Game	Paper	Movie
#Users	6,034	3,757	5,219	69,167
#Items	3,953	5,113	25,975	10,019
#Ratings	575,271	115,139	125,580	5,003,437
%Density	2.4118%	0.5994%	0.0926%	0.7220%

4 Experiments

In this section, our proposed method is applied to a wide range of real-world recommendation datasets to show its superiority. ***Please refer to Appendix.C for more results about experiments***.

4.1 Experimental Setups

To begin with, we perform empirical experiments on several common recommendation benchmarks: *MovieLens-1m*, *Steam-200k*, *CiteULike* and *MovieLens-10m*. The detailed statistics in terms of these datasets are summarized in Tab.1. For the datasets with explicit feedbacks, we follow the previous works [14, 44] and transfer them into implicit feedback. Secondly, we evaluate the performance with five metrics, including **Precision** (P@N), **Recall** (R@N), **Normalized Discounted Cumulative Gain** (NDCG@N), **Mean Average Precision** (MAP), and **Mean Reciprocal Rank** (MRR). Moreover, we compared our proposed method with 14 competitive competitors: a) Item-based CF method, **itemKNN** [27]; b) MF-based algorithms, including the combination of MF and deep learning models and multi-vector MF-based approaches: **GMF**, **MLP**, **NeuMF** [14], **M2F** [12] and **MGMF** [12]; c) CML-based methods, including **UniS** [35], **PopS** [53], **2stS** [44], **HarS** [18, 11], **TransCF** [36], **LRML** [41], **AdaCML** [62] and **HLR** [43].

4.2 Overall Performance

The experimental results of all the involved competitors are shown in Tab.2 and Tab.5 (in Appendix.C.5). Consequently, we can draw the following conclusions: 1) In most cases, the best

https://doi.org/10.52202/068431-0178

Table 2: Performance comparisons on MovieLens-1m and Steam-200k datasets.

	Type	Method	P@3	R@3	NDCG@3	P@5	R@5	NDCG@5	MAP	MRR
MovieLens-1m	Item-based	itemKNN	12.24	2.90	12.41	12.43	4.29	12.79	8.34	26.16
	MF-based	GMF	14.10	2.81	14.33	14.28	4.08	14.73	8.29	29.51
		MLP	13.95	2.78	14.22	14.06	3.98	14.56	8.30	29.39
		NeuMF	16.43	3.20	16.87	16.73	4.68	17.40	9.69	33.23
		M2F	8.61	1.84	9.36	7.60	2.30	8.67	2.95	20.40
		MGMF	17.38	3.51	18.08	17.63	5.05	18.52	10.12	35.15
	CML-based	UniS	17.56	3.71	17.89	18.34	5.60	18.79	12.40	35.77
		PopS	12.96	3.11	13.30	12.82	4.41	13.40	7.59	28.61
		2stS	21.07	4.84	21.35	21.81	7.07	22.29	14.42	40.36
		HarS	24.88	5.86	25.38	24.89	8.25	25.77	15.74	45.15
		TransCF	10.03	1.84	10.31	10.90	3.09	11.20	7.07	23.66
		LRML	17.15	3.52	17.56	17.45	5.12	18.08	10.42	34.36
		AdaCML	19.06	4.12	19.31	19.74	6.23	20.20	13.30	37.36
		HLR	21.10	4.80	21.53	21.61	7.06	22.28	13.95	40.71
	Ours	DPCML1	19.12	4.14	19.34	19.90	6.27	20.29	13.24	37.55
		DPCML2	**25.18**	**6.06**	**25.64**	**25.35**	**8.51**	**26.16**	**16.09**	**45.32**
Steam-200k	Item-based	itemKNN	12.58	9.47	13.23	6.47	3.90	7.23	11.74	23.33
	MF-based	GMF	12.57	6.17	13.29	14.22	6.86	15.39	9.72	28.38
		MLP	17.07	9.63	17.49	16.89	8.49	17.67	15.15	34.54
		NeuMF	17.36	9.65	17.95	17.41	8.79	18.45	15.11	35.55
		M2F	11.33	5.69	11.95	11.44	5.73	12.98	6.43	25.05
		MGMF	12.51	6.14	13.25	14.45	6.88	15.55	9.63	28.40
	CML-based	UniS	20.71	11.97	21.42	20.92	10.36	21.61	18.88	40.10
		PopS	18.05	11.58	18.76	14.94	7.98	15.78	15.13	34.04
		2stS	25.20	14.62	26.20	23.97	11.91	25.35	21.48	46.17
		HarS	26.66	15.74	27.93	24.94	12.78	26.63	23.25	48.84
		TransCF	13.30	6.61	13.58	15.26	7.09	15.89	11.08	26.29
		LRML	14.91	7.48	15.43	16.49	8.06	17.51	12.24	31.89
		AdaCML	23.02	13.19	23.38	22.35	11.31	23.23	19.88	42.03
		HLR	20.30	11.65	20.96	19.79	9.88	20.94	17.06	39.26
	Ours	DPCML1	25.39	14.84	26.56	23.88	12.11	25.25	22.26	46.79
		DPCML2	**29.88**	**17.13**	**31.22**	**28.70**	**14.51**	**30.56**	**24.10**	**51.95**

