

Article

DTGCF: Diversified Tag-Aware Recommendation with Graph Collaborative Filtering

Yi Zuo, Shengzong Liu * and Yun Zhou

School of Information Technology and Management, Hunan University of Finance and Economics,
Changsha 410205, China

* Correspondence: lsz@hufe.edu.cn

Abstract: In tag-aware recommender systems, users are strongly encouraged to utilize arbitrary tags to mark items of interest. These user-defined tags can be viewed as a bridge linking users and items. Most tag-aware recommendation models focus on improving the accuracy by introducing ingenious design or complicated structures to handle the tagging information appropriately. Beyond accuracy, diversity is considered to be another important indicator affecting the user satisfaction. Recommending more diverse items will provide more interesting items and commercial sales. Therefore, we propose a diversified tag-aware recommendation model based on graph collaborative filtering. The proposed model establishes a generic graph collaborative filtering framework tailored for tag-aware recommendations. To promote diversity, we adopt two modules: personalized category-boosted negative sampling to select a certain proportion of similar but negative items as negative samples for training, and adversarial learning to make the learned item representation category-free. To improve accuracy, we conduct a two-way TransTag regularization to model the relationship among users, items, and tags. Blending these modules into the proposed framework, we can optimize both the accuracy and diversity concurrently in an end-to-end manner. Experiments on Movielens datasets show that the proposed model can provide diverse recommendations while maintaining a high level of accuracy.

Keywords: accuracy; diversity; graph collaborative filtering; tag-aware recommendation



Citation: Zuo, Y.; Liu, S.; Zhou, Y.
DTGCF: Diversified Tag-Aware
Recommendation with Graph
Collaborative Filtering. *Appl. Sci.*
2023, *13*, 2945. <https://doi.org/10.3390/app13052945>

Academic Editors: Chuan Shi, Cheng
Yang, Yong Fang and Lichao Sun

Received: 26 January 2023

Revised: 20 February 2023

Accepted: 22 February 2023

Published: 24 February 2023



Copyright: © 2023 by the authors.
Licensee MDPI, Basel, Switzerland.
This article is an open access article
distributed under the terms and
conditions of the Creative Commons
Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Tag-aware recommender systems (TRS), also known as folksonomies, have been successfully applied in a wide range of web services and applications, such as books, music, movies, and videos. The core character of TRS is that users are encouraged to mark any items with arbitrary tags [1]. These user-defined tags are generally constituted by concise words or phrases, which can reflect the behavioral habits and preferences of target users. Meanwhile, items will be annotated with various tags defined by different users. Accordingly, these tags can also describe the characteristics of items from multiple angles. In this way, the implicit relationship between users and items is established through tags. More importantly, these tags contain more plentiful information than ratings and are more direct and feasible than reviews.

The integration of social tagging information brings benefits to the recommender systems, but also results in many new issues to be processed. For instance, since any word or phrase can be taken as a tag, this inevitably causes redundancy and ambiguity in the tag potential space [2,3]. To guarantee the recommendation performance, customization strategies should be taken into account to address the tagging information in a reasonable and effective way. At the early stage, many researchers have attempted to exploit traditional machine learning methods, including tensor-based techniques [4], clustering-based techniques [5], and graph-based techniques [6]. However, they are usually shallow structures, which are insufficient to learn the rich semantic information from various tags.

Owing to its strong potential of feature extraction, deep learning has been widely used in various fields [7,8]. In recent years, it has also been introduced into TRS to extract more useful tag-based features of users and items in recent years. For example, ACF [3] leverages the deep autoencoders to learn low-dimensional user features based on tags. DSPR [9] employs the multi-layer perceptron (MLP) to deal with tagging information for capturing more abstract user and item representation. TRSDL [10] even establishes two different deep architectures: a deep neural network for learning item features and a recurrent neural network for user features. Furthermore, TNAM [11] and AIRec [12] integrate the attention network to capture tag-based features representing multi-aspect user preferences or item characteristics. In these feature-based models, users and items are generally represented by the associated tags. However, the tagging behaviors are rather sparse in the real scenario, leading to performance degradation. To cope with this problem, TGCN [13] and LFGCF [14] adopt the graph convolutional network (GCN) to aggregate more information from multi-hop neighbors. Moreover, the topological structure can also be learned to further address the problem of ambiguity and redundancy in TRS.

With the advancement of tag-aware recommendations, more complicated structures are established to boost the recommendation accuracy. In most cases, the accuracy is the only target or the dominant target to be optimized. However, an accurate recommendation cannot indicate a satisfactory one in real applications [15]. For example, a user who prefers rock music can easily obtain a recommendation list full of rock music to guarantee accuracy. Since this rock fan knows rock music well and has even heard most of it, this accurate recommendation may be not valuable. In addition, recommendations full of similar items will make users feel bored and dissatisfied. Reinforced through repeated exposure to similar contents, the users will be trapped in a narrow scope of topics, resulting in the echo chamber or filter bubble phenomenon [16]. A good recommender system should have the capability to identify items of interest that users cannot find themselves. Therefore, diversified recommendations, aiming at providing items with a variety of choices and novel experiences, are receiving increasing attention to break the echo chamber [17]. The recommendation diversity is useful to individual users as well as businesses. Users can access a wider range of contents and find more interest and novel things. For the e-commerce enterprises, diverse recommendations will increase their product sales and promote user satisfaction and user loyalty [18].

However, the improvement of recommendation diversity is inevitably at the cost of the decrease of accuracy. Generally speaking, it is easy to promote diversity by recommending unpopular items selected randomly, which will greatly reduce the accuracy. Therefore, the core of diversified recommendation is to trade-off the accuracy and diversity. Early works often utilize a two-stage scheme, which first adopts an accuracy-focused approach to yield the candidate item set and then selects a list of items by a re-ranking or post-processing strategy, such as clustering [19], MMR [20,21] and DPP [22]. Since the candidate generation and post-processing are two independent processes, the diversified strategy is decoupled from the model optimization. The recommendation performance will mainly depend on the user and item representations learned in the upstream matching process. It is uncertain whether the decoupling design can diversify the recommendation with acceptable accuracy loss [23]. For this reason, recent works have turned to constructing an end-to-end (one-stage) supervised learning framework to improve diversity [23–25].

In this work, we focus on providing diversified tag-aware recommendations with minimal costs on accuracy. To this end, we propose a diversified tag-aware recommendation algorithm based on graph collaborative filtering (DTGCF). We establish a tag-aware GCN to learn representation of users, items, and tags. The advantage of the proposed GCN is two-fold. On the one hand, it can combine the node features of neighbors and topological structures to generate more accurate and reasonable node representations, thus improving the recommendation accuracy. On the other hand, more diverse items are able to be exposed to users due to the inherent high-order connectivity in the constructed graph, which can boost the recommendation diversity. Specifically, we first transform the user-item-tag

tensor into two graphs: user–tag graph and item–tag graph. Inspired by LightGCN [26], we then establish a light tag-aware GCN to obtain node representations from the constructed two graphs.

To further improve the accuracy, we employ a two-way TransTag regularization to model the relationship among users, items, and tags. To promote the recommendation diversity, we design personalized category-boosted negative sampling to choose a certain proportion of similar but negative items as the negative training set, where the probability of selecting these items for each user is adaptively determined by his propensity to diversity. In addition, we adopt adversarial learning to make the item representation category-free in the learned embedding space. In this way, items from diverse categories are more likely to be recommended. Combining these strategies, the proposed DTGCF can provide recommendations with respect to accuracy and diversity within an end-to-end framework.

The main contributions of our work are given as follows:

- We propose a novel diversified tag-aware recommendation model based on GCN, which combines the optimization process and the diversified strategies within an end-to-end framework;
- To balance the recommendation accuracy and diversity, we elaborately design three modules. Among them, personalized category-boosted negative sampling and adversarial learning are tailored for promoting diversity, while the two-way TransTag regularization function is well-designed for accuracy;
- We conduct extensive experiments on real-world datasets to show the superiority of the proposed DTGCF on diversified tag-aware recommendation.

The remainder of this paper is organized as follows: Section 2 briefly reviews the related work. Some preliminaries are presented in Section 3. Section 4 details the proposed DTGCF. Experiments are established in Section 5. Finally, we give some conclusions in Section 6.

2. Related Work

As a powerful tool for providing more efficient and effective choices, recommender systems (RSs) have attracted more and more attention in recent years. Traditional RSs, including collaborative filtering based RS [27], content-based RS [28], and hybrid RS [29], have been extensively used in many fields. Furthermore, a variety of new RS research areas have emerged with great success, such as conversational RS [30], knowledge graph-based RS [31], session-based RS [32], and tag-aware RS [1]. In our work, we focus on diversified tag-aware recommendation based on graph collaborative filtering. Therefore, we only summarize the related work on tag-aware recommendation, GCN-based recommendation, and diversified recommendation in this section.

