Multi-auxiliary Augmented Collaborative Variational Auto-encoder for Tag Recommendation

JING YI and XUBIN REN, Wuhan University, China
ZHENZHONG CHEN, Wuhan University, China and Hubei Luojia Laboratory, China

Recommending appropriate tags to items can facilitate content organization, retrieval, consumption, and other applications, where hybrid tag recommender systems have been utilized to integrate collaborative information and content information for better recommendations. In this article, we propose a multi-auxiliary augmented collaborative variational auto-encoder (MA-CVAE) for tag recommendation, which couples item collaborative information and item multi-auxiliary information, i.e., content and social graph, by defining a generative process. Specifically, the model learns deep latent embeddings from different item auxiliary information using variational auto-encoders (VAE), which could form a generative distribution over each auxiliary information by introducing a latent variable parameterized by deep neural network. Moreover, to recommend tags for new items, item multi-auxiliary latent embeddings are utilized as a surrogate through the item decoder for predicting recommendation probabilities of each tag, where reconstruction losses are added in the training phase to constrain the generation for feedback predictions via different auxiliary embeddings. In addition, an inductive variational graph auto-encoder is designed to infer latent embeddings of new items in the test phase, such that item social information could be exploited for new items. Extensive experiments on MovieLens and citeulike datasets demonstrate the effectiveness of our method.

CCS Concepts: • Information systems → Multimedia information systems;

Additional Key Words and Phrases: Tag recommendations, variational auto-encoder, hybrid systems, deep generative models

ACM Reference format:
Jing Yi, Xubin Ren, and Zhenzhong Chen. 2023. Multi-auxiliary Augmented Collaborative Variational Auto-encoder for Tag Recommendation. ACM Trans. Inf. Syst. 41, 4, Article 106 (April 2023), 25 pages.
https://doi.org/10.1145/3578932

1 INTRODUCTION

The activity that users annotate items, such as movies and articles, with some key words is called tagging. Online systems like Movielens¹ allow users to tag items. Appropriate tags can facilitate

¹https://movielens.org/.
Fig. 1. On the left are exemplars of item tagging in articles. On the right is the illustration of social interactions between items. Auxiliary information could be obtained by social links among items except for item content information.

content organization, retrieval, consumption, and so on [2, 13, 29, 30, 49]. Tag recommenders aim to recommend suitable tags for items to help users tag more easily by creating convenient annotations. Moreover, tag recommenders help to increase the quality of the generated tags and consequently improve the quality of the information retrieval (IR) services such as search [23], automatic classification [10], and content recommendation [11] that rely on tags as the data source. Therefore, it is very important to recommend suitable tags for items, especially in the Internet era that contains a great number of items.

Tag recommenders assist the tagging process by utilizing historical interactions between items and tags that are given to the object by online users. Collaborative information contained in item-tag interactions reflects the items’ common relatedness of the tag, which has been shown the effectiveness for tag recommendations [34, 35, 40]. Among them, Rendle and Schmidt-Thieme [35] proposed a pairwise interaction tensor factorization (PITF) model and adapted the Bayesian Personalized Ranking (BPR) framework. However, the sparse feedback of item-tag interactions and the cold-start problem in tag recommendations make systems using only collaborative information hard to work. However, content-based tag recommendations [22, 41] have been widely studied. Hassan et al. [14] modeled textual content in scientific articles using bidirectional Gated Recurrent Units (bi-GRUs) with word-level and sentence-level attention mechanisms. Tang et al. [41] proposed a content-based tag recommender system that utilized GRU layers to extract semantic and structural features in text content, where tag correlations and tag-content overlapping were considered.

However, hybrid methods that could benefit from both collaborative information and content information have been less explored. In the face of huge sets of items and tags, item-tag interactions are very sparse. Therefore, only utilizing the interaction information to recommend suitable tags for items is not optimal. Coupling item content information and collaborative information is important to alleviate the sparsity problem. Moreover, the item has multiple auxiliary information. Beyond the content information (e.g., textual or visual contents of the item), the social graph information, as illustrated in Figure 1, contains plenty of social-oriented information, and fully exploiting them can maximally benefit the tag recommendation process. Among them, collaborative topic regression (CTR) [43] used a Bayesian probabilistic model to couple the textual information learned by latent Dirichlet allocation (LDA) and the collaborative information learned by
probabilistic matrix factorization (PMF). CTR-SR [46] extended it by integrating social graph information of items using a linear model with limited representation abilities.

Considering that previous work using linear models is not optimal in utilizing auxiliary information and interaction information, how to more efficiently model the rich nonlinear correlation of item-tag to improve the performance of recommendation is a problem. With the help of the flexibility to integrate different information of the Bayesian probabilistic generative process and the powerful feature learning capabilities of deep learning methods, in this article, we propose to use deep generative models to learn the latent variables of item collaborative and multi-auxiliary information by defining a generative process to couple them. In this way, the collaborative information contained in item-tag interactions and relevant item auxiliary features could be comprehensively extracted with a deep generative model for tag recommendations. Variational Auto-Encoder (VAE) [18] and Variational Graph Auto-Encoder (VGAE) [19] are employed to model the content and social graph information of items, respectively, where a generative distribution over each auxiliary information can be obtained by introducing a latent variable parameterized by the deep neural network. Moreover, we substitute the linear PMF model with a multinomial VAE (Mult-VAE) model inspired by Reference [25] to better capture the item collaborative information from the sparse item-tag ratings. Specifically, we utilize a multinomial likelihood variational auto-encoder that assumes the item-by-tag interactions of each item follow a multinomial distribution such that a higher probability mass could be put on the observed interactions. The deep latent variables learned from item multi-auxiliary information are injected into the modeling of latent item embeddings that contain collaborative information by defining a generative process, which bridges the hybrid information in a unified framework. Variational inference is utilized for parameter estimation.

Another challenge comes with how to recommend tags for new items that do not have collaborative information during the test phase. Especially for tag recommendations, new items spring up soon. Therefore, extending hybrid recommenders to deal with totally new items is necessary and it is worth studying how to use item multi-auxiliary information to recommend new items without significantly reducing the recommendation performance.

To solve the above challenges, we propose a Multi-Auxiliary Augmented Collaborative Variational Auto-encoder (MA-CVAE) for tag recommendations. Specifically, we define a probabilistic generative process to couple item collaborative information and multiple auxiliary information into a unified framework, where deep generative models are utilized to model different information. Item auxiliary information consists of various available side information of items, such as content information (e.g., textual or visual content of the item), as well as the social graph information (e.g., co-consumption of users between items, citations between articles, or co-director between movies). The Product-of-Experts (PoE) principle is exploited to integrate item multi-auxiliary Gaussian variables, since the mean vector of the product embedding is a weighted sum of the semantic information in the content and social graph according to their informative levels for the recommendation. In addition, to solve the item cold-start problem, we use a tightly coupled hybrid model to constrain the similarity of collaborative and content auxiliary embeddings during training. Moreover, we extend VGAE to inductive VGAE through sub-graph sampling and neighborhood aggregation so new item nodes can be inferred during testing. We further add the generative losses of each auxiliary embedding through the item decoder, where the generation of recommendation probabilities for each tag could be further strengthened via multiple auxiliary information for new items.

The main contributions of our method are as follows:

- By defining a probabilistic generative process, item multiple auxiliary information (content and social graph) are integrated into the modeling of collaborative information, where a
deep hybrid model (MA-CVAE) is proposed to fully exploit the nonlinear correlations between items and tags. Specifically, the closeness of the latent item variable and each auxiliary variable is deduced to be achieved by dual mean square error (MSE) loss within a tightly coupled framework. Therefore, item auxiliary information could be exploited to alleviate the sparsity and make better tag recommendations.

- To address the cold-start problem of items, item multi-auxiliary latent embeddings are utilized as a surrogate through the item decoder for predicting recommendation probabilities of each tag. Specifically, generative losses of different item auxiliary embeddings through the item decoder are added in the training to enhance the prediction ability of interaction feedback without collaborative information. Moreover, an inductive variational graph auto-encoder is proposed to solve the problem of adding nodes in the graph by sub-graph sampling and neighborhood aggregation. In this way, item social graph information along with the item content information could be fully exploited to better infer the embeddings of new items for cold-start recommendations.

- As a Bayesian probabilistic model, the tight coupling between item collaborative information and multiple auxiliary information is realized with deep generative models to address the sparsity and cold-start problems of tag recommenders. The dual optimization regularizes the modeling of collaborative information to prevent bad model solutions due to sparsity by item multi-auxiliary embeddings. In turn, ratings help the modeling of item multi-auxiliary embeddings to learn more recommendation-oriented embeddings, which facilitates the item cold-start recommendation. Extensive experiments have verified the effectiveness of our method in tag recommendations.