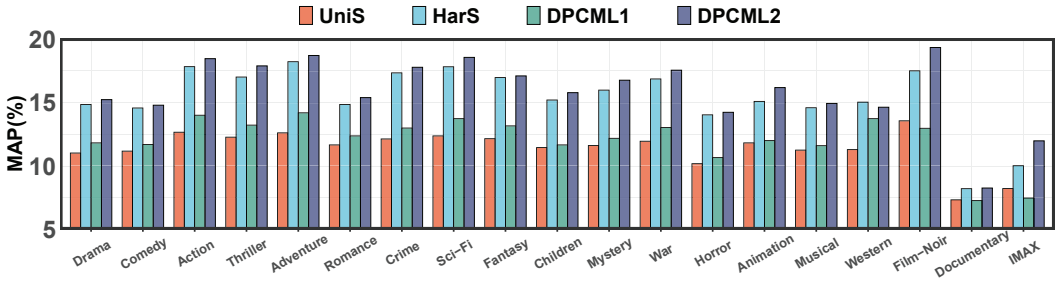


Figure 4: Fine-grained performance over each interest group on MovieLens-10m dataset.

performance of CML-based methods consistently surpasses the best MF-based competitors. This suggests that it is necessary to develop CML-based RS algorithms. 2) Our proposed method consistently surpasses all the competitors significantly on all datasets, except the results for MAP and MRR on CiteULike. Even for the failure results, the performance is fairly competitive compared with the competitors. This shows the effectiveness of our proposed algorithm. 3) Compared with studies targeting joint accessibility (i.e., M2F and MGMF), our proposed method significantly outperforms M2F and MGMF on all benchmark datasets. This shows the advantage of the CML-based paradigm that deserves more attention along this direction in future work.

4.3 Quantitative Analysis

Fine-grained Performance Comparison. Fig.4 presents the MAP metric over each interest group (movie genre) on MovieLens-10m. We can observe that our proposed framework could not only significantly outperform their single-vector counterparts in the majority interests but also improve the performance of minority groups in most cases. Especially, compared with HarS, the performance

https://doi.org/10.52202/068431-0178

Table 3: The diversity performance comparison of CML-based algorithms on Steam-200k and MovieLens-1m datasets. Here a higher value implies more diverse recommendation results.

Method	MaxDiv@3	MaxDiv@5	MaxDiv@10	MaxDiv@20
Steam-200k				
UniS	1.354	4.750	23.520	117.927
HarS	1.752	6.809	40.378	236.794
DPCML1 w/o DCRS	1.643	5.857	30.425	155.193
DPCML1	1.822	6.713	34.727	179.065
DPCML2 w/o DCRS	2.958	11.398	65.398	365.458
DPCML2	2.977	11.472	65.952	369.876
MovieLens-1m				
UniS	1.739	6.142	30.127	140.095
HarS	2.443	8.826	46.390	244.078
DPCML1 w/o DCRS	1.623	5.857	29.500	140.057
DPCML1	1.744	6.195	30.755	145.615
DPCML2 w/o DCRS	2.827	10.423	55.612	292.089
DPCML2	3.144	11.498	60.696	313.086

improvement of DPCML2 on minority interests is sharp. This shows that DPCML could reasonably focus on potentially interesting items even with the imbalanced item distribution.

Recommendation Diversity Evaluation. We test the performance of DPCML against CML-based competitors with *max-sum diversification (MaxDiv)* [4]. The diversity results are shown in Tab.3. We observe that: a) For methods within the same negative sampling strategy (i.e., UniS and DPCML1, HarS and DPCML2), our proposed DPCML could achieve relatively higher max-sum values. This suggests the improvement of DPCML in terms of promoting recommendation diversity. b) In most cases (except for DPCML1 w/o DCRS on the MovieLens-1m dataset), DPCML outperforms other competitors even without regularization. c) Most importantly, equipped with the regularization term DCRS, DPCML could achieve better diversification results against w/o DCRS.