2.1. Tag-Aware Recommendation

At the beginning, many efforts have extended the traditional recommendation solutions to TRS. For example, researchers modified the classic collaborative filtering (CF) to generate tag-aware recommendations [2,33,34]. To deal with the problem of ambiguity and redundancy in the tagging information, Shepitsen et al. [5] employed the clustering method to aggregate redundant tags. Li et al. [35] devised a generic framework based on Bayesian personalized ranking (BPR), in which a tag mapping scheme is used to alleviate the problem of high dimension and sparsity of tagging information. To solve the sparsity problem, Pan et al. [36] proposed a social tag expansion model that explores the relationship between tags and assigns appropriate weights to expanded tags.

Recent works show the superiority of neural network-based methods on addressing social tagging information. Zuo et al. [3] leveraged the deep autoencoders to learn low-dimensional dense features of users based on tags and conducts a user-based collaborative filtering (CF) to produce recommendations. Xu et al. [9] devised a deep-semantic similarity-based neural network to capture more accurate tag-based user and item representation. Liang et al. [10] combined two different deep networks: a deep neural network to extract

nonlinear features of items and a recurrent neural network to model user's dynamic changes of interest. Huang et al. [11] introduced a neural attention network to automatically assign tag weights for personalized users. Chen et al. [12] constructed a hierarchical attention network to extract multi-aspect user representations and devised an intersection module to learn conjunct features from the intersection of user and item tags.

2.2. GCN-Based Recommendation

Due to its powerful capability of representation learning on graph-structured data, GCN has been introduced to recommender systems to learn more robust representations for users and items. Berg et al. [37] first leveraged GCN on the user-item interaction bipartite graph for matrix completion. Ying et al. [38] developed a data-efficient GCN-based recommendation model, which performs random walks to structure the convolutions efficiently. NGCF [39] improves recommendation performance by explicitly modeling the high-order connectivity in the user-item graph. LightGCN [26] eliminates the linear transformation and the nonlinear activation function in NGCF to form a simplified GCN, which is validated to be effective and efficient. Naturally, GCN has been introduced into TRS to boost tag-aware recommendations. Chen et al. [13] developed a novel tag graph convolutional network, which fully makes use of semantic information of multi-hop neighbors in the constructed user-tag-item graph. Zhang et al. [14] proposed a light tag-aware GCN to learn the high-order representations of users and items, further improving the recommendation accuracy.

2.3. Diversified Recommendation

A good recommender system should have the capability of providing diverse recommendations. Ziegler et al. [40] first introduced diversification in improving recommendation lists to promote user satisfaction. Qin et al. [41] formulated the diversified recommendation as a linear combination of a rating function and an entropy regularizer. Zhou et al. [42] integrated an accuracy-focused method and a diversity-promoted strategy to tackle the accuracy-diversity dilemma. Ashkan et al. [43] performed a re-ranking procedure by maximizing the designed diversity-weighted utility function. Sha et al. [44] combined the accuracy, coverage, and diversity of recommendations to form a combinatorial optimization problem, which is optimized by an efficient greedy approach. With the capability of evaluating the overall diversity of the recommendation list, determinantal point process (DPP) was widely introduced as a re-ranking strategy to yield diverse item lists. In order to realize DPP in an efficient way, several methods were proposed to reduce the computational expense [22,45,46]. In addition to the re-ranking strategy, several researchers focused on developing an end-to-end to simultaneously consider the recommendation accuracy and diversity. For example, Liang et al. [24] adopted a generalized bilateral branch network to promote both domain-level and user-level diversity. Zheng et al. [23] proposed a diversified GCN-based recommendation algorithm, which carefully designs three diversity-promoted strategies including rebalanced neighbor discovering, category-boosted negative sampling, and adversarial learning. Yang et al. [25] devised a diversified recommendation model based on LightGCN, where additional diversity-promoted modules are well-designed.

To summarize, most tag-aware recommendation algorithms focus on improving recommendation accuracy. These algorithms deal with complex tagging data by introducing clever designs or complex structures. However, in addition to accuracy, diversity is considered to be another important indicator of user satisfaction. At the same time, in order to maintain high accuracy while promoting diversity, we introduce graph convolution network to process tagging information.

3. Preliminaries

In a typical TRS, there are three sets of entities: a user set $U = \{u_1, \dots, u_M\}$, an item set $I = \{i_1, \dots, i_N\}$, and a tag set $T = \{t_1, \dots, t_L\}$, where M , N , and L are sizes of the corresponding set. Different from the traditional recommendation systems, the user-item

interactions in TRS can be represented by a ternary relation set: $S \subseteq U \times I \times T$, where an element $(u, i, t) \in S$ means that user u has marked item i with the tag t . Mathematically, we often represent the ternary relation as a three-order tensor $Y \in \mathbb{R}^{M \times N \times L}$, whose element is defined as:

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

where $y_{u,i,t} = 1$ means that user u has interacted with item i . Let R be the user-item interaction matrix, where $R(u, i) = 1$ if the tagging behavior between user u and item i is observed; otherwise, $R(u, i) = 0$. The positive samples for training can be easily obtained by selecting one pair (u, i) from the observed user-item interaction set.

In our work, the user-item-tag tensors are first decomposed into two adjacent matrices: user-tag matrix and item-tag matrix, as shown in Figure 1. To be specific, we define the set of observed user-tag interactions Ω_{ut} and item-tag interactions Ω_{it} as:

$$\begin{aligned} \Omega_{ut} &= \{(u, t) \mid \exists i \in I : (u, i, t) \in S\}, \\ \Omega_{it} &= \{(i, t) \mid \exists u \in U : (u, i, t) \in S\}. \end{aligned} \quad (2)$$

Each entry $e_{u,t}$ in the user-tag matrix is set as: $e_{u,t} = 1$ if $(u, t) \in \Omega_{ut}$, and 0, otherwise. Analogously, we set $e_{i,t} = 1$ in the item-tag matrix as: $e_{i,t} = 1$ if $(i, t) \in \Omega_{it}$, and 0, otherwise. According to the corresponding matrix, we can then establish two undirected bipartite graphs: user-tag graph $G_u = (U \cup T, \Omega_{ut})$ and item-tag graph $G_i = (I \cup T, \Omega_{it})$. Note that user-item interactions are constructed by the related tagging behaviors. The task of our work is to mine the implicit feedback in tagging records, explore tag-based graph collaborative information, and learn a recommendation model for providing satisfactory recommendations to each user.

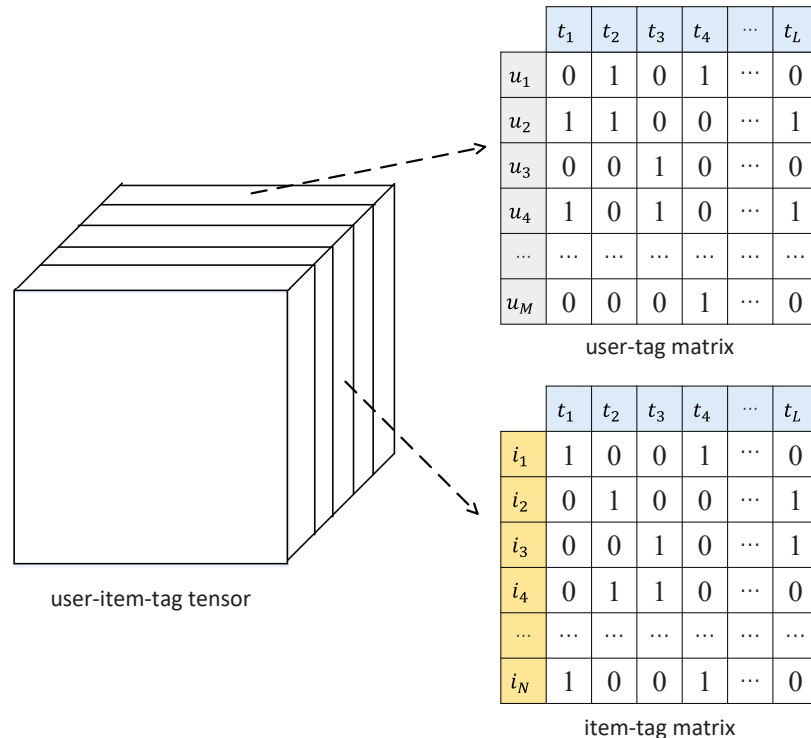


Figure 1. Illustration of decomposing the user-item-tag tensor into two adjacent matrices: user-tag matrix and item-tag matrix.

4. Methodology

In this section, we present the details of the proposed DTGCF. Specifically, we first construct the user-tag graph and the item-tag graph from the whole tagging behavior

data. Then, we establish the tag-aware GCN to extract more abstract representation for users, items, and tags by aggregating rich information from multi-hop neighbors. Based on these representations, the proposed DTGCF finally provides personalized recommendations that consider both accuracy and diversity. The holistic architecture of the proposed DTGCF is presented in Figure 2. Moreover, to generate accurate and diverse ranked item lists, we design the personalized category-boosted negative sampling, two-way TransTag regularization, and adversarial learning for tag-aware GCN, which are elaborated on the following subsections.