The rest of the article is organized as follows: Section 2 reviews the related work. Section 3 describes the proposed MA-CVAE model in detail, and Section 4 presents the experimental evaluations. Section 5 summarizes the article with conclusions.

2 RELATED WORK

In this section, we first review the related work about tag recommendation (TR) from three aspects, i.e., collaborative-based TR that merely relies on the item-tag interaction matrix, content-based TR that utilizes item content information for modeling content filtering, and hybrid methods that merge two information sources. Then, we summarize the previous work on variational recommender systems where collaborative-based, content-based, and hybrid methods are all included. Specifically, collaborative-based methods utilize variational auto-encoder for reconstructing sparse user-aware interactions among items, and content-based methods apply the powerful feature extraction ability of VAE for item-side or user-side feature extraction. Hybrid methods seek to utilize both rating information and auxiliary information.

2.1 Tag Recommendation

Existing tag recommendation tasks mainly include object-centered and personalized tasks [2, 50]. This article focuses on the object-centered tag recommendation without taking users’ preferences into account. From the perspective of exploited data sources, tag recommendation can be roughly categorized into collaborative-based, content-based, and hybrid methods. Collaborative-based methods utilize the existing tagging history as input. Rendle and Schmidt-Thieme [35] presented the factorization model PITF (Pairwise Interaction Tensor Factorization) and adapted the Bayesian Personalized Ranking (BPR) framework. Fang et al. [9] exploited the Gaussian radial basis function to increase the model’s capacity, which could be considered as a nonlinear extension of Canonical Decomposition. Chen et al. [6] integrated the graph neural networks into the
pairwise interaction tensor factorization model to better capture tagging patterns in the item-tag interaction graph.

Content-based methods model the semantic and structural content of items to recommend suitable tags. Maity et al. [28] learned the content representation from question title and body to recommend appropriate question tags on Stack Overflow. Yu et al. [48] extended labeled latent Dirichlet allocation (LLDA) [32] by explicitly specifying several relevant words for a given tag. Then, it allowed generating the content directly using these words, where the tags were treated as the supervision information of the corresponding content. Khezrian et al. [17] used the BERT pre-training technique in tag recommendation tasks for online Q&A and open-source communities for the first time. Hassan et al. [14] adopted deep recurrent neural networks, i.e., bi-GRUs, to encode titles and abstracts of scientific articles into semantic vectors for enhancing the recommendation task. Specifically, word-level attention and sentence-level attention were utilized for better modeling item contents. Tang et al. [41] combined RNN with topical distributions to learn text representations, where the content-tag overlapping and the tag correlation were further considered. Nie et al. [29] constructed a hypergraph by integrating multiple facets, including Question-Answer content analytics, tag-sharing information, as well as user connections, and then selected candidate tags by simultaneously considering informativeness, stability, and closeness. Nie et al. [30] further considered the newly posted question tagging problem by learning the question and topic embeddings from deep neural networks and projecting them into the same space for a similarity measure.

Hybrid methods integrate collaborative and content information for better recommendations. Song et al. [37] introduced a graph-based method, which represented the tagged data into two bipartite graphs of (document, tag) and (document, word). Then, it found document topics by leveraging graph partitioning algorithms. Zhang et al. [52] proposed an optimization model to integrate item contents into user interests where different impacts of item features on user preference toward an item have been extracted for item recommendation. Sun et al. [38] proposed a hierarchical attention model, where collaborative embeddings and content embeddings were fused through an attention module, where collaborative embeddings and content embeddings were fused through an attention module. Wang et al. [43] proposed a CTR model to recommend articles to users, which combined a collaborative filtering matrix decomposition algorithm based on hidden factors and a content analysis algorithm based on probabilistic topic models. These two models were unified into a probabilistic generative framework, which could weigh the importance of the content of the article and the collaborative information. Wang et al. [46] adapted the framework of CTR for tag recommendation problems to seamlessly integrate both item-tag matrix information and item content information, where social networks between items were integrated into the framework for better tag recommendation. Considering the limited representational capabilities of the linear method and topic model to learn interactions and content for the recommendation task, we utilize deep generative models to learn the hidden variables of item collaborative information, content, and social graph. Moreover, by introducing multinomial VAE [25], the generative process could be revised to item-based only with an item decoder to predict recommendation probabilities of each tag. Therefore, compared to probabilistic matrix factorization models, tag latent embeddings are excluded, which has more flexibility and conciseness.

2.2 Variational-based Recommendation

VAEs [18] generate observation \( x \) via the latent variable \( z \), which construct a variational posterior of the unobserved variable \( z \) given the input \( x \) to approximate the true posterior \( p(z|x) \) due to the nonlinearity of the conditional likelihood. Specifically, the generative model uses a decoder network to reconstruct \( x \) from \( z \). The inference model also uses a neural network as an encoder to learn parameters of the approximated posterior distribution \( q_\phi(z|x) \). By minimizing the
Kullback-Leibler (KL) divergence between the parametric posterior \( q(z|x) \) and the true posterior \( p(z|x) \), the goal of VAEs to maximize the log marginal likelihood \( \log p(x) \) deduces to minimize an Evidence Lower BOrder (ELBO):

\[
L(x^{(i)}; \theta, \phi) = \mathbb{E}_{z \sim q_{\phi}(z|x^{(i)})} [\log p_{\theta}(x^{(i)}|z)] - \text{KL}(q_{\phi}(z|x^{(i)})||p(z)),
\]

where \( x^{(i)} \) is a sample of the observed variable \( x \), \( \theta \), and \( \phi \) are parameters of the decoder and encoder. The first term on the right-hand side (RHS) of Equation (1) represents the reconstruction error, which encourages the latent variable \( z \) to better generate the observed data \( x \) for strong feature-learning ability. The second term on the RHS of Equation (1) is a regularization term, which penalizes the approximated posterior probability \( q_{\phi}(z|x^{(i)}) \) to be far away from prior probability \( p(z) \). The reparameterization trick is applied to remove the stochastic sampling from the formation, and thus the gradient could be calculated and back-propagation could be performed. Furthermore, by introducing a hyperparameter \( \beta \) before the second term of RHS of Equation (1), VAE can be extended to beta-VAE\[16\], which controls the balance of reconstruction and regularization terms.

VAEs\[18\] have been extended for recommendations and other tasks\[4, 44\] due to the modeling ability for high-dimensional data. Liang et al.\[25\] assumed the item ratings of a user to follow a multinomial distribution and designed a VAE-based framework for interaction-based modeling. Askari et al.\[1\] proposed a Joint Variational Auto-encoder (JoVA), which was an ensemble of two VAEs to jointly learn both user and item representations to predict user preferences. Other efforts have been made to model features with VAE in recommendations. Li and She\[24\] proposed a probabilistic matrix factorization-based model, where item content features have been integrated to alleviate the sparsity problem. Specifically, item content features were modeled with VAE to form item content variables, where item collaborative variables and content variables were added together for fusing two information sources. Chen et al.\[8\] proposed a deep generative model, LVSM to address the item cold-start top-N recommendation problem. The model could capture local aspects of items and measure global item similarity based on deep representations extracted from item features through a variational EM procedure.

VAEs are also utilized for the modeling of hybrid recommendations. Chen and de Rijke\[7\] proposed to simultaneously recover user ratings and side information of items by using a VAE, where user ratings and side information were encoded and decoded collectively through the same inference network and generation network. Due to the heterogeneity of user ratings and side information, the final layer of the generation network followed different distributions. Lee et al.\[21\] proposed to model the auxiliary information and the implicit user feedback in a variational approach for hybrid item recommendations. Two strategies were introduced for integrating multiple item auxiliary features: (1) a conditional VAE\[36\] that modeled the conditional distribution given another modality; (2) a joint multimodal VAE (JMVAE)\[39\] that modeled the joint distribution of different modalities by a single latent variable. Ma et al.\[27\] introduced a partial VAE, which could efficiently handle the missing ratings using amortized partial inference technique without relying on ad hoc assumptions such as Zero Imputation. The authors further designed a partial inference network for auxiliary distribution by a permutation invariant set function to encode auxiliary information on the user side and item side for better hybrid recommendations. Wang et al.\[45\] proposed a personalized online course recommender system, where the extracted latent representations of the employees’ competencies from their skill profiles and the personal demands of employees were integrated into a unified Bayesian inference view. The graphical model comprehensively combined the conventional latent factor models (LFM)-based collaborative filtering method with auto-encoding variational inference for topic modeling. However, the VAE framework is not currently used for hybrid tag recommendation and, therefore, we aim to develop a
Fig. 2. On the left is the framework of the proposed MA-CVAE. Item content $x$ and social graph $G^s$ are different auxiliary information that are augmentations of collaborative information contained in item-tag ratings $r$. On the right is the zoom-in of the inference network and generation network of VAE for content information and inductive VGAE for social information.