Effect of the Diversity Control Regularization. Fig.5 illustrates a 3D-barplot based on the results of grid search on Steam-200k. From the results, we can observe that the proposed regularization scheme could significantly boost performance on all metrics. Moreover, there would induce different performances with different diversity values. This suggests that controlling a proper diversity of the embeddings for the same user is essential to accommodate their preferences better.

Empirical Justification of Corol.1. Fig.7 shows the empirical results on Steam-200k dataset. Based on these results, we can see that, with the increase of C, the empirical risk (i.e., training loss) of DPCML ($C > 1$) is significantly smaller than CML ($C = 1$). In addition, DPCML could substantially improve the performance of the validation/test set. Thus, we can conclude that DPCML could induce a smaller generalization error than traditional CML. This is consistent with Corol.1.

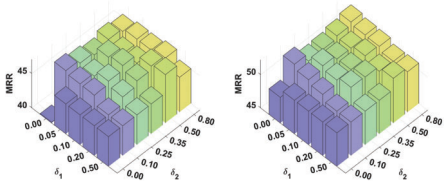
Sensitive Analysis of C. Fig.8 demonstrates the performance of DPCML methods with different C on Steam-200k dataset. We observe that a proper C could significantly improve the performance. Besides, leveraging C too aggressively for DPCML2 may adversely hurt the performance since models optimized with hard samples are more likely to lead to the over-fitting problem with the increasing parameters.

Sensitivity analysis of η. We investigate the sensitivity of $\eta \in \{0, 1, 3, 5, 10, 20, 30\}$ for recommendation results on the Steam-200k dataset. The experimental results are listed in Tab.6 and Tab.7. We can conclude that a proper η (roughly 10) could significantly improve the performance, suggesting the essential role of the proposed diversity control regularization scheme.

Training Efficiency. Since DPCML includes multiple user representations, it will inevitably introduce extra complexity to the overall optimization. We further investigate the efficiency of our proposed algorithm, presented in Fig.6. This trend suggests that our proposed algorithm could achieve competitive performance with acceptable efficiency.

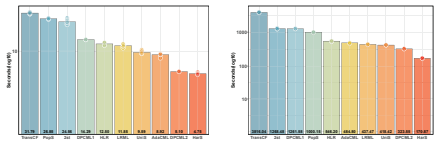
Ablation Studies of DCRS. In order to show the effectiveness of our proposed DCRS, we compare its performance with the its three variants: **a) w/o DCRS**. **b) DCRS** $- \delta_1$ and **c) DCRS** $- \delta_2$. Please refer to Appendix.C.6.8 to the details of each variant. The empircal results on Steam-200k dataset are provided in Tab.8 and Tab.9. From the above results, we can see that: In most cases, only

https://doi.org/10.52202/068431-0178



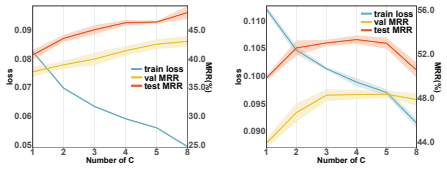
(a) DPCML1 (MRR) (b) DPCML2 (MRR)

Figure 5: Sensitivity analysis about δ_1 and δ_2 on Steam-200k datasets.



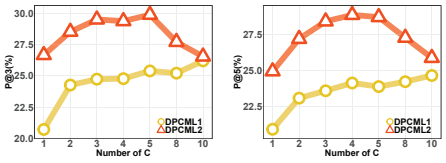
(a) CiteULike (b) MovieLens-10m

Figure 6: Training efficiency comparison among CML-based competitors.



(a) DPCML1 (b) DPCML2

Figure 7: Empirical justification of Corol.1



(a) P@3 (b) P@5

Figure 8: Sensitive Analysis of different C.

employing one of the two terms of DCRS could still improve the recommendation performance. However, none of them could outperform our proposed method. This strengthens the effectiveness of our proposed regularization scheme.

DCRS for MF-based Systems. We attempt to apply the proposed diversity control regularization scheme (DCRS) for M2F [52, 12]. In addition, we further explore the effectiveness of DCRS for the general framework of joint accessibility (GFJA, Eq.(9) in the main paper). The experimental results are summarized in Tab.10. The overall performance suggests the superiority of our proposed method against the current multi-vector-based competitors.