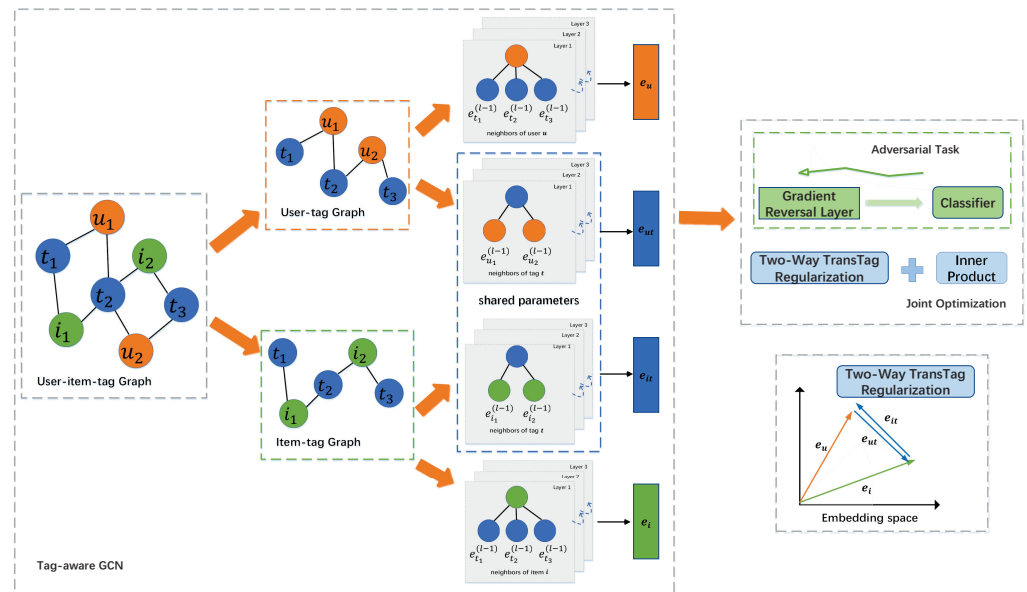


Figure 2. The overall structure of the proposed DTGCF. The user–item–tag tensor is first transformed into two undirected bipartite graphs: user–tag graph and item–tag graph. Then, a tag-aware GCN is established to generate representations of users, items, and tags, respectively. Finally, the proposed model is trained in an end-to-end way with a joint learning framework, which combines the recommendation task with an adversarial task and a two-way TransTag regularization. The lower right corner of the figure illustrates the two-way TransTag regularization which models the relationship among users, items, and tags.

4.1. Tag-Aware GCN

As introduced previously, the basic idea of GCN is to learn node representation by executing graph convolution iteratively to aggregate the features of neighbors. Based on the user–tag graph and the item–tag graph, we construct a tag-aware GCN which is composed of three main components: node embedding, graph convolution, and model optimization.

4.1.1. Node Embedding

In our tag-aware GCN, nodes includes users, items, and tags. Following the commonly embedding method used in recommender systems [26,47], we map the one-hot ID feature of each node to a dense vector. For simplicity and convenience, the embedded dimensions of different types of nodes are set to the same value. Accordingly, we can define the node embedding as the following look-up table:

$$E = \{e_{u1}, \dots, e_{uM}, e_{i1}, \dots, e_{iN}, e_{t1}, \dots, e_{tL}\}, \quad (3)$$

where e_u , e_i , and e_t denote the d -dimensional dense vectors for user, item, and tag, respectively. Note that these embeddings are trainable parameters and further utilized as initial inputs of GCN for information aggregation. In other words, they can be regarded as the output of 0th layer, namely $e_{u/i/t}^{(0)}$. In addition, the tag embeddings are shared between the

two tag-based graph, as shown in Figure 2. As a result, these tags can naturally establish potential links between users and items.

4.1.2. Graph Convolution

The core of graph convolution is to design appropriate aggregation function for integrating the features of neighbors effectively. Many researchers have proposed different aggregation methods, including the weighted sum aggregator [48], LSTM aggregator [49] and attention-based aggregator [50]. Moreover, most of the related studies have also introduced feature transformation or nonlinear activation into the aggregator. Although achieving good performance on node or graph classification with rich semantic input features, they may be cumbersome for the recommendation task, where each node is only represented by the ID embeddings.

Based on these considerations above, we adapt the similar graph convolution operators on tag-aware graphs used in LightGCN [26], where only the most essential component of GCN, namely neighbor aggregation, is retained. Suppose that $e_u^{(k)}$ and $e_{ut}^{(k)}$ are the embeddings of user u and tag t at the k th layer, respectively. The aggregation function on the user–tag graph can be defined as:

$$\begin{aligned} e_u^{(k+1)} &= \sum_{t \in N_u} \frac{1}{\sqrt{|N_u|} \sqrt{|N_{ut}|}} e_{ut}^{(k)}, \\ e_{ut}^{(k+1)} &= \sum_{u \in N_{ut}} \frac{1}{\sqrt{|N_u|} \sqrt{|N_{ut}|}} e_u^{(k)}, \end{aligned} \quad (4)$$

where N_u and N_{ut} represent the neighbor set of users and tags in the user–tag graph G_u , respectively. Following the standard GCN [51], we utilize the normalization term $\frac{1}{\sqrt{|N_u|} \sqrt{|N_{ut}|}}$ to prevent the increase scale of embeddings caused by the graph convolution operator. A similar aggregation operator is performed on the item–tag graph at the k th layer to generate item representation $e_i^{(k+1)}$ and tag representation $e_{it}^{(k+1)}$. It is worth noting that, except at the 0th layer, the obtained tag representations from two graphs at k th layer are obviously different. By applying GCN on the user–tag graph (item–tag graph), node embeddings can propagate back and forth between users (items) and tags. Consequently, the learned tag representations e_{ut} and e_{it} can describe the user interests and item characteristics, respectively.

After performing the k th graph convolution operator, the proposed model can produce k embedding vectors for each node, which are from different receptive fields and thus have different semantic information. To combine those embeddings of different layers, we leverage the simple weighted sum function to generate the final representation for each node, which is verified to be effective and workable [26]. Specifically, we define the readout function of combining outputs of all K layers as:

$$e_u = \frac{1}{K+1} \sum_{k=0}^K e_u^{(k)}, \quad e_i = \frac{1}{K+1} \sum_{k=0}^K e_i^{(k)}, \quad e_t = \frac{1}{K+1} \sum_{k=0}^K e_t^{(k)}, \quad (5)$$

where e_u , e_i , and e_t are the final representation of users, items, and tags, respectively.

4.1.3. Model Optimization

Based on the final representation of users and items, we take the inner product to predict the interaction probability between user u and item i :

$$\hat{y}_{u,i} = e_u^T e_i \quad (6)$$

where $\hat{y}_{u,i}$ is also considered as the ranking score for providing recommendations. In our tag-aware GCN, the model parameters to be trained are only the embeddings at the 0th

layer, including $e_u^{(0)}$ for all users, $e_i^{(0)}$ for all items, and $e_t^{(0)}$ for all tags. To optimize the model parameters, we adopt the log loss as the loss function, which is formulated as:

$$L_G = - \sum_{(u,i) \in R^+ \cup R^-} (y_{u,i} \log \hat{y}_{u,i} + (1 - y_{u,i}) \log(1 - \hat{y}_{u,i})) \quad (7)$$

where R^+ and R^- are the positive and negative instance set, respectively. $y_{u,i}$ is the ground truth of the user–item interaction, which is set to 1 if the pair $(u, i) \in R^+$. Otherwise, we set $y_{u,i} = 0$. By optimizing the loss function, positive item representations are learned to be close to user representations, while negative item representations are pushed farther.

4.2. Personalized Category-Boosted Negative Sampling

One of the core issues of training recommendation models is to obtain an effective and reasonable training data set, which contains a positive sample set and a negative sample set. However, in the recommendation task, we are only accessible to the positive sample from the implicit feedback, such as click and annotation [52]. Those items that have not been interacted with by a given user are presumed to be negative samples. In other words, the negative instance set can be composed of all unobserved interactions. It is unnecessary and rather time-consuming to use all negative samples for training. To tackle this problem, negative samples are mostly generated by randomly selecting from unobserved interactions in practice. This process is also known as negative sampling.

Most studies are devoted to modification of the negative sampling strategy, thus improving the recommendation accuracy [53,54]. However, only a few works attempted to leverage this strategy to promote the item diversity. For instance, a diversity-promoted negative sampler is proposed in the recent work [23], where those similar but negative samples are selected with a boosted higher probability. By twisting the distribution of negative samples, the trained recommendation model can discriminate the user preferences among similar items. In addition, a tunable hyper-parameter α is utilized to control the proportion of selecting similar negative items. It assumes that every user has the same tendency on item diversity, so the same value of α is applied in the negative sampler for all users. However, we argue that different users might have different tendencies on diversity. Some users are willing to explore new and unknown things, while others will conservatively choose their favorite fields.