3 MULTI-AUXILIARY AUGMENTED COLLABORATIVE VARIATIONAL AUTO-ENCODER

In this section, we describe the proposed multiple auxiliary information augmented variational auto-encoder for tag recommendations as shown in Figure 2. MA-CVAE is a generative model where different item auxiliary information, i.e., content and social graph, are generated through item auxiliary variables, and item-tag ratings are generated by latent item variables. Different item auxiliary information is injected into the modeling of latent item variables by adding multi-auxiliary variables and collaborative variables, which bridges hybrid information together into a unified framework. Notations used in this article are summarized in Table 1. Note that we use Capital non-boldface symbols such as $R$ to denote the corresponding random vectors of $r$, and $R^m$ is used to denote the random matrix for stacked $r$. Capital boldface symbols such as $R$ are used to denote matrices.

3.1 Problem Statements

Let $\mathcal{V}$ and $\mathcal{T}$ denote the set of $I$ items and $J$ tags, respectively. Assume we have $C = [c_1, c_2, \ldots, c_I]$, where $c_i$ denotes the content of item $i$. The item social graph $G$ is a multigraph where multiple edges are distinguished by the attributes of the links. Here, we denote the intrinsic links as the edges between items once after the item is produced such as citations between scientific articles and co-star information between movies. While extrinsic links are denoted as edges between items after user co-consumption of the items such as two items with five or more users interacting in common could be linked to an edge. The adjacency matrix is $A \in \mathcal{R}^{I \times I}$, where we merge these two types of links between items and normalize the edge weights to be $\{0, 1\}$. The collaborative information is represented by an item-tag tagging matrix $R = [r_{ij}]_{I \times J}$, where $r_{ij} = 1$ means that the tag $j$ is tagged to the item $i$. Given the item $i$, the task of tag recommendation is to find a list of tags $T(i) \subseteq \mathcal{T}$ that is likely to be annotated to the item $i$ by calculating the relatedness between
Table 1. Notations Used in Our Method

| Notation | Description |
|----------|-------------|
| V        | item set with I items |
| T        | tag set with J tags |
| K        | dimension of latent variables |
| v        | the latent item variable with the matrix of V |
| c        | the latent item content variable with the matrix of C |
| s        | the latent item social variable with the matrix of S |
| r        | the rating vector with the matrix of R |
| x        | the content vector with the matrix of X |
| Gs       | the stacked sub-graph matrix with the full-graph matrix of G |
| A        | the adjacency matrix of the full social graph G |
| V        | latent item random variable |
| C        | latent item content random variable |
| Sm       | stacked latent item social random variables in the sub-graph |
| R        | rating random variable |
| X        | content random variable |
| Gms      | stacked social random variables in the sub-graph |
| \(\theta_v, \theta_c, \theta_s\) | trainable generative parameters for item, content, and social variables |
| \(\phi_v, \phi_c, \phi_s\) | trainable inference parameters for item, content, and social variables |
| \(p_{\theta_v}(R|V)\) | conditional likelihood of the ratings, with \(\theta_v\) to be trainable weights |
| \(p_{\theta_c}(X|C)\) | conditional likelihood of the content, with \(\theta_c\) to be trainable weights |
| \(p_{\theta_s}(G^{m_s}|S^{m_s})\) | conditional likelihoods of the sub-graph, with \(\theta_s\) to be trainable weights |
| \(p(V)\) | prior of the item random variable |
| \(p(C)\) | prior of the item content random variable |
| \(p(S^{m_s})\) | priors of the item social random variables in the sub-graph |
| \(q_{\phi_v}(V|R)\) | variational posterior of item random variable |
| \(q_{\phi_c}(C|X)\) | variational posterior of item content random variable |
| \(q_{\phi_s}(S^{m_s}|G^{m_s})\) | variational posteriors of item social random variables in the sub-graph |

the item \(i\) and the candidate tag \(j\). In this article, we utilize item content information \(X\), item social graph \(G\), and collaborative information \(R\) for hybrid tag recommendations.

### 3.2 Generative Process

We consider item content and social graph to be generated by their latent content variables and social variables through generative networks, respectively. Specifically, to utilize the content information of items, we assign a latent content variable \(c\) for each item. To further employ the social network of items, which contains plenty of contiguous relations between items, we assign a latent social variable \(s\) for each item. We draw the latent item content variable and item social variable in a latent low-dimensional space of \(K\) dimensions from Gaussian distributions:

\[
\begin{align*}
    c & \sim N \left( 0, \lambda_C^{-1}I_K \right), \\
    s & \sim N \left( 0, \lambda_S^{-1}I_K \right),
\end{align*}
\]

where \(\lambda_C\) and \(\lambda_S\) are precision vectors of the item content variable and social variable. The content of the item \(x\) is then generated from its latent content variable \(c\) through a generation neural network, e.g., **multiple layer perceptrons (MLP)** as with variational auto-encoder [18]. For the
generation of item social graph, sub-graph sampling as in Reference [12] is utilized for inductive learning and mini-batch training. The sub-graph $G^s$ of the item (here, we focus on the generation of the edges, i.e., the adjacency matrix $A^s$ of the sub-graph) can be generated by latent item social variables $S^s$, which consist of variables of the item and its sampled neighbors, through an inner product decoder as with variational graph auto-encoder [19]. Exactly, the content and social graph of the item are generated from its latent content variable $c$ and social variables $S^s$ through generation neural networks parameterized by $\theta$:  
\[ x \sim p_{\theta_c}(X|C), \]
\[ G^s \sim p_{\theta_s}(G^s|m^s), \]
where $G^s$ is a sampled sub-graph from the social graph $G$ with neighbors of the item node, which will be further discussed in the following inductive variational graph auto-encoder section. $G^s$ and $m^s$ are stacked social random variables and latent social random variables for the sub-graph.

To comprehensively utilize the item content and social network information, a product-of-experts principle is employed to fuse $c$ and $s$ as a multi-auxiliary variable. For Gaussian variables, the product is also Gaussian where the new mean becomes $\mu_m = (\mu_c \lambda_C + \mu_s \lambda_S) / (\lambda_C + \lambda_S)$, and the new variance becomes $\lambda_m = (\lambda_C \lambda_S) / (\lambda_C + \lambda_S)$. Since the mean of a Gaussian variable depicts its semantic structure and variance denotes uncertainty theoretically, the mean vector of the multi-auxiliary variable is a weighted sum of the semantic information in the content and social graph according to their informative levels for the recommendation.

To fully explore the collaborative information, we explicitly introduce $v^\dagger$ to embed the item collaborative information for the item and draw it from a Gaussian distribution as:
\[ v^\dagger \sim N(0, I_K). \]

Then, we set the latent item variable $v$ to be composed of both item collaborative and multi-auxiliary latent variables as follows:
\[ v = v^\dagger + \text{PoE}(c, s, \lambda_C^{-1} I_K, \lambda_S^{-1} I_K). \]
Given the product of $c$ and $s$, and then, the latent item variable $v$ follows the conditional distribution $N(\text{PoE}(c, s, \lambda_C^{-1} I_K, \lambda_S^{-1} I_K), I_K)$, which is the key to introduce mutual regularization between $v$ and $\text{PoE}(c, s, \lambda_C^{-1} I_K, \lambda_S^{-1} I_K)$ in the Maximum a Posteriori (MAP) objective. Moreover, with the PoE principle, we could define the conditional generative process of $v$ given $c$ and $s$ to be:
\[ p(v|c, s) = p(v|c)p(v|s). \]