5 Conclusion & Future Remarks

This paper pays attention to developing an effective CML-based algorithm when users have multiple categories of interests. First, we point out that the current CML framework might induce preference bias, especially when the item category distribution is imbalanced. To this end, we propose a novel algorithm called DPCML. The key idea is to include multiple representations for each user in the model design. Meanwhile, a novel *diversity control regularization* scheme is specifically tailored to serve our purpose better. To see the generalization ability of DPCML on unseen test data, we also provide high probability upper bounds for the generalization error. Finally, the experiments over a range of benchmark datasets speak to the efficacy of DPCML. However, following the paradigm of CML, a possible limitation of DPCML is that it generally applies to implicit feedback but not explicit feedback since CML only cares about the relative preference ranking instead of concrete magnitude. In the future, we will explore how to improve the recommendation diversity based on explicit feedback.

6 Acknowledgements

This work was supported in part by the National Key R&D Program of China under Grant 2018AAA0102000, in part by National Natural Science Foundation of China: U21B2038, 61931008, 62025604, 61971016, 6212200758 and 61976202, in part by the Fundamental Research Funds for the Central Universities, in part by Youth Innovation Promotion Association CAS, in part by the Strategic Priority Research Program of Chinese Academy of Sciences, Grant No.XDB28000000, in part by the China National Postdoctoral Program for Innovative Talents under Grant BX2021298, and in part by China Postdoctoral Science Foundation under Grant 2022M713101.

References

[1] Askari, B., Szlichta, J., and Salehi-Abari, A. Variational autoencoders for top-k recommendation with implicit feedback. In *SIGIR*, pp. 2061–2065, 2021.

[2] Bao, S., Xu, Q., Ma, K., Yang, Z., Cao, X., and Huang, Q. Collaborative preference embedding against sparse labels. In *ACM MM*, pp. 2079–2087, 2019.

[3] Bartlett, P. L. and Mendelson, S. Rademacher and gaussian complexities: Risk bounds and structural results. In *COLT*, volume 2111, pp. 224–240, 2001.

[4] Borodin, A., Lee, H. C., and Ye, Y. Max-sum diversification, monotone submodular functions and dynamic updates. In *SIGMOD/PODS*, pp. 155–166, 2012.

[5] Canévet, O. and Fleuret, F. Efficient sample mining for object detection. In *ACML*, pp. 48–63, 2014.

[6] Chen, J., Lian, D., and Zheng, K. Improving one-class collaborative filtering via ranking-based implicit regularizer. In *AAAI*, pp. 37–44, 2019.

[7] Curmei, M., Dean, S., and Recht, B. Quantifying availability and discovery in recommender systems via stochastic reachability. In *ICML*, pp. 2265–2275, 2021.

[8] Dar, Y., Muthukumar, V., and Baraniuk, R. G. A farewell to the bias-variance tradeoff? an overview of the theory of overparameterized machine learning. 2021.

[9] Dean, S., Rich, S., and Recht, B. Recommendations and user agency: the reachability of collaboratively-filtered information. In *FAT*, pp. 436–445, 2020.

[10] Ding, J., Quan, Y., He, X., Li, Y., and Jin, D. In *IJCAI*, pp. 2230–2236, 2019.

[11] Gajic, B., Amato, A., and Gatta, C. Fast hard negative mining for deep metric learning. *Pattern Recognition*, 112:107795, 2021.

[12] Guo, W., Krauth, K., Jordan, M. I., and Garg, N. The stereotyping problem in collaboratively filtered recommender systems. In *EAAMO*, pp. 6:1–6:10, 2021.

[13] He, X., Zhang, H., Kan, M., and Chua, T. Fast matrix factorization for online recommendation with implicit feedback. In *SIGIR*, pp. 549–558, 2016.

[14] He, X., Liao, L., Zhang, H., Nie, L., Hu, X., and Chua, T. Neural collaborative filtering. In *WWW*, pp. 173–182, 2017.

[15] He, X., Du, X., Wang, X., Tian, F., Tang, J., and Chua, T. Outer product-based neural collaborative filtering. In *IJCAI*, pp. 2227–2233, 2018.

[16] Heckel, R. and Ramchandran, K. The sample complexity of online one-class collaborative filtering. In *ICML*, pp. 1452–1460, 2017.