Inspired by this, we design a personalized category-boosted negative sampling, which can adaptively determine the strength of sampling the similar negative items for each user. The procedure of the proposed personalized category-boosted negative sampling is presented in Algorithm 1. Specifically, we first compute the diversity score for each user by:

$$d_u = \frac{|C_u|}{|C_{all}|}, \quad (8)$$

where $|C_u|$ represents the number of categories covered by the interacted items of user u , and $|C_{all}|$ is the number of categories totally in the system. After performing normalization, we can obtain the diversity score table Du for all users. According to the diversity score, we then set the strength parameter for each user as:

$$\alpha_u = \alpha_0 \times d_u, \quad (9)$$

where α_0 is a shared parameter to control similar sampling for all users, which has the same function of α used in [23]. Finally, we choose the negative samples for each user from the same category or the whole item set according to the strength parameter α_u .

For those users that have higher diversity scores, more items from positive categories will be selected as negative samples, which enables the positive items from negative categories to be easier to be recommended than those from positive categories, thus providing more diverse recommendations. On the contrary, more negative samples are randomly

chosen from all categories for users with small diversity scores, which makes the recommendation model pay more attention on relevance.

Algorithm 1 Personalized category-boosted negative sampling

Input: positive sample set P ; item set I ; item category table C ; negative sampling size ns ; initial similar sampling weight α_0 ; user diversity score table Dv

Output: training sample set Ω

```

1:  $\Omega \leftarrow P$ 
2: for each positive sample  $(u, i, t)$  in  $P$  do
3:    $Ni \leftarrow I \setminus i$ 
4:    $Ci \leftarrow I_{C(i)} \setminus i$ 
5:    $\alpha_u \leftarrow \alpha_0 * Dv(u)$ 
6:   for  $j \leftarrow 1$  to  $ns$  do
7:      $r \leftarrow \text{rand}(0,1)$ 
8:     if  $r < \alpha_u$  then
9:        $i^- \leftarrow \text{randomly sample from } Ci$ 
10:    else
11:       $i^- \leftarrow \text{randomly sample from } Ni$ 
12:     $\Omega \leftarrow \Omega + (u, i^-, t)$ 
13: return  $\Omega$ 

```

4.3. Adversarial Learning

As introduced previously, we can learn effective user representations and item representations by optimizing the loss function. In an accuracy-focused recommendation model, the learned positive item representation should be closer to the target user representation than negative item representation. Essentially, item representations can reflect the interests of the corresponding users. Nevertheless, items have their own unique characteristics such as category. Especially in TRS, the user-defined tags contain rich semantic information. Similar items within the same category are more likely to have similar representations. Consequently, more items in the positive categories are easier to be recommended, which limits the opportunity for those items from diverse categories to be exposed to users.

Based on the above considerations, we adopt the strategy of adversarial learning used in [23,55] to further improve the recommendation diversity. Concretely, we perform an extra category classification task based on the learned item representations. The trained classifier is expected to predict the item category as accurately as possible. For the recommendation model, we hope that it can learn item representations which can deceive the classifier most while providing high-quality recommendations. In other words, the recommendation model and the classifier are trained in an adversarial way.

To construct an item category classifier, we utilize a fully connected layer with a softmax function. Formally, assuming that the final representations of item i is h_i , we can calculate the corresponding category probabilities by:

$$p_i = \text{softmax}(W^T h_i), \quad (10)$$

where W is the learnable parameter and the softmax function is used to transform the category scores into a probability distribution. Each element of p_i represents the probability of belonging to a certain class from the category set C and the sum of all elements is equal to 1. We adopt the multi-class cross entropy as the loss function, which is defined as

$$L_C(i) = - \sum_j^{|C|} c_i(j) \log p_i(j), \quad (11)$$

where $c_i(j)$ is 0 or 1, indicating whether j is the correct category of item i . By averaging the loss of all training samples, we can obtain the whole loss of category classification L_C .

Within the framework of adversarial learning, the goal of the classifier is to minimize the classification loss L_C , while the goal of the proposed recommendation model is to minimize $L_G - \lambda L_C$, where L_G is the loss of the recommendation task, and λ is used to trade off the main task and the extra adversarial task. Note that the sign “-” before L_C means that the classification loss is reversed to force representation of items in the same category away from each other. To train the model based on adversarial learning in an elegant way, we resort to the gradient reversal layer (GRL), which is verified to be effective and efficient in [56]. Mathematically, GRL can be regarded as a “pseudo-function” $GR_\lambda(x)$ that has the following forward- and back-propagation behavior:

$$\begin{aligned} GR_\lambda(x) &= x, \\ \frac{dGR_\lambda(x)}{dx} &= -\lambda \mathbf{I}, \end{aligned} \quad (12)$$

where \mathbf{I} denotes an identity matrix and λ is the tunable parameter. Fundamentally, GRL serves as an identity transformation at the stage of the forward propagation. More importantly, GRL can obtain the gradient from the next layer, multiply it by $-\lambda$, and pass it to the previous layer at the stage of back-propagation.

By inserting a GRL in between the learned item representation and the category classifier, our adversarial learning can be achieved in a deep feed-forward architecture through standard back-propagation training. With the aid of adversarial learning, the proposed model can learn representations of items that retain user interests to a large extent and make these representations category-free. Accordingly, neighbors of items in the embedding space will cover more diverse categories, thus conducing to provide diverse recommendations.

4.4. Two-Way TransTag Regularization

Generally, users have diverse interests and items are of diverse characteristics. The user’s preference for a target item occurs when some features of the item exactly match the user’s interest in some aspect. In social tagging systems, these user–item interactions are hidden in the tagging behavior data. Nevertheless, the diverse semantic information cannot be well modeled by GCN. For this purpose, we design a two-way TransTag regularization function to capture diverse user interests or item characteristics.

The TransTag regularization function is inspired by TransE [57], a common method for knowledge graph embedding, which models relationships as translations in the embedding space. The rationale behind the TransTag regularization function is that the user–tag–item triplet (u, t, i) indicates that user u prefers item i from the aspect of tag t . Hence, the item embedding e_i should be a nearest neighbor of $e_u + e_t$ in the underlying embedding space, i.e., $e_u + e_t \approx e_i$. It is worth mentioning that the one-way TransTag regularization function has been successfully applied in recent tag-aware recommendations [14,26].

Note that a user prefers items based on diverse interests represented by tags, and an item with diverse characteristics is also oriented towards appropriate users. It will result in insufficient expression ability only considering one-way translation $u \rightarrow i$ or $i \rightarrow u$ [24]. Therefore, we conduct a two-way TransTag regularization function. Given a user–tag–item triplet (u, t, i) , the distance function in our model is defined as:

$$\begin{aligned} d_{u \rightarrow i}(u, t, i) &= \|e_u + e_{ut} - e_i\|, \\ d_{i \rightarrow u}(i, t, u) &= \|e_i + e_{it} - e_u\|, \end{aligned} \quad (13)$$

where e_u and e_{ut} are the respective final representations of user u and tag t learned from the user–tag graph, while e_i and e_{it} are the respective final item and tag representations learned from the item–tag graph.

Based on the distance function, we adopt a margin based pairwise ranking loss defined as:

$$L_E = \sum_{(u,t,i) \in S^+, (u',t,i') \in S^-} \text{ReLU}(d_{u \rightarrow i}(u, t, i) - d_{u \rightarrow i}(u', t, i') + m) + \text{ReLU}(d_{i \rightarrow u}(i, t, u) - d_{i \rightarrow u}(i', t, u') + m), \quad (14)$$

where m is the margin parameter, $(u, t, i) \in S^+$ denotes the positive sample, and (u', t, i') indicates the negative sample that is generated by randomly replacing either the user or item node in (u, t, i) . By minimizing the loss function L_E , we bridge the information gap between users and items based on tags.

4.5. Joint Optimization

To provide personalized recommendations considering both accuracy and diversity, we augment a GCN-based tag-aware recommendation model with an additional adversarial task of item category classification and a two-way TransTag regularization. We jointly optimize the proposed model in an end-to-end manner by minimizing the integrative loss:

$$L = L_G - \lambda_1 L_C + \lambda_2 L_E + \lambda_3 \|\Theta\|_2^2, \quad (15)$$

where λ_1 , λ_2 , and λ_3 indicate the strengths of the two augmented objectives and the L_2 regularization term, respectively.

5. Experiments

To demonstrate the effectiveness of the proposed DTGCF, we perform extensive experiments and exhibit detailed results in this section.

5.1. Experimental Setup

5.1.1. Data Description and Evaluation Indicators

To evaluate the performance of different models on tag-aware recommendations, we utilize a common benchmark dataset: Movielens (<https://grouplens.org/datasets/movielens/>, accessed on 1 August 2022), where only those users who have both rating and tag information can retain. More importantly, the dataset also provides additional item category information. Following a similar preprocessing strategy [3,13], we eliminate the infrequent tags that are used less than $5 \times$ in the Movielens 10M dataset and less than $10 \times$ in the Movielens 20M dataset. The statistics of the experimental datasets are summarized in Table 1. In addition, we split the data to generate training, validation, and testing set according to a ratio of 7:1:2.