The latent item variable $v$ is then transformed via a non-linear function, which is parameterized by a neural network, to produce a probability distribution over $J$ tags as in Reference [25]. The item-by-tag interaction of the item is assumed to be:
\[ \pi(v) \propto \text{Softmax}(\text{NN}(v; \theta_v)), \]
\[ r \sim \text{Multinomial}(N_s, \pi(v)), \]
where $\text{NN}$ denotes Neural Network and we utilize an MLP for implementation. $\pi(v)$ is computed by an $\text{NN}(v; \theta_v)$ with the output normalized via a Softmax function. $\theta_v$ is the trainable weights of the deep generation network. Given the total number of interactions $N_s$ from the item, $r$ is sampled from the Multinomial distribution parameterized by $\pi(v)$. The multinomial distribution has been proven to be good at modeling implicit feedback data, since the model would assign more probability mass to tags that are more likely to be interacted, which will perform well under the top-$N$ ranking loss.
With the above generative process defined, the joint distribution of observable and hidden variables of the item and its neighbors in the social graph can be formulated as follows:

\[
p_{\theta}(V, C, S^{m*}, R, X, G^{m*}) = p_{\theta_c}(X|C)p(C)p_{\theta_s}(G^{m*}|S^{m*})p(S^{m*})p(V|C, S)p_{\theta_v}(R|V),
\]

where \(\{R, X, G^{m*}\}\) is the set of all observed variables, \(\{V, C, S^{m*}\}\) is the set of all latent variables needed to be inferred. The joint distribution of item content variable \(p_{\theta_c}(X, C)\) is factorized as \(p_{\theta_c}(X|C)p(C)\), with the prior distribution \(p(C)\) of latent item content variable to be Gaussian distribution and the stochastic content decoder \(p_{\theta_c}(X|C)\) to be parameterized by neural networks. Similarly, the joint distribution of item social variables \(p_{\theta_s}(G^{m*}|S^{m*})\) is factorized as the conditional generative distribution \(p(V|C, S)\) of latent item variable, and the item decoder \(p_{\theta_v}(R|V)\). Moreover, \(p_{\theta_s}(G^{m*}|S^{m*})\) can be factorized into the product of per node distributions \(\Pi p_{\theta_s}(G|S)\) due to the assumption of marginal independence among items in the sub-graph.

### 3.3 Inference Process

Since the generative processes of the latent item variable and multiple auxiliary variables are nonlinear (i.e., parameterized as deep neural networks), the posterior distributions for \(V, C, S\) are intractable. Therefore, we resort to variational inference with an approximated posterior as \(q_{\phi}(V, C, S^{m*}|R, X, G^{m*})\), where we assume the approximated posterior to come from tractable families of distributions parameterized by deep neural networks. Then, the distribution closest to the true posterior measured by the KL-divergence could be found in those families [3]. According to conditional independence, the joint posterior of all hidden variables can be factorized into the product of three compact parts as follows:

\[
q_{\phi}(V, C, S^{m*}|R, X, G^{m*}) = q_{\phi_c}(V|R) \cdot q_{\phi_s}(C|X) \cdot q_{\phi_v}(S^{m*}|G^{m*}).
\]

Previous work [3] proves that the minimization of the KL-divergence is equivalent to the maximization of the Evidence Lower BOund (ELBO) [18]:

\[
\mathcal{L} = \mathbb{E}_{q_{\phi}}[\log p_{\theta}(V, C, S^{m*}, R, X, G^{m*})] - \log q_{\phi}(V, C, S^{m*}|R, X, G^{m*})
\]

\[
= \mathbb{E}_{q_{\phi}}[\log p_{\theta_c}(X|C) + \log p_{\theta_s}(G^{m*}|S^{m*}) + \log p(V|S) + \log p(V|C) + \log p_{\theta_v}(R|V)]
\]

\[
- \mathbb{E}_{q_{\phi}}(q_{\phi_c}(C|X)||p(C)) - \mathbb{E}_{q_{\phi_s}}(q_{\phi_s}(S^{m*}|G^{m*})||p(S^{m*})) - \mathbb{E}_{q_{\phi_v}}(\log q_{\phi_v}(V|R)).
\]

By substituting the joint distribution of Equation (11), we can rewrite the ELBO as in Equation (13). \(\phi\) consists of the parameters of the inference networks where \(q_{\phi}\) is an abbreviation for \(q_{\phi}(V, C, S^{m*}|R, X, G^{m*})\). As with the CVAE paper [24], the entropy of \(q_{\phi}(V|R)\) is regarded as a constant and omitted from the ELBO.

### 3.4 Maximum A Posteriori Estimation

Maximum a Posteriori estimation can be performed by considering the variational distributions of \(q_{\phi_c}(C|X), q_{\phi_s}(S^{m*}|G^{m*})\) and \(q_{\phi_v}(V|R)\), as well as maximizing the objective with respect to \(C, S^{m*}\) and \(V\) by an EM-style algorithm using block coordinate ascent. Since each variable in our article is defined as a Gaussian distribution, and the negative logarithmic likelihood of the Gaussian
distribution is L2-norm loss, the objective in Equation (13) could further be rewritten as:
\[
L_{MAP}(V, C, S^{m*}; \theta, \phi) = \mathbb{E}_{q_{\phi}(V|R)}[\log p_{\theta}(R|V)] + \mathbb{E}_{q_{\phi}(C|X)}[\log p_{\theta}(X|C)] - KL(q_{\phi}(C|X)||p(C)) \\
+ \mathbb{E}_{q_{\phi}(S^{m*}|G^{m*})}[\log p_{\theta}(G^{m*}|S^{m*})] - KL(q_{\phi}(S^{m*}|G^{m*})||p(S^{m*})) \\
- \lambda_C \mathbb{E}_{q_{\phi}(C,V|X,R)} \|V - C\|_F^2 - \lambda_S \mathbb{E}_{q_{\phi}(S,V|G,R)} \|V - S\|_F^2,
\]
where \(\lambda_C\) and \(\lambda_S\) are precision vectors of item content variables and social variables, respectively. \(F\) is the Frobenius norm and the L2-norm is utilized here. \(\mathbb{E}_{q_{\phi}(V|R)}[\log p_{\theta}(R|V)]\), \(\mathbb{E}_{q_{\phi}(C|X)}[\log p_{\theta}(X|C)]\) and \(\mathbb{E}_{q_{\phi}(S^{m*}|G^{m*})}[\log p_{\theta}(G^{m*}|S^{m*})]\) represent the inference and generation process of item variable, content variable, and social variables, respectively, in the form of variational auto-encoders.

We first fix the parameters related to item auxiliary variables \((C\) and \(S\)) and optimize the parameters related to item variable \(V\) as the Multinomial-VAE. The objective for \(V\) thus becomes:
\[
L_{item} = \mathbb{E}_{q_{\phi}(V|R)}[\log p_{\theta}(R|V)] - \lambda_C \mathbb{E}_{q_{\phi}(V|X,R)} \|V - \hat{C}\|_F^2 - \lambda_S \mathbb{E}_{q_{\phi}(V|G,R)} \|V - \hat{S}\|_F^2,
\]
where \(\hat{C}\) equals the mean vector produced by the item content inference network, and \(\hat{S}\) equals the mean vector produced by the item social graph inference network. From Equation (15), we could see that the closeness between the latent item embedding and each latent auxiliary embedding is achieved by a mean square error (MSE) loss, which could leverage item multiple auxiliary information into the embedding process of collaborative information. In this way, multiple auxiliary information is incorporated to further alleviate the sparsity of the implicit feedback. Furthermore, to better strengthen the closeness of two components and infer representations for new items using only item auxiliary information, we introduce an additional constraint. Specifically, we add the implementation of the \(p_{\theta}(R|C)\) and \(p_{\theta}(R|S)\) to the objective of the item end, using the same decoder as the item decoder \(p_{\theta}(R|\cdot)\). By adding the additional reconstruction losses, the generation of reconstruction probabilities for each tag could be inferred via item multi-auxiliary embeddings for new items. Therefore, cold-start items that do not contain any collaborative information could be recommended with suitable tags.

Then, we fix the parameters of variable \(V\) and latent item social variable \(S\) to optimize the item content VAE objective. We isolate the terms related to \(C\) and the objective thus becomes:
\[
L_{content} = \mathbb{E}_{q_{\phi}(C|X)}[\log p_{\theta}(X|C)] - KL(q_{\phi}(C|X)||p(C)) \\
- \lambda_C \mathbb{E}_{q_{\phi}(C|X)} \|\hat{V} - C\|_F^2.
\]

Finally, we fix the parameters of item variable \(V\) and content variable \(C\), and the item social inductive VGAE objective can be optimized. The objective for VGAE after isolating the terms related to \(S\) becomes:
\[
L_{social} = \mathbb{E}_{q_{\phi}(S^{m*}|G^{m*})}[\log p_{\theta}(G^{m*}|S^{m*})] - KL(q_{\phi}(S^{m*}|G^{m*})||p(S^{m*})) \\
- \lambda_S \mathbb{E}_{q_{\phi}(S^{m*}|G^{m*})} \|\hat{V}^{m*} - S^{m*}\|_F^2,
\]
where \(\hat{V}^{m*}\) denotes the stacked mean vector for item variables, which is inferred by the item inference network. For both \(L_{content}\) and \(L_{social}\), the objective consists of three parts: (1) the reconstruction part learns a latent Gaussian variable to reconstruct the input; (2) the KL-divergence part
assumes a Normal Gaussian as the prior of the latent variable, which penalizes learned latent variable to encode excessive noisy information; (3) the MSE part constricts the closeness of the item variable and each auxiliary variable, which tightly couples the collaborative information and multi-auxiliary information by mutual constraints. Therefore, collaborative information in ratings could help the modeling of item multi-auxiliary embeddings to learn more recommendation-oriented embeddings.