[17] Henriques, J. F., Carreira, J., Caseiro, R., and Batista, J. Beyond hard negative mining: Efficient detector learning via block-circulant decomposition. In *ICCV*, pp. 2760–2767, 2013.

[18] Hsieh, C.-K., Yang, L., Cui, Y., Lin, T.-Y., Belongie, S., and Estrin, D. Collaborative metric learning. In *WWW*, pp. 193–201, 2017.

[19] Jiang, M., Cui, P., Chen, X., Wang, F., Zhu, W., and Yang, S. Social recommendation with cross-domain transferable knowledge. *IEEE TKDE*, 27(11):3084–3097, 2015.

[20] Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[21] Ledoux, M. and Talagrand, M. *Probability in Banach Spaces: isoperimetry and processes*. 1991.

[22] Lee, D., Kang, S., Ju, H., Park, C., and Yu, H. Bootstrapping user and item representations for one-class collaborative filtering. In *SIGIR*, pp. 1513–1522, 2021.

https://doi.org/10.52202/068431-0178

[23] Lei, Y., Ding, L., and Bi, Y. Local rademacher complexity bounds based on covering numbers. *Neurocomputing*, 218:320–330, 2016.

[24] Lei, Y., Ledent, A., and Kloft, M. Sharper generalization bounds for pairwise learning. In *NeurIPs*, 2020.

[25] Li, S. and Liu, Y. Sharper generalization bounds for clustering. In *ICML*, pp. 6392–6402, 2021.

[26] Li, Z., Xu, Q., Jiang, Y., Ma, K., Cao, X., and Huang, Q. Neural collaborative preference learning with pairwise comparisons. *IEEE Trans. Multim.*, 23:1977–1989, 2021.

[27] Linden, G., Smith, B., and York, J. Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing*, (1):76–80, 2003.

[28] Long, P. M. and Sedghi, H. Generalization bounds for deep convolutional neural networks. In *ICLR 2020*, 2020.

[29] Lv, Y., Zheng, Y., Wei, F., Wang, C., and Wang, C. AICF: attention-based item collaborative filtering. *Adv. Eng. Informatics*, 44:101090:1–11, 2020.

[30] Ma, J., Zhou, C., Cui, P., Yang, H., and Zhu, W. Learning disentangled representations for recommendation. In *NeurIPS*, pp. 5712–5723, 2019.

[31] Ma, J., Zhou, C., Yang, H., Cui, P., Wang, X., and Zhu, W. Disentangled self-supervision in sequential recommenders. In *KDD*, pp. 483–491, 2020.

[32] McDiarmid, C. Concentration. In *Probabilistic methods for algorithmic discrete mathematics*, pp. 195–248. 1998.

[33] Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of Machine Learning*. MIT Press, 2012.

[34] Nakkiran, P., Kaplun, G., Bansal, Y., Yang, T., Barak, B., and Sutskever, I. Deep double descent: Where bigger models and more data hurt. In *ICLR*, 2020.

[35] Pan, R., Zhou, Y., Cao, B., Liu, N. N., Lukose, R. M., Scholz, M., and Yang, Q. One-class collaborative filtering. In *ICDM*, pp. 502–511, 2008.

[36] Park, C., Kim, D., Xie, X., and Yu, H. Collaborative translational metric learning. In *ICDM*, pp. 367–376, 2018.

[37] Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. Automatic differentiation in pytorch. 2017.

[38] Rendle, S. and Freudenthaler, C. Improving pairwise learning for item recommendation from implicit feedback. In *WSDM*, pp. 273–282, 2014.

[39] Rendle, S., Freudenthaler, C., Gantner, Z., and Schmidt-Thieme, L. BPR: bayesian personalized ranking from implicit feedback. In *UAI*, pp. 452–461, 2009.

[40] Takács, G. and Tikk, D. Alternating least squares for personalized ranking. In *RecSys*, pp. 83–90, 2012.

[41] Tay, Y., Tuan, L. A., and Hui, S. C. Latent relational metric learning via memory-based attention for collaborative ranking. In *WWW*, pp. 729–739, 2018.

[42] Togashi, R., Kato, M., Otani, M., and Satoh, S. Density-ratio based personalised ranking from implicit feedback. In *WWW*, pp. 3221–3233, 2021.

[43] Tran, V., Salha-Galvan, G., Hennequin, R., and Moussallam, M. Hierarchical latent relation modeling for collaborative metric learning. In *RecSys*, pp. 302–309, 2021.

[44] Tran, V.-A., Hennequin, R., Royo-Letelier, J., and Moussallam, M. Improving collaborative metric learning with efficient negative sampling. In *SIGIR*, pp. 1201–1204, 2019.