Table 1. Statistical information of experimental datasets.

Dataset	#Users	#Items	#Tags	#Categories	#Interactions
Movielens 10M	1651	5381	1586	19	36,728
Movielens 20M	6961	17,441	5006	19	394,745

In order to comprehensively evaluate the performance of tag-aware recommendations from multiple perspectives, we adopt the following different types of evaluation indicators.

- **Accuracy.** We take two commonly used metrics to estimate the recommendation accuracy: Hit Ratio (HR) and Recall [58]. HR considers whether the user interacted with its recommended items at least one time, while Recall calculates the ratio between the number of correctly predicted items and the size of the user's actual interaction item set. A higher value of HR or Recall indicates more accurate recommendations;
- **Diversity.** We take the intra-list distance (ILD) and category coverage (CC) [59] to estimate the recommendation diversity. ILD calculates the average distance between

all pairs of items in the top- N recommendation list, while CC measures the diversity by computing the number of recommended categories. A higher value of ILD or CC means more diverse recommendations;

- Trade-off. We take the F-score (F1) indicator [24] to measure the trade-off between accuracy and diversity. Formally, F-score can be defined as:

$$F1 = \frac{2 \times \text{accuracy} \times \text{diversity}}{\text{accuracy} + \text{diversity}}, \quad (16)$$

where accuracy can be HR or Recall and diversity can be ILD or CC. In our experiments, we use HR and CC to calculate F-score.

5.1.2. Baselines

We compare the proposed DTGCF with the following baselines:

- CCF [5]: CCF is a clustering-based CF model, which adopts hierarchical clustering to aggregate redundant tags. User and item representations based on these tag clusters are generated and used to provide recommendations;
- ACF [3]: ACF leverages the deep autoencoders to extract tag-based low-dimensional dense features for users and employs user-based collaborative filtering to generate recommendations according to those extracted features;
- DSPR [9]: It is a tag-aware deep recommendation model which adopts deep neural networks to learn nonlinear latent features for users and items from the tag space;
- TGCN [13]: It constructs a tag graph convolutional network, which employs a type-aware neighbor sampling strategy and an attentive CNN-based information updating method for improved tag-aware recommendations;
- LFGCN [14]: It establishes a tag-aware light graph convolutional network to integrate topological structure for learning more accurate representation of users and items;
- MMR [20]: MMR is a diversity-based re-ranking method by maximizing the marginal relevance. This method is initially used to reorder documents and generate summaries and is further extended into item recommendation [21];
- DPP [22]: DPP is an elegant probabilistic model of repulsion, in which higher probability is assigned to those item sets that are more diverse. In this way, it can balance the recommendation accuracy and diversity, providing re-ranked recommendation lists;
- DGCN [23]: It is a GCN-based diversified recommendation model which performs three diversity-based strategies: rebalanced neighbor discovering, category boosted negative sampling, and adversarial learning.

To summarize, these baseline methods can be divided into two types. The first type (from CCF to LFGCN) belongs to the accuracy-focused tag-aware recommendation model. Among them, CCF, ACF, and DSPR are feature-based tag-aware models, while TGCN and LFGCN are GCN-based models. The second type (from MMR to DGCN) is the diversity-boosted recommendation model. Note that MMR and DPP are post-processing methods, while DGCN is an end-to-end model for diversified recommendation.

5.1.3. Parameter Settings

For the baseline methods, we directly use the default hyper-parameters if they are reported in the corresponding previous work, and we conduct grid search to carefully tune the other parameters which are not presented explicitly. For those models based on deep networks, we adopt the Xavier method [60] to initialize the model parameters and the min-batch Adam optimizer [61] to optimize the model. The batch size is fixed to 1024, and the learning rate is tuned in the range of $\{0.0001, 0.0005, 0.001, 0.005\}$. We fix the size of embeddings (users, items, and tags) to be 64 and set the number of selecting negative sample as 4 for a fair comparison. The hyper-parameter λ to control the strength of L_2 regularization is searched from $\{1 \times 10^{-6}, 1 \times 10^{-5}, \dots, 1 \times 10^{-2}\}$. Each model is trained until convergence, and the early stopping approach based on validation data is also

conducted to prevent overfitting. For our model, we set the margin in the pairwise ranking loss $m = 1$, and the number of layers $K = 3$. We tune the initial similar sampling weight α_0 in $\{0.02, 0.04, 0.06, \dots, 0.30\}$ with an increment of 0.02, the hyper-parameters λ_1 , and λ_2 in $\{1 \times 10^{-5}, 1 \times 10^{-3}, \dots, 1 \times 10^{-1}\}$. The proposed model is implemented with Pytorch based on an Nvidia GeForce RTX 3080 Ti GPU with 12 GB memory.

5.2. Performance Comparison

Table 2 presents the performance comparison of the proposed DTGCF and other compared methods on two datasets in terms of accuracy, diversity, and trade-off metrics. From the results, we can have several findings:

- (1) For the tag-aware recommendation models, we can observe that CCF and ACF consistently perform worse than other models based on deep learning, such as DSPR, TGCN, and LFGCF, revealing the superiority of deep neural networks on representation learning adequately. Although introducing the deep autoencodes, ACF do not construct the direct correlation between the deep learning model and the recommendation performance, which limits the model expression ability. Hence, the resulting user representation may not be accurate enough for providing high-quality recommendations;
- (2) GCN-based tag-aware models, including TGCN, LFGCF, and DTGCF, outperform other models for a large margin in terms of accuracy, verifying the importance of integrating rich semantic information from neighbors. In addition, the results of LFCGC and DTGCF are significantly better than that of TGCN, which demonstrates the effectiveness and robustness of the simplified architecture;
- (3) In terms of diversity, the results of MMR and DPP are obviously better than those of accuracy-focused models such as CCF, ACF, TGCN, and LFGCF. It can be found that the recommendation accuracy significantly degrades so as to keep a good diversity. In particular, DGCN performs slightly worse than TGCN in terms of accuracy but achieves significantly better diversity, indicating the rationality of the end-to-end diversified recommendation model;
- (4) The proposed DTGCF achieves better comprehensive performance than all the base-lines according to the trade-off metric (F1). The relative improvement of DTGCF over the best competitors on two datasets are 7.40%, and 5.98% on F1@10, and 5.62%, and 5.53% on F1@20, respectively. In terms of accuracy, the proposed model ranks first in most cases, except for Recall@20 on the 10M dataset, and Recall@10 on the 20M dataset. In addition, DTGCF also outperforms diversity-promoted methods in most cases in terms of diversity. Although performing slightly worse than DPP on 20M dataset in ILD@20, our model has substantially better performance in terms of accuracy. Compared to DGCN, DTGCF generates better results in both accuracy and diversity on two datasets, which demonstrates the effectiveness of our diversified tag-aware model.

To summarize, the proposed DTCCF can gain superior or competitive performance in the light of the recommendation accuracy while greatly promoting the diversity. We attribute the advantages of our model to the following three aspects. Firstly, a simplified GCN customized for tag-aware recommendations is conducted to ensure a considerable level of recommendation accuracy. Secondly, we leverage the personalized category-boosted negative sampling and adversarial learning to substantially improve the recommendation diversity. Thirdly, the two-way TransTag regularization is performed to properly model the three-dimensional relation among users, tags, and items, which can further enhance the ability of feature expression and generate more satisfactory recommendations.

Table 2. Overall performance comparison on two datasets in terms of accuracy, diversity, and trade-off metrics. The best result is in bold, and the second best one is marked with an underline. Statistical tests are conducted between the best result and the second best result. If significant differences are detected (p -value < 0.05), the corresponding best result is marked with a symbol “*”.