### 3.5 Reparameterization Trick

We utilize the reparameterization trick [18] to make the sampling outside the model for amendable gradient-based optimization. For the three Gaussian latent variables \( z \in \{ v, c, s \} \), we calculate their mean \( \mu_z \) and logarithm of standard deviation \( \sigma_z \) via the corresponding encoder network, which could then be scaled by samples drawn from a fixed Gaussian distribution. The transformation is:

\[
  z([\mu_z, \sigma_z], \epsilon) = \mu_z + \epsilon \odot \sigma_z, \tag{18}
\]

where \( \epsilon \sim \mathcal{N}(0, I_K) \). Through reparameterization, the random elements are separated, where differentiation is amendable in the backward propagation. According to previous work [18], we utilize the Monte Carlo gradient estimator to sample for the expectations in the objective, which has shown that the variance of the sampling is small, and one sample for each data point suffices for convergence.

With the reparameterization trick, we can iteratively optimize the Mult-VAE part, the item content VAE part, and the item social inductive VGAE part for tag recommendation, which forms a tightly coupled hybrid model. In this way, the item content variable and social variable are coupled into the latent item variable with the MSE term. That is, each item auxiliary representation can improve the performance of interaction-based VAE, especially on items with sparse interactions. Moreover, collaborative information can guide the learning of item multiple auxiliary feature mapping. Specifically, the latent item variable that contains collaborative information is further exploited to constrain the updating of parameters of networks for the item content variable and item social variable. Therefore, item collaborative and auxiliary information could be exploited in a complementary manner. More item content and social network information could be extracted to assist the expression of items for sparse datasets that contain less collaborative filtering information and vice versa. In this way, better recommendations could be made by mutual restraint. Furthermore, \( \lambda_S \) and \( \lambda_C \) could be viewed as hyper-parameters that balance the item’s social and content components. The training procedure of MA-CVAE is summarized in Algorithm 1.

### 3.6 Inductive Variational Graph Auto-encoder

To handle the transductive characteristic of variational graph auto-encoder [19], which cannot generalize to unseen nodes by learning embeddings for each node in the graph during training, we propose an inductive VGAE framework inspired by Reference [12]. Specifically, we utilize a graph sampler to sample a sub-graph of the nodes that are to be embedded in the mini-batch, and then an aggregation function is employed to fuse features from the nodes’ local neighbors. During training, the parameters of the aggregation function could be learned through mini-batch gradient descent. Therefore, the embeddings of new nodes that are unseen from the training set could be obtained by leveraging node features of their neighbors using the learned aggregation operation. Meanwhile, the inductive variational graph auto-encoder can be applied for large graphs in recommendation systems, such as the user-item bipartite graph, through mini-batch training and inference.

We exploit the features of items in the item social graph, where textual Term Frequency-Inverse Document Frequency (TF-IDF) attributes of items are utilized as node features. We employ an inner-product decoder as the generative process of our inductive VGAE. Inspired by
ALGORITHM 1: Training MA-CVAE with SGD.

Input: Item-tag interaction matrix $R$; Item content matrix $X$; Item social graph $G = \{A, X\}$

Randomly initialize $\theta, \phi$.

while not converged do

  // Update item Mult-VAE.
  forall the $v \in V$ do
    Randomly sample a batch of items with $R$.
    Infer the content mean $C$ and social mean $S$.
    Sample $V$ via reparametrization trick. Compute the item loss via Equation (15).
  end

  Compute the gradient of the item loss.
  Update $\theta_v, \phi_v$ by taking stochastic gradient steps.

  // Update Content VAE.
  forall the $v \in V$ do
    Randomly sample a batch of items with $X$.
    Infer the item mean $V$ and social mean $S$.
    Sample $C$ via reparametrization trick. Compute the content loss via Equation (16).
  end

  Compute the gradient of the content loss.
  Update $\theta_c, \phi_c$ by taking stochastic gradient steps.

  // Update Social inductive VGAE.
  forall the $v \in V$ do
    Randomly sample a batch of items for sub-graph $G^s$.
    Infer the item mean $V^s$ and content mean $C^s$.
    Conduct the neighbor aggregation via Equation (20).
    Sample $S^s$ via reparametrization trick. Compute the social loss via Equation (17).
  end

  Compute the gradient of the social loss.
  Update $\theta_s, \phi_s$ by taking stochastic gradient steps.

end

return $\theta, \phi$

Bayesian personalized ranking (BPR) loss [33], which is proven to be more suitable for recommendations, the decoder becomes:

$$
\log p(G^{m^r}|S^{m^r}) = \sum_{(v, v^+, v^-) \in R} \log \zeta \left( s_{v^+}^T s_{v^+} - s_v^T s_{v^-} \right),
$$

(19)

where $\zeta$ is an activation function to increase nonlinearities and $R = \{(v, v^+, v^-) | (v, v^+) \in E, (v, v^-) \notin E\}$ is a set of triplets of three items. $v$ is the target item to be embedded, $v^+$ is the $l$-hop neighbor of item $v$ that is sampled through Random Walk [31], and $v^-$ is the randomly sampled negative item that does not interact with item $v$.

For the encoder, we utilize a mean aggregation function to leverage features from neighbors, which can be presented as:

$$
s_k^v \leftarrow \zeta \left( W \cdot \text{MEAN} \left( \left\{ s_k^{v-1} \right\} \cup \left\{ s_k^{u-1}, \forall u \in N(v) \right\} \right) \right),
$$

(20)

where $N(v)$ is the set of item $v$’s neighbors in the sampled sub-graph. In this way, the item’s previous layer embedding $s_k^{v-1}$ is concatenated with the aggregated neighborhood vector $s_k^{u-1}$, which can be viewed as a skip connection [15]. It is worth noting that for new items, links between
items contain only intrinsic relations (e.g., citations for articles and co-star for movies) among items without extrinsic links (co-interactions of users for items), since a newly uploaded item is more likely to have no interactions by users. In this way, tag recommendations suggest suitable tags for items and then improve the click-through rate of the items.

3.7 Prediction

3.7.1 For Existing Items. Let $D$ be the observed data. With the MA-CVAE trained, the weights of the inference networks and generation networks of item implicit feedback, content, and social graph are learned. Then, the prediction for existing items that are existing in the training set becomes:

$$
\mathbb{E} [r|D] = p_{\theta_v} (r|\mu_v) + p_{\theta_v} \left( r| \frac{\mu_c \lambda_C + \mu_s \lambda_S}{\lambda_C + \lambda_S} \right),
$$

(21)

where $\mu_v$ denotes the mean vector of the latent item variables through the Mult-VAE encoder. $(\mu_c \lambda_C + \mu_s \lambda_S)/(\lambda_C + \lambda_S)$ calculates the mean vector of the product of item content variables and social variables.

3.7.2 For New Items. For totally new items, the product-of-experts of item content embeddings and social embeddings, which could be easily obtained by the inference network of item content and item social graph, are utilized to make predictions via the item decoder. The predictions can be represented as:

$$
\mathbb{E} [r|D] = p_{\theta_v} \left( r| \frac{\mu_c \lambda_C + \mu_s \lambda_S}{\lambda_C + \lambda_S} \right).
$$

(22)

4 EXPERIMENTS

To evaluate our proposed MA-CVAE, we conduct extensive experiments to answer the following research questions:

- **RQ1** How does MA-CVAE perform compared with the state-of-the-art methods for tag recommendations? Among them, collaborative-based, content-based, and hybrid methods are all included to make comprehensive comparisons.

- **RQ2** How does MA-CVAE perform under cold-start item scenarios? Comparisons are made among our proposed method and other content-based baselines for new items that are added to the catalog without any interaction. In this situation, collaborative-based models and hybrid methods that are not specifically designed for new items fail to recommend.

- **RQ3** Look inside the proposed MA-CVAE—how is the performance of MA-CVAE affected by the parameters $\lambda_C$ and $\lambda_S$, which influence the balance of item content information and social information? How is the interpretability of MA-CVAE when visualizing some examples?