[45] Wang, C., Zhou, T., Chen, C., Hu, T., and Chen, G. Off-policy recommendation system without exploration. In *PAKDD*, volume 12084, pp. 16–27. Springer, 2020.

[46] Wang, H., Chen, B., and Li, W. Collaborative topic regression with social regularization for tag recommendation. In *IJCAI*, pp. 2719–2725, 2013.

[47] Wang, H., Li, Y., and Frimpong, F. Group recommendation via self-attention and collaborative metric learning model. *IEEE Access*, 7:164844–164855, 2019.

[48] Wang, M., Gong, M., Zheng, X., and Zhang, K. Modeling dynamic missingness of implicit feedback for recommendation. In *NeurIPS*, pp. 6670–6679, 2018.

[49] Wang, W., Feng, F., He, X., Nie, L., and Chua, T. Denoising implicit feedback for recommendation. In *WSDM*, pp. 373–381, 2021.

[50] Wang, X., He, X., Wang, M., Feng, F., and Chua, T. Neural graph collaborative filtering. In *ACM SIGIR*, pp. 165–174, 2019.

[51] Wang, X., Wang, R., Shi, C., Song, G., and Li, Q. Multi-component graph convolutional collaborative filtering. In *AAAI*, pp. 6267–6274, 2020.

[52] Weston, J., Weiss, R. J., and Yee, H. Nonlinear latent factorization by embedding multiple user interests. In *RecSys*, pp. 65–68, 2013.

[53] Wu, G., Volkovs, M., Soon, C. L., Sanner, S., and Rai, H. Noise contrastive estimation for one-class collaborative filtering. In *SIGIR*, pp. 135–144, 2019.

[54] Wu, H., Zhou, Q., Nie, R., and Cao, J. Effective metric learning with co-occurrence embedding for collaborative recommendations. *Neural Networks*, 124:308–318, 2020.

[55] Wu, Q., Zhang, H., Gao, X., Yan, J., and Zha, H. Towards open-world recommendation: An inductive model-based collaborative filtering approach. In *ICML*, pp. 11329–11339, 2021.

[56] Xu, D., Ruan, C., Körpeoglu, E., Kumar, S., and Achan, K. Rethinking neural vs. matrix-factorization collaborative filtering: the theoretical perspectives. In *ICML*, pp. 11514–11524, 2021.

[57] Yang, T. and Ying, Y. AUC maximization in the era of big data and AI: A survey. *ACM Computing Surveys*, 2022.

[58] Yang, Z., Ding, M., Zhou, C., Yang, H., Zhou, J., and Tang, J. Understanding negative sampling in graph representation learning. In *KDD*, pp. 1666–1676, 2020.

[59] Yao, Y., Tong, H., Yan, G., Xu, F., Zhang, X., Szymanski, B. K., and Lu, J. Dual-regularized one-class collaborative filtering with implicit feedback. *WWW*, 22(3):1099–1129, 2019.

[60] Zhang, Q. and Ren, F. Prior-based bayesian pairwise ranking for one-class collaborative filtering. *Neurocomputing*, 440:365–374, 2021.

[61] Zhang, Q. and Ren, F. Double bayesian pairwise learning for one-class collaborative filtering. *Knowl. Based Syst.*, 229:107339, 2021.

[62] Zhang, T., Zhao, P., Liu, Y., Xu, J., Fang, J., Zhao, L., Sheng, V. S., and Cui, Z. Adacml: Adaptive collaborative metric learning for recommendation. In *DASFAA*, volume 11447, pp. 301–316, 2019.

[63] Zheng, Y., Tang, B., Ding, W., and Zhou, H. A neural autoregressive approach to collaborative filtering. In *ICML*, pp. 764–773, 2016.

[64] Zhou, D. The covering number in learning theory. *J. Complex.*, 18(3):739–767, 2002.

[65] Zhou, X., Liu, D., Lian, J., and Xie, X. Collaborative metric learning with memory network for multi-relational recommender systems. In *IJCAI*, pp. 4454–4460, 2019.

 https://doi.org/10.52202/068431-0178

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [N/A]
   (c) Did you discuss any potential negative societal impacts of your work? [No] There are no immediate societal impacts associated with our research.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] See the supplementary materials.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Sec.C.4 in the Appendix.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes] See Sec.C
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [No] We use open-source software and datasets.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

https://doi.org/10.52202/068431-0178