Datasets	Models	Accuracy				Diversity				Trade-Off	
		Recall@10	Recall@20	HR@10	HR@20	CC@10	CC@20	ILD@10	ILD@20	F1@10	F1@20
10M	CCF	0.0451	0.0753	0.1423	0.2040	0.5124	0.6524	0.7452	0.7884	0.2227	0.3108
	ACF	0.0485	0.0784	0.1476	0.2074	0.5048	0.6432	0.7243	0.7543	0.2284	0.3137
	DSPR	0.0612	0.1025	0.1824	0.2216	0.5160	0.6445	0.7338	0.7856	0.2695	0.3298
	TGCN	0.0824	0.1246	0.2520	0.3152	0.5138	0.6580	0.7382	0.7821	0.3381	0.4262
	LFGCF	0.1050	0.1484	0.3035	0.3526	0.5027	0.6440	0.7336	0.7885	0.3785	0.4557
	MMR	0.0413	0.0645	0.1395	0.1875	0.5538	0.7064	0.7960	0.8108	0.2229	0.2963
	DPP	0.0438	0.0723	0.1412	0.1954	0.5672	0.6928	0.8072	0.8262	0.2261	0.3048
	DGCN	0.0845	0.1215	0.2476	0.3025	0.5524	0.7156	0.7852	0.8214	0.3419	0.4252
	DTGCF	0.1086	<u>0.1452</u>	0.3124 *	0.3580	0.5821 *	0.7340 *	0.8124 *	0.8350 *	0.4065 *	0.4813 *
20M	CCF	0.0504	0.0768	0.1550	0.2114	0.5161	0.6258	0.7161	0.7525	0.2384	0.3160
	ACF	0.0523	0.0812	0.1632	0.2230	0.5026	0.6114	0.7120	0.7552	0.2464	0.3268
	DSPR	0.0689	0.1138	0.2035	0.2458	0.5214	0.6136	0.7212	0.7584	0.2927	0.3510
	TGCN	0.0933	0.1274	0.2664	0.3086	0.5126	0.6218	0.7268	0.7636	0.3506	0.4125
	LFGCF	0.1174	0.1362	0.3156	0.3552	0.5082	0.6134	0.7142	0.7650	0.3894	0.4499
	MMR	0.0431	0.0685	0.1468	0.1940	0.5528	0.6624	0.7848	0.8134	0.2320	0.3001
	DPP	0.0468	0.0786	0.1528	0.2016	0.5631	0.6748	0.8134*	0.8065	0.2404	0.3105
	DGCN	0.0912	0.1256	0.2610	0.2912	0.5648	0.6673	0.7768	0.7972	0.3570	0.4055
	DTGCF	<u>0.1136</u>	0.1421 *	0.3182	0.3624 *	0.5872 *	0.6882 *	<u>0.8072</u>	0.8264 *	0.4127 *	0.4748 *

5.3. Ablation Study

In this section, we perform ablation studies to investigate the effect of the designed components in DTGCF, including personalized category-boosted negative sampling, adversarial learning, and two-way TransTag regularization. Specifically, we compare our model with the basic tag-aware GCN without any additional component and the tag-aware GCN with only one component. In addition, we also compare the performance with DPP and DGCN. Table 3 lists the comparison results on the Movielens 10M dataset, from which we have the following observations:

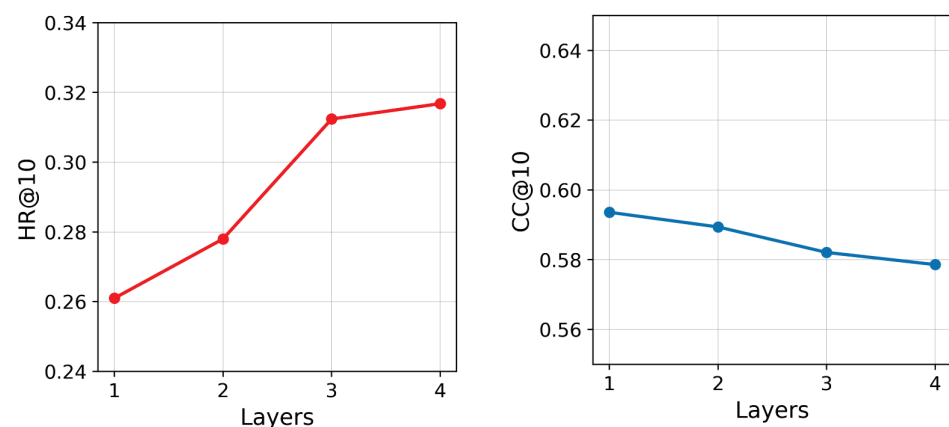
- (1) In terms of accuracy, our basic GCN without any additional component performs substantially better than DPP and DGCN, which demonstrates the effectiveness of the proposed GCN architecture for tag-aware recommendations. Nevertheless, the diversity of the basic GCN is not good compared to DPP and DGCN, resulting in pool user satisfaction;
- (2) Each component plays an obvious positive role in improving diversity or accuracy. After introducing personalized category-boosted negative sampling into the tag-aware GCN, the recommendation diversity is significantly promoted while keeping comparable accuracy performance. The utilization of adversarial learning can achieve an impressive increase in diversity, but make a large sacrifice on accuracy, which confirms the existing accuracy-diversity dilemma. In particular, the basic GCN with adversarial learning has competitive diversity performance compared to DPP and DGCN. By adding the two-way TransTag regularization function, the model achieves obvious improvement on accuracy and slight enhancement on diversity;
- (3) In terms of the trade-off metrics, the proposed DTGCF is consistently superior to all the variant methods, which indicates that, by integrating three components of different characteristics within a joint learning framework, the proposed DTGCF can provide more satisfactory recommendations with respect to both accuracy and diversity.

Table 3. Ablation study on the Movielens 10M dataset.

Method	Recall@10	HR@10	CC@10	ILD@10	F1@10
DPP	0.0438	0.1412	0.5672	0.8072	0.2261
DGCN	0.0912	0.2476	0.5524	0.7768	0.3419
tag-aware GCN	0.0924	0.2882	0.5136	0.7330	0.3692
personalized negative sampling	0.0918	0.2534	0.5424	0.7694	0.3454
adversarial learning	0.0813	0.2403	0.5854	0.8164	0.3407
two-way TransTag	0.1124	0.3156	0.5187	0.7485	0.3924
DTGCF	0.1086	0.3124	0.5821	0.8124	0.4065

5.4. Parameter Analysis

We first investigate the impact of the layer number K , which is a key hyper-parameter in GCN-based recommendation models. A larger value of K indicates that the model will leverage more embedding propagation layers stacked to generate node representations. We vary the value of K in the range of $\{1, 2, 3, 4\}$ and present the corresponding results with respect to accuracy and diversity in Figure 3. The metrics of HR@10 and CC@10 are used to evaluate the recommendation accuracy and diversity. Note that using only one layer though the proposed DTGCF surpasses ACF and DSPR in both accuracy and diversity, which verifies the benefit of aggregating rich information from neighbors. It can be observed that, as the layer number increases, the recommendation accuracy of the model is obviously improved, but the significant advancement diminishes when the layer number is larger than 3. However, with the increase of the number of layers, the diversity of the proposed model decreases relatively slowly compared to the speed of accuracy change. It justifies the rationality and effectiveness of our diversity-promoted strategies.

**Figure 3.** Results of different number layers on the Movielens 10M dataset in terms of HR@10 and CC@10.

Then, we conduct experiments to examine the impact of the hyper-parameter α_0 , which is used to control the probability of selecting similar but negative samples. Specifically, we search the value of α_0 in $\{0.02, 0.04, 0.06, \dots, 0.30\}$ with an increment of 0.02. The experimental results are shown in Figure 4. We can see that the diversity of model is continuously improved with the increase of α_0 . Since more negative items are chosen from positive categories, the trained model will learn users' interest across categories and recommend items from more diverse categories. In particular, with a relatively small α_0 , the recommendation accuracy is quite stable and even gains a slightly improvement. However, the accuracy declines sharply when α_0 becomes too large to capture reasonable representations. Therefore, a smaller value of α_0 is recommended to balance the recommendation accuracy and diversity.

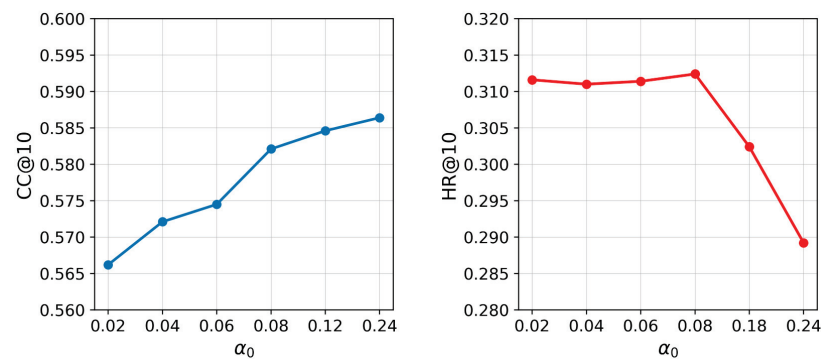


Figure 4. Results of different values of α_0 on the Movielens 10M dataset in terms of HR@10 and CC@10.

5.5. Further Discussion

In the proposed model, we adopt an improved category-boosted negative sampling, which adaptively determines the probability of selecting similar but negative samples for each user according to its historic diversity. To verify the adaptive ability, we compare the diversity distribution of the final recommendations of LFGCF, DPP, DGCN, and our DTGCF. To be specific, we compute the diversity score of recommended top- N items for each user, where N is set as 10. The detailed results are exhibited in Figure 5.