4.1 Experimental Settings

4.1.1 Datasets. In our experiments, four real-world datasets are utilized to evaluate the effectiveness of our method.

- **MovieLens 20M** It is a stable benchmark dataset\(^2\) for recommender systems, abbreviated as ml-20m, which contains 20 million ratings rated by 138,000 users on 27,000 movies. We filter out the tagging information with the tag relevance scores provided by Reference [42] no bigger than 0.75 in the dataset to remove some noisy tags. We then collect the plot of movies as the textual content of items (movies) using the links in IMDB. The crew information is

---

\(^2\)https://grouplens.org/datasets/movielens/.
Table 2. Statistics of Evaluation Datasets after Preprocessing

| Dataset      | #Items | #Tags | #Item-Tag | Sparsity  |
|--------------|--------|-------|-----------|-----------|
| ml-20m       | 1,065  | 992   | 137,748   | 13.038%   |
| ml-25m       | 31,377 | 25,928| 1,015,676 | 0.125%    |
| citeulike-a  | 12,734 | 11,785| 195,139   | 0.130%    |
| citeulike-t  | 19,634 | 13,162| 220,377   | 0.085%    |

provided by IMDB,3 and we use the match information provided by ml-20m to correspond a moviId to an imdId for constructing the item social graph.

- **MovieLens 25M** This dataset is an extension of ml-20m, which contains about 25 million ratings and 1,093,360 tagging interactions from 62,423 movies by 162,541 users. We utilize all item-tag interactions annotated by users without filtering (with a vast majority of movie-tag pairs having relevance data less than 0.6) for keeping the huge amount of item-tag interactions and noisy characteristics of the dataset. Similar to ml-20m, we collect the plot of movies as the textual information and utilize links in IMDB to establish an item social graph.

- **citeulike-a** This dataset is from CiteULike, which helps users manage academic articles by creating their own collections of articles. On this platform, you can tag and rate your collected references, which is the source of our article-tag interactions. We use the collected tags and articles released by Reference [46], which contains tagging information, textual information (i.e., title and abstract), and information needed for item social graph (i.e., citations and user-item ratings) in the dataset.

- **citeulike-t** It is an extension of the citeulike-a dataset collected by Reference [46], which contains more items (articles) and tags than citeulike-a. However, the tagging interactions are more sparse, which makes the methods using only collaborative item-tag rating information even harder to perform.

We preprocess the content of items (i.e., text information) as in Reference [46], where we convert all characters to lowercase, remove the stop words, and conduct lemmatization. Finally, the top 8,000 distinct words are selected as our vocabulary for the citeulike-a dataset. For citeulike-t, ml-20m, and ml-25m, the vocabulary sizes are 20,000, 24,453, and 13,032. We then utilize the TF-IDF of the textual content as our content features for four datasets.

To exploit social information between items, we construct an item social graph using two types of attributes of links (i.e., intrinsic and extrinsic edges). For intrinsic links, we employ citation links existing among items that are available in citeulike to construct the citation social network (which is a directed graph) for academic articles. While on MovieLens, we exploit co-actor, co-writer, and co-director information as the inherent attributes of movies, where we assign a link between two items if they have one common actor, writer, or director. However, the extrinsic links are applied to construct the item co-consumption social graph. Specifically, we employ the co-interacted pattern of users where two items with four or more users interacting in common are linked by an edge for citeulike, and 50 for ml-20m and ml-25m. We then merge these two networks between items and normalize the weights to be 1 if there exists one edge in any of the two networks. After constructing the item social graph, the number of edges is 368,603, 355,181, 3,496,719, and 8,934,186 for citeulike-a, citeulike-t, ml-20m, and ml-25m, respectively. More specifically, we remove the tags used less than three times and the items without any links in the item social graph. The statistics after preprocessing of our four datasets are listed in Table 2.

---

3https://www.imdb.com/interfaces/.
4.1.2 Evaluation Protocols. For each dataset, we first split out 1,000 items for new items that do not exist in the training set. Then, we randomly select tags for each remaining item as training, validation, and test sets with a 6:2:2 ratio. Specifically, items in the training are treated as existing items, and items excluded in the training set are as new items. The tags in the validation and test set are all included in the training set.

We use Recall@N, NDCG@N, and MRR@N as our evaluation metrics as in Reference [38], where we do not choose Precision@N for the reason that an unobserved tag for an item may be, since the tag is not suitable for the item or that the tag is not considered by users to annotate to the item [47]. Recall@N calculates the hit ratio of the tested models:

$$\text{Recall}_v@N = \frac{\#\text{Hits}_v@N}{|y_v|},$$  \hspace{1cm} (23)

where $|y_v|$ denotes the number of all interacted tags in the test set for item $v$. Hence, Recall$_v@N$ calculates the number of hit tags in top-$N$ recommendation list among Ground Truth of item $v$ and Recall@N averages Recall$_v@N$ for all items in the test set.

Normalized Discounted Cumulative Gain (NDCG) assigns different importance to different ranks, which is calculated as:

$$\text{NDCG}_v@N = \frac{\sum_{i=1}^{N} 2^{r_i} - 1}{\log_2(i + 1)} \left( \frac{|y_v|}{\log_2} \right),$$  \hspace{1cm} (24)

where $r_i$ represents the relevance degree of the item at position $i$ and it is assigned as binary (i.e., 0 or 1) in implicit recommendation scenario. The numerator $\log_2(i + 1)$ calculates the discounted cumulative gain (DCG) score, which would increase according to its ranking position when the item in the top-$N$ list hits the Ground Truth $y_v$. Here, the denominator $\frac{|y_v|}{\log_2}$ is to normalize NDCG$_v@N$ to be in the range $[0, 1]$.

Mean reciprocal rank (MRR) measures where the first correctly predicted tag in the recommendation list appears, which calculates as:

$$\text{MRR}_v@N = \left\{ \begin{array}{ll} \frac{1}{r(T, y_v)}, & \text{if } \exists t \in T \text{ such that } t \in y_v, \\ 0, & \text{otherwise} \end{array} \right.$$  \hspace{1cm} (25)

where $T$ is a set of recommended top-$N$ tags, $y_v$ is a set of tags that the item $v$ interacts with, and $r(T, y_v)$ denotes the position of the first correctly recommended tag in the recommendation list for item $v$. MRR@N calculates the average of MRR$_v@N$ for all items in the test set.

4.1.3 Baseline Methods. To evaluate the effectiveness of our model, we compare it with the following state-of-the-art methods for tag recommendations:

- **CF [20]** This is a matrix-factorization-based collaborative filtering method that factorizes the training matrix into two low-rank matrices and recovers the original matrix by the inner product of them. It only uses the item-tag matrix information.
- **BPR [33]** BPR is a generic optimization criterion that tries to classify the predicted difference between the positive pair and the negative pair. We model the two-way interactions between tags and items by factorizing item and tag embeddings to make recommendations.
- **GNN-PTR [6]** It is a graph neural networks-boosted personalized tag recommendation model, which integrates the graph neural networks into the pairwise interaction tensor factorization model. The modeling of users is excluded to make it applicable to our setting.
- **Bi-GRU+Att [14]** This is a content-based tag recommendation method, where deep learning methods are utilized for capturing semantic meanings in the text.
recurrent units (Bi-GRUs) with attention mechanisms are employed to encode text information into semantic vectors.

- **ITAG** [41] This content-based tag recommendation method takes tag correlation and content-tag overlapping modeling into consideration beyond capturing textual semantic embeddings using Recurrent Neural Networks.

- **CTR-SR** [46] It is a hybrid method that combines the item-tag matrix, item content information, and item social information into a unified framework through a hierarchical Bayesian model.

- **HAM-TR** [38] It models two important attentive aspects with a hierarchical attention model, which exploits two levels of attention to effectively aggregate different elements and different information of content information and collaborative information, respectively. To make it comparable to our object-oriented tag recommendation model, we eliminated the corresponding modeling of users and make a recommendation based on the similarity of items and tags.

Here, the first three baselines are collaborative-based methods using only item-tag interactions, where CF and BPR are matrix-based and GNN-PTR is graph-based. Bi-GRU+Att and ITAG are content-based methods that utilize item content to predict tag interactions of items. The last two baselines are hybrid methods, where HAM-TR exploits both collaborative information and item content information. CTR-SR further utilizes item social graph besides ratings and item contents for hybrid tag recommendations.

### 4.1.4 Parameter Settings

The validation set is utilized to select the best parameters. Empirically, we set batch size and embedding size both to be 64; other parameters are found by performing a grid search as follows: \( \lambda_C, \lambda_S \in \{0.1, 0.5, 1, 2, 10, 100\} \), learning rate \( \in \{0.01, 0.001, 0.0001, 0.0005, 0.00001\} \). The models for collaborative information, content, and social graph are pre-trained in a plain Mult-VAE, VAE, and VGAE manner to first learn initial starting points for the network weights. We set the learning rate of pre-trained VAE, pre-trained VGAE, and pre-trained Mult-VAE to be 0.001, empirically. However, the learning rates of our MA-CVAE are carefully tuned, which controls the balance of the learning speed of collaborative information, content information, and social information. By default, we set our learning rate to be 1e-5 and 5e-4 for Multi-VAE and VGAE, respectively, for four datasets. And the learning rate of content VAE model are set to be 1e-4, 1e-3, 5e-4, and 5e-4 for citeulike-a, citeulike-t, ml-20m, and ml-25m datasets. The overall architecture for the Mult-VAE is \([J \rightarrow 600 \rightarrow 64 \rightarrow 600 \rightarrow J]\) with \(J\) tags. Both the inference network and the generation network are chosen to be a two-hidden-layer network architecture (\([64 \rightarrow 64]\)) for content VAE network. For VGAE, we set the neighbors to sample for each node in VGAE to be \([20, 20]\) for citeulike-a dataset and \([10, 10]\) for citeulike-t, ml-20m, and ml-25m datasets with a two-layer neighborhood aggregation. Moreover, we tune the parameters \(\lambda_C\) and \(\lambda_S\), which control the balance of content information and social information. We set \(\lambda_C\) and \(\lambda_S\) both to be 10 by default and further discussion of these two parameters is included in the following section of sensitivity to parameters. The experiments are conducted on a device with GeForce GTX 1080 Ti and 11G GPU memory.

### 4.2 Overall Comparison

The overall performance of our proposed MA-CVAE and the state-of-the-art baselines for tag recommendations are summarized in Table 3, where the upper part is various baselines and the bottom part is our proposed method with ablations. MA-CVAE\(_{w/o}\) only conducts Mult-VAE on collaborative information, MA-CVAE\(_{content}\) and MA-CVAE\(_{social}\) perform with augmented item content...
Table 3. Overall Performance Comparisons between the Proposed MA-CVAE and Various Baselines on Recall@20, NDCG@20, and MRR@20

|                  | ml-20m Recall | ml-20m NDCG | ml-20m MRR | ml-25m Recall | ml-25m NDCG | ml-25m MRR | citeulike-a Recall | citeulike-a NDCG | citeulike-a MRR | citeulike-t Recall | citeulike-t NDCG | citeulike-t MRR |
|------------------|---------------|-------------|------------|---------------|-------------|------------|-------------------|----------------|----------------|-------------------|----------------|----------------|
| CF               | 0.1504        | 0.0861      | 0.1714     | 0.0781        | 0.0424      | 0.0739     | 0.0913            | 0.0441         | 0.0831         | 0.0812            | 0.0365         | 0.0576         |
| BPR              | 0.3461        | 0.2084      | 0.2364     | 0.0924        | 0.1002      | 0.0951     | 0.1275            | 0.0634         | 0.1081         | 0.1172            | 0.0555         | 0.0819         |
| GNN-PTR          | 0.3554        | 0.2094      | 0.2214     | 0.0853        | 0.1152      | 0.1052     | 0.1240            | 0.0594         | 0.0989         | 0.1196            | 0.0542         | 0.0763         |
| Bi-GRU-Att       | 0.2521        | 0.0199      | 0.1632     | 0.0773        | 0.0070      | 0.0708     | 0.2089            | 0.0183         | 0.1711         | 0.2006            | 0.0140         | 0.1437         |
| ITAG             | 0.2930        | 0.0852      | 0.0627     | 0.0896        | 0.0423      | 0.0395     | 0.1402            | 0.0397         | 0.0347         | 0.1282            | 0.0410         | 0.0377         |
| CTR-SR           | 0.1280        | 0.0601      | 0.0965     | 0.0462        | 0.0160      | 0.0393     | 0.2052            | 0.0847         | 0.1189         | 0.1167            | 0.0479         | 0.0537         |
| HAM-TR           | 0.2479        | 0.0705      | 0.0496     | 0.1436        | 0.0711      | 0.1152     | 0.2105            | 0.0652         | 0.0498         | 0.1691            | 0.0484         | 0.0396         |
| **p-value**      | **8.49E-1**   | **7.34E-4** | **1.49E-3**| **1.91E-3**   | **5.6E-1**  | **1.46E-5**| **9.62E-5**       | **1.23E-5**    | **8.76E-6**    | **9.37E-6**      | **4.8E-7**     | **4.6E-6**     |
| p-value          | 0.4956        | 0.2371      | 0.4202     | 0.1066        | 0.0519      | 0.1052     | 0.2558            | 0.1150         | 0.2333         | 0.2351            | 0.1084         | 0.1789         |
| MA-CVAE-content  | 0.5559        | 0.2721      | 0.4813     | 0.1297        | 0.0631      | 0.1270     | 0.3153            | 0.1517         | 0.3168         | 0.3260            | 0.1622         | 0.2716         |
| MA-CVAE-social   | 0.5555        | 0.2730      | 0.4829     | 0.1306        | 0.0644      | 0.1286     | 0.3266            | 0.1575         | 0.3273         | 0.3363            | 0.1686         | 0.2841         |
| MA-CVAE          | **0.5658**    | **0.2822**  | **0.5000** | **0.1453**    | **0.0715**  | **0.1406** | **0.3277**        | **0.1596**     | **0.3329**     | **0.3369**        | **0.1683**     | **0.2818**     |
| **run_time**     | **5.9min**    | **40.7min** | **10.0min**| **10.15min**  | **9.75min** | **9.46E-3**| **9.8E-3**        | **9.5E-3**     | **9.8E-3**     | **9.6E-3**        | **9.5E-3**     | **9.8E-3**     |

information and social information, respectively. Each metric is averaged across all test users, and we report top-20 evaluations.

First, for ml-20m, citeulike-a, and citeulike-t datasets, by comparing collaborative-based methods (i.e., CF, BPR, and GNN-PTR) and content-based methods (i.e., Bi-GRU-Att and ITAG), we can see that content-based methods perform better on sparse datasets like citeulike. However, on more dense datasets like ml-20m, collaborative-based methods perform much better, which demonstrates the effectiveness of collaborative filtering by using co-occurrence information of collaborative-based methods. For the hybrid method CTR-SR, it achieves satisfactory results on the citeulike-a dataset, while on the other two datasets whose content features are high-dimensional and sparse, it cannot perform well. This is because of the weakness of LDA in modeling high-dimensional and sparse features. However, the other hybrid method, HAM-TR, which fuses collaborative and content information with weighted average using latent factor methods for modeling both information, achieves somewhat satisfactory results.

For ml-25m, since item-tag interactions in this dataset are annotated by users and not filtered according to the content similarity between items and tags, content-based methods (Bi-GRU-Att and ITAG) do not perform well on ml-25m. This is due to the low correlation between the tag and item content, where noise contained in item contents is modeled. Collaborative filtering-based methods and the hybrid model HAM-TR can achieve better performance by using collaborative rating data. Among them, GNN-PTR outperforms other matrix factorization-based methods by efficient graph convolution. The NDCG score of GNN-PTR is superior to our method, which may be due to its effectiveness on the aggregation of higher-order neighbors on the item-tag bipartite graph to achieve a good sort of recalled items. In addition, considering that: (1) the main difference between ml-20m and ml-25m after preprocessing exists in the movie-tag relevance degrees. Specifically, ml-20m contains more relevant item-tag pairs where all item-tag pairs have the movie-tag relevance scores provided by Reference [42] bigger than 0.75, while ml-25m has more noisy and low-relevance item-tag pairs where a vast majority of pairs have relevance data less than 0.6; (2) our model achieves better NDCG performance on the ml-20m dataset compared with GNN-PTR. We analyze that the main reason for having a relatively poor NDCG performance on the ml-25m dataset compared with GNN-PTR may be due to the fact that ml-25m dataset contains more item-tag pairs with lower relevance. Although our method can filter out some noise and learn recommendation-relevant features according to the dual constraint of collaborative information, MA-CVAE is still affected to a certain extent, since more noisy or mismatched item-tag pairs would make it difficult for our hybrid model to learn co-occurrence information of item contents and tags, which leads to a decrease in NDCG.
Overall, our proposed MA-CVAE achieves significantly competitive results among all baselines. From the p-value of t-tests on Recall, NDCG, and MRR, we could see that all the p-values are smaller than 0.05 (except for NDCG on ml-25m) for all tests, which demonstrates the scores obtained by the proposed model are significantly bigger than that of the other baselines. Compared to collaborative-based methods, MA-CVAE\(_{w/o}\) with only collaborative information outperforms the strong GNN-PTR by a large margin (except for NDCG on ml-25m) mainly due to the modeling of collaborative information via a deep generative model and applying a multinomial likelihood for the implicit interaction data, which has been proved to be good for top-N recommendations. For content-based methods, MA-CVAE\(_{content}\) performs better than them due to the effective modeling of collaborative information and content information. MA-CVAE\(_{content}\) also outperforms the hybrid method HAM-TR on three datasets by modeling content with deep generative models and tight coupling of item content information and collaborative information. CTR-SR, which makes use of item content and social graph information, applies linear methods to model the item auxiliary information. Therefore, it achieves inferior performance compared to our MA-CVAE method. We further attribute the performance improvements to the following two reasons: (1) By defining a probabilistic generative process, the tight coupling of collaborative information and item auxiliary information is achieved by MSE losses between the item latent embeddings and multiple auxiliary embeddings. In this way, the two types of information could be maximally utilized in a tuning way. (2) The modeling of collaborative information and multi-auxiliary information is conducted with deep generative models, which have good representational abilities on sparse data. Moreover, the variational-based methods could be robust to noises.