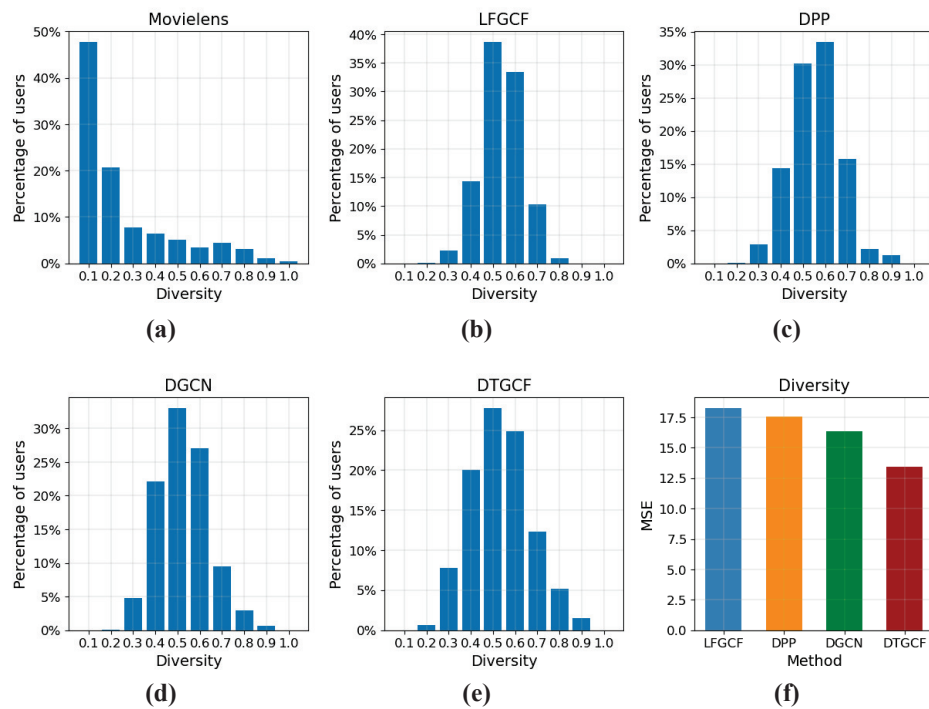


Figure 5. The results of user diversity distribution. (a) is the real diversity distribution of users on Movielens 10M; (b–e) are the predicted diversity distribution of LFGCF, DPP, DGCN, and DTGCF; (f) presents the MSE results of different methods.

Since interaction behaviors with both rating and tagging are rather sparse, the real diversity distribution of users is seriously skewed, as shown in Figure 5a. Although DPP and DGCN achieve remarkably better performance than LFGCF on diversity, they make a great sacrifice on accuracy to gain enough diversity. Note that DPP and DGCN utilize a fixed hyper-parameter to control the diversity of all users, so the distribution of predicted items obviously cannot match the actual situation. Unlike them, the proposed DTGCF takes the historical diversity of different users into account, which will promote the recommendation diversity while ensuring comparable accuracy. Moreover, we calculate the mean square error (MSE) of different models to measure the differences between the predicted diversity and the ground-truth for each user. As presented in Figure 5f, the MSE result of our model

is much better than other competitors, demonstrating the effectiveness of our diversity-promoted strategy adequately.

6. Conclusions

In this work, we construct an end-to-end GCN-based framework for diversified tag-aware recommendation. Benefiting from its great power in representation learning, the tailored tag-aware GCN can capture more accurate and reasonable representation of users, items, and tags. In particularly, we augment the GCN-based model with three modules. Among them, personalized category-boosted negative sampling and adversarial learning are designed for promoting diversity, while the two-way TransTag regularization function is adopted for accuracy. Experimental results show that the proposed model outperforms other compared algorithms on diversified tag-aware recommendation. Further ablation studies also validate the effectiveness and rationality of the augmented modules.

Future work will continue to improve the proposed model in the following aspects. Firstly, other side information, such as reviews and social relations, can be added into our framework to enrich semantic information of nodes, thus boosting the recommendation performance. Secondly, we focus on improving individual diversity in this work. Aggregate diversity of recommendations across all users, also known as long-tail recommendation, is also worth studying. Therefore, other diversity-promoting strategies should be carefully designed to be integrated into our model.

Author Contributions: The overall study supervised by S.L.; Methodology, hardware, software, and preparing the original draft by Y.Z. (Yi Zuo); Review and editing by Y.Z. (Yun Zhou). All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by the National Natural Science Foundation of China (Grant No. 61902117 and Grant No. 72073041), the National Natural Science Foundation of Hunan Province (Grant No. 2020JJ5010), and the Scientific Research Project of the Hunan Provincial Department of Education (Grant No. 22A0667).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author, upon reasonable request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Zhang, Z.K.; Zhou, T.; Zhang, Y.C. Tag-aware recommender systems: A state-of-the-art survey. *J. Comput. Sci. Technol.* **2011**, *26*, 767–777. [\[CrossRef\]](#)
2. Tso-Sutter, K.H.; Marinho, L.B.; Schmidt-Thieme, L. Tag-aware recommender systems by fusion of collaborative filtering algorithms. In Proceedings of the 2008 ACM Symposium on Applied Computing, Fortaleza, Brazil, 16–20 March 2008; pp. 1995–1999.
3. Zuo, Y.; Zeng, J.; Gong, M.; Jiao, L. Tag-aware recommender systems based on deep neural networks. *Neurocomputing* **2016**, *204*, 51–60. [\[CrossRef\]](#)
4. Symeonidis, P.; Nanopoulos, A.; Manolopoulos, Y. Tag recommendations based on tensor dimensionality reduction. In Proceedings of the 2008 ACM Conference on Recommender Systems, Lausanne, Switzerland, 23–25 October 2008; pp. 43–50.
5. Shepitsen, A.; Gemmell, J.; Mobasher, B.; Burke, R. Personalized recommendation in social tagging systems using hierarchical clustering. In Proceedings of the 2008 ACM Conference on Recommender Systems, Lausanne, Switzerland, 23–25 October 2008; pp. 259–266.
6. Zhang, Z.K.; Zhou, T.; Zhang, Y.C. Personalized recommendation via integrated diffusion on user–item–tag tripartite graphs. *Phys. A Stat. Mech. Its Appl.* **2010**, *389*, 179–186. [\[CrossRef\]](#)
7. Le, N.Q.K.; Do, D.T.; Hung, T.N.K.; Lam, L.H.T.; Huynh, T.T.; Nguyen, N.T.K. A computational framework based on ensemble deep neural networks for essential genes identification. *Int. J. Mol. Sci.* **2020**, *21*, 9070. [\[CrossRef\]](#) [\[PubMed\]](#)
8. Kha, Q.H.; Ho, Q.T.; Le, N.Q.K. Identifying SNARE Proteins Using an Alignment-Free Method Based on Multiscan Convolutional Neural Network and PSSM Profiles. *J. Chem. Inf. Model.* **2022**, *62*, 4820–4826. [\[CrossRef\]](#)