As an ablation study of our method, MA-CVAE\(_{w/o}\) with only collaborative information achieves inferior results compared to MA-CVAE\(_{content}\) and MA-CVAE\(_{social}\) with item content information and social information, respectively. It demonstrates that different item auxiliary information all play vital roles in tag recommendations, especially the social information on citeulike-t dataset. Moreover, by defining a generative process, we fuse multiple auxiliary information via a product-of-experts module and tightly couple the collaborative information and multi-auxiliary information, which achieves the best performance. The running time on the training set of our proposed MA-CVAE has been reported at the bottom of the table, which could demonstrate the time efficiency of our algorithm and applicability to larger-scale datasets.

4.3 Recommendation for Cold-start Items

We explore the effectiveness of our proposed method when recommending tags for new items, which do not exist in the training set. Collaborative-based (i.e., CF, BPR, and GNN-PTR) and hybrid methods (i.e., CTR-SR and HAM-TR) in our baselines are dependent on interactions when making predictions for items, and thus they cannot be applied to item cold-start scenarios. However, the spring-up of new items in tag recommendation scenarios is a common problem that needs to be solved. For our proposed MA-CVAE, by modeling item auxiliary information with deep generative models and constraining the generation of implicit feedback using different auxiliary embeddings, the new items could be recommended using item multi-auxiliary information. Here, the results between content-based baselines and our proposed MA-CVAE are listed in Table 4.

Results show that Bi-GRU+Att could make good recommendations on citeulike datasets whose vocabulary sizes are smaller, and however, it cannot perform well on the ml-20m dataset with a straightforward multi-class classification setting. ITAG could achieve good results on ml-20m due to the modeling of tag correlations with GRU layers, while it performs badly on citeulike datasets. They both cannot perform well on the ml-25m dataset, which has a lower relevance between item contents and tags. However, our proposed method performs much better than content-based
Table 4. Performance Comparisons between the Proposed MA-CVAE and Other Baselines on Cold-start Items on Recall@20, NDCG@20, and MRR@20

| Method                  | ml-20m      | ml-25m      | citeulike-a | citeulike-t |
|-------------------------|-------------|-------------|-------------|-------------|
|                         | Recall | NDCG | MRR | Recall | NDCG | MRR | Recall | NDCG | MRR | Recall | NDCG | MRR |
| Bi-GRU+Att              | 0.2023 | 0.0660 | 0.2425 | 0.0490 | 0.0067 | 0.0433 | 0.2113 | 0.0922 | 0.2885 | 0.1859 | 0.0504 | 0.2334 |
| ITAG                    | 0.2930 | 0.0854 | 0.0601 | 0.0810 | 0.0412 | 0.0425 | 0.1259 | 0.0363 | 0.0324 | 0.1284 | 0.0407 | 0.0395 |
| MA-CVAE_content         | 0.2647 | 0.1167 | 0.5940 | 0.1315 | 0.0546 | 0.1834 | 0.2605 | 0.1176 | 0.7566 | 0.2640 | 0.1233 | 0.5317 |
| MA-CVAE_social          | 0.2840 | 0.1247 | 0.6029 | 0.1435 | 0.0633 | 0.2203 | 0.2734 | 0.1228 | 0.7678 | 0.2952 | 0.1402 | 0.5829 |
| MA-CVAE                 | 0.2983 | 0.1313 | 0.6153 | 0.1445 | 0.0639 | 0.2292 | 0.2862 | 0.1291 | 0.7971 | 0.3085 | 0.1477 | 0.6077 |

Fig. 3. Sensitivity to parameters. (a), (d) The effect of $\lambda_C$ and $\lambda_S$ on ml-20m. (b), (e) The effect of $\lambda_C$ and $\lambda_S$ on citeulike-a. (c), (f) The effect of $\lambda_C$ and $\lambda_S$ on citeulike-t.

methods on four datasets, which illustrates the effectiveness of our method to recommend new items using item multi-auxiliary embeddings as surrogates for the item latent embeddings. This is achieved by constraining the closeness of the item latent embeddings and multi-auxiliary embeddings by MSE losses and additional generation losses. Moreover, through the ablation study in the bottom part of the table, we can observe that different item auxiliary information all play an important role in tag recommendations for new items.

4.4 Further Discussion on MA-CVAE

4.4.1 Sensitive to Parameters. The parameters $\lambda_C$ and $\lambda_S$ influence the balance between the utilization of content and the social graph of item multiple auxiliary information. We set $\lambda_C \in \{0.1, 0.5, 1, 2, 10, 1000\}$ when fixing $\lambda_S$ to be 10, and set $\lambda_S \in \{0.1, 0.5, 1, 2, 10, 100\}$ when fixing $\lambda_C$ to be 10. From Figure 3, we have the following observations: (1) For both $\lambda_C$ and $\lambda_S$, the performance increases first and begins to decrease at some point, which demonstrates that when fixing the utilization of one auxiliary information, further exploiting another auxiliary information could improve the performance by injecting more information. However, if one kind of auxiliary
information overweights the multiple information, the performance decreases much, because the
other information could not be utilized well and the facilitation effect between each piece of in-
formation is reduced. (2) For citeulike datasets, the impact of performance by varying $\lambda_S$ is larger
than varying $\lambda_C$ and vice versa on ml-20m. It demonstrates that different content and social infor-
mation have different importance degrees on different datasets and, therefore, it is useful to choose
appropriate $\lambda_C$ and $\lambda_S$.

4.4.2 Interpretability. To further intuitively analyze the performance of our proposed model,
we show several visualization results of MA-CVAE in this section. These results are chosen from
the test set of the citeulike-a dataset, where the items are articles. We use “true tags” to indicate
the tags that the user assigns to the article, “pred tags” to indicate the tags recommended to the
user, ”Hit C” to indicate the partial content of the article itself, and ”Hit S” to indicate the partial
content of the adjacency articles in the item social graph. Among them, the first two items are
existing items in the training set, and the last one is a new item.

From Table 5, we can see that our proposed method can recommend suitable tags for items with
a good hit ratio of our predicted tags on true tags. Moreover, MA-CVAE could recommend some ap-
propriate tags that are not included in the true tags, such as “ppi (protein-protein-interaction)”
for Article I. Furthermore, by observing ”Hit C” and ”Hit S,” which represent the partial content
of the content itself and the content of its adjacent neighborhood items, we can figure out that
using item content only cannot always hit the right tag. However, through properly modeling the
content of the social network, the content of neighborhoods could be aggregated, which con-
tributes to making much better recommendations. For example, the tag “graph” for Article I needs
to sufficiently model item social network information to be properly recommended, which is per-
formed by a variational graph auto-encoder model in our proposed MA-CVAE. However, for Arti-
icle IV, which is a new item, we can see that our method can still recommend suitable tags with a
high hit ratio, and the item social network information (i.e., citations) performs a significant role
in recommendations.

5 CONCLUSIONS AND FUTURE WORK

In this article, we seamlessly integrate the collaborative information and item multi-auxiliary in-
formation by defining a generative process parameterized by deep variational auto-encoders. The
latent item embedding that contains collaborative information and multiple auxiliary embeddings
are tightly coupled by constraining their closeness with MSE losses. Moreover, new items could be
recommended by constraining the generation of implicit interactions through the item decoder by
multiple auxiliary embeddings in the training phase, as well as designing an inductive variational
graph auto-encoder to enable the inference of new items. Extensive experiments on real-world
datasets have demonstrated the effectiveness of MA-CVAE on both existing and new items.

In future work, we would like to deepen and widen our proposed MA-CVAE from the follow-
ing aspects: (1) disentangled settings, where personality and matching are considered as two as-
pects of tag recommendations. Specifically, personality represents taggers’ personal preference
among tags, while matching denotes the representative and properly matched tags of items.
The disentanglement for the two aspects should be adaptively learned for users, since demands
that whether personality is more important or not is different among users. (2) Considering the
temporal characteristics of item-tag interaction where spatio-temporal methods [5, 51] could be
leveraged