9. Xu, Z.; Chen, C.; Lukaszewicz, T.; Miao, Y.; Meng, X. Tag-aware personalized recommendation using a deep-semantic similarity model with negative sampling. In Proceedings of the 25th ACM International on Conference on Information and Knowledge Management, Indianapolis, IN, USA, 24–28 October 2016; pp. 1921–1924.
10. Liang, N.; Zheng, H.T.; Chen, J.Y.; Sangaiah, A.K.; Zhao, C.Z. TrsdI: Tag-aware recommender system based on deep learning-intelligent computing systems. *Appl. Sci.* **2018**, *8*, 799. [[CrossRef](#)]
11. Huang, R.; Wang, N.; Han, C.; Yu, F.; Cui, L. TNAM: A tag-aware neural attention model for Top-N recommendation. *Neurocomputing* **2020**, *385*, 1–12. [[CrossRef](#)]
12. Chen, B.; Ding, Y.; Xin, X.; Li, Y.; Wang, Y.; Wang, D. AIRec: Attentive intersection model for tag-aware recommendation. *Neurocomputing* **2021**, *421*, 105–114. [[CrossRef](#)]
13. Chen, B.; Guo, W.; Tang, R.; Xin, X.; Ding, Y.; He, X.; Wang, D. TGCN: Tag graph convolutional network for tag-aware recommendation. In Proceedings of the 29th ACM International Conference on Information & Knowledge Management, Virtual, 19–23 October 2020; pp. 155–164.
14. Zhang, Y.; Xu, C.; Wu, X.; Zhang, Y.; Dong, L.; Wang, W. LFGCF: Light Folksonomy Graph Collaborative Filtering for Tag-Aware Recommendation. *arXiv* **2022**, arXiv:2208.03454.
15. Cheng, P.; Wang, S.; Ma, J.; Sun, J.; Xiong, H. Learning to recommend accurate and diverse items. In Proceedings of the 26th International Conference on World Wide Web, Perth, Australia, 3–7 April 2017; pp. 183–192.
16. Ge, Y.; Zhao, S.; Zhou, H.; Pei, C.; Sun, F.; Ou, W.; Zhang, Y. Understanding echo chambers in e-commerce recommender systems. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual, 25–30 July 2020; pp. 2261–2270.
17. Sun, J.; Guo, W.; Zhang, D.; Zhang, Y.; Regol, F.; Hu, Y.; Guo, H.; Tang, R.; Yuan, H.; He, X.; et al. A framework for recommending accurate and diverse items using bayesian graph convolutional neural networks. In Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Virtual, 6–10 July 2020; pp. 2030–2039.
18. Huang, Y.; Wang, W.; Zhang, L.; Xu, R. Sliding Spectrum Decomposition for Diversified Recommendation. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, Virtual, 14–18 August 2021; pp. 3041–3049.
19. Berbague, C.E.; Karabadj, N.E.i.; Seridi, H.; Symeonidis, P.; Manolopoulos, Y.; Dhifli, W. An overlapping clustering approach for precision, diversity and novelty-aware recommendations. *Expert Syst. Appl.* **2021**, *177*, 114917. [[CrossRef](#)]
20. Carbonell, J.; Goldstein, J. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Melbourne, Australia, 24–28 August 1998; pp. 335–336.
21. Luan, W.; Liu, G.; Jiang, C.; Zhou, M. MPTR: A maximal-marginal-relevance-based personalized trip recommendation method. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 3461–3474. [[CrossRef](#)]
22. Chen, L.; Zhang, G.; Zhou, E. Fast greedy map inference for determinantal point process to improve recommendation diversity. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 1–12.
23. Zheng, Y.; Gao, C.; Chen, L.; Jin, D.; Li, Y. DGCN: Diversified Recommendation with Graph Convolutional Networks. In Proceedings of the Web Conference 2021, Ljubljana, Slovenia, 19–23 April 2021; pp. 401–412.
24. Liang, Y.; Qian, T.; Li, Q.; Yin, H. Enhancing domain-level and user-level adaptivity in diversified recommendation. In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual, 11–15 July 2021; pp. 747–756.
25. Yang, L.; Wang, S.; Tao, Y.; Sun, J.; Liu, X.; Yu, P.S.; Wang, T. DGRec: Graph Neural Network for Recommendation with Diversified Embedding Generation. *arXiv* **2022**, arXiv:2211.10486.
26. He, X.; Deng, K.; Wang, X.; Li, Y.; Zhang, Y.; Wang, M. Lightgcn: Simplifying and powering graph convolution network for recommendation. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual, 25–30 July 2020; pp. 639–648.
27. Schafer, J.B.; Frankowski, D.; Herlocker, J.; Sen, S. Collaborative filtering recommender systems. In *The Adaptive Web: Methods and Strategies of Web Personalization*; Springer: Berlin/Heidelberg, Germany, 2007; pp. 291–324.
28. Aggarwal, C.C.; Aggarwal, C.C. Content-based recommender systems. In *Recommender Systems: The Textbook*; Springer: Berlin/Heidelberg, Germany, 2016; pp. 139–166.
29. Çano, E.; Morisio, M. Hybrid recommender systems: A systematic literature review. *Intell. Data Anal.* **2017**, *21*, 1487–1524. [[CrossRef](#)]
30. Jannach, D.; Manzoor, A.; Cai, W.; Chen, L. A survey on conversational recommender systems. *ACM Comput. Surv.* **2021**, *54*, 105. [[CrossRef](#)]
31. Guo, Q.; Zhuang, F.; Qin, C.; Zhu, H.; Xie, X.; Xiong, H.; He, Q. A survey on knowledge graph-based recommender systems. *IEEE Trans. Knowl. Data Eng.* **2020**, *34*, 3549–3568. [[CrossRef](#)]
32. Wang, S.; Cao, L.; Wang, Y.; Sheng, Q.Z.; Orgun, M.A.; Lian, D. A survey on session-based recommender systems. *ACM Comput. Surv.* **2021**, *54*, 154. [[CrossRef](#)]
33. Marinho, L.B.; Schmidt-Thieme, L. Collaborative tag recommendations. In *Data Analysis, Machine Learning and Applications*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 533–540.
34. Zhao, S.; Du, N.; Nauerz, A.; Zhang, X.; Yuan, Q.; Fu, R. Improved recommendation based on collaborative tagging behaviors. In Proceedings of the 13th International Conference on Intelligent User Interfaces, Gran Canaria, Spain, 13–16 January 2008; pp. 413–416.

35. Li, H.; Diao, X.; Cao, J.; Zhang, L.; Feng, Q. Tag-aware recommendation based on Bayesian personalized ranking and feature mapping. *Intell. Data Anal.* **2019**, *23*, 641–659. [[CrossRef](#)]
36. Pan, Y.; Huo, Y.; Tang, J.; Zeng, Y.; Chen, B. Exploiting relational tag expansion for dynamic user profile in a tag-aware ranking recommender system. *Inf. Sci.* **2021**, *545*, 448–464. [[CrossRef](#)]
37. Berg, R.v.d.; Kipf, T.N.; Welling, M. Graph convolutional matrix completion. *arXiv* **2017**, arXiv:1706.02263.
38. Ying, R.; He, R.; Chen, K.; Eksombatchai, P.; Hamilton, W.L.; Leskovec, J. Graph convolutional neural networks for web-scale recommender systems. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, London, UK, 19–23 August 2018; pp. 974–983.
39. Wang, X.; He, X.; Wang, M.; Feng, F.; Chua, T.S. Neural graph collaborative filtering. In Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, Paris, France, 21–25 July 2019; pp. 165–174.
40. Ziegler, C.N.; McNee, S.M.; Konstan, J.A.; Lausen, G. Improving recommendation lists through topic diversification. In Proceedings of the 14th International Conference on World Wide Web, Chiba, Japan, 10–14 May 2005; pp. 22–32.
41. Qin, L.; Zhu, X. Promoting diversity in recommendation by entropy regularizer. In Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, Beijing, China, 3–9 August 2013.
42. Zhou, T.; Kuscsik, Z.; Liu, J.G.; Medo, M.; Wakeling, J.R.; Zhang, Y.C. Solving the apparent diversity-accuracy dilemma of recommender systems. *Proc. Natl. Acad. Sci. USA* **2010**, *107*, 4511–4515. [[CrossRef](#)] [[PubMed](#)]
43. Ashkan, A.; Kveton, B.; Berkovsky, S.; Wen, Z. Optimal greedy diversity for recommendation. In Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, Buenos Aires, Argentina, 25–31 July 2015.
44. Sha, C.; Wu, X.; Niu, J. A framework for recommending relevant and diverse items. *Proc. IJCAI* **2016**, *16*, 3868–3874.
45. Gillenwater, J.; Kulesza, A.; Mariet, Z.; Vassilytskii, S. A tree-based method for fast repeated sampling of determinantal point processes. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; pp. 2260–2268.
46. Warlop, R.; Mary, J.; Gartrell, M. Tensorized determinantal point processes for recommendation. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA, 4–8 August 2019; pp. 1605–1615.
47. He, X.; Liao, L.; Zhang, H.; Nie, L.; Hu, X.; Chua, T.S. Neural collaborative filtering. In Proceedings of the 26th International Conference on World Wide Web, Perth, Australia, 3–7 April 2017; pp. 173–182.
48. Xu, K.; Hu, W.; Leskovec, J.; Jegelka, S. How powerful are graph neural networks? *arXiv* **2018**, arXiv:1810.00826.
49. Hamilton, W.; Ying, Z.; Leskovec, J. Inductive representation learning on large graphs. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1–11.
50. Velickovic, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph attention networks. *Stat* **2017**, *1050*, 20.
51. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv* **2016**, arXiv:1609.02907.
52. Rendle, S.; Freudenthaler, C.; Gantner, Z.; Schmidt-Thieme, L. BPR: Bayesian personalized ranking from implicit feedback. *arXiv* **2012**, arXiv:1205.2618.
53. Ding, J.; Feng, F.; He, X.; Yu, G.; Li, Y.; Jin, D. An improved sampler for bayesian personalized ranking by leveraging view data. In Proceedings of the Web Conference 2018, Lyon, France, 23–27 April 2018; pp. 13–14.
54. Ding, J.; Quan, Y.; He, X.; Li, Y.; Jin, D. Reinforced Negative Sampling for Recommendation with Exposure Data. *Proc. IJCAI* **2019**, *2019*, 2230–2236.
55. Maddalena, L.; Giordano, M.; Manzo, M.; Guarracino, M.R. Whole-graph embedding and adversarial attacks for life sciences. In *Trends in Biomathematics: Stability and Oscillations in Environmental, Social, and Biological Models*; Springer: Berlin/Heidelberg, Germany, 2022; pp. 1–21.
56. Ganin, Y.; Lempitsky, V. Unsupervised domain adaptation by backpropagation. In Proceedings of the International Conference on Machine Learning, Lille, France, 6–11 July 2015; pp. 1180–1189.
57. Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; Yakhnenko, O. Translating embeddings for modeling multi-relational data. *Adv. Neural Inf. Process. Syst.* **2013**, *26*, 1–9.
58. Fayyaz, Z.; Ebrahimian, M.; Nawara, D.; Ibrahim, A.; Kashef, R. Recommendation systems: Algorithms, challenges, metrics, and business opportunities. *Appl. Sci.* **2020**, *10*, 7748. [[CrossRef](#)]
59. Han, J.; Yamana, H. A survey on recommendation methods beyond accuracy. *IEICE Trans. Inf. Syst.* **2017**, *100*, 2931–2944. [[CrossRef](#)]
60. Glorot, X.; Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, Sardinia, Italy, 13–15 May 2010; pp. 249–256.
61. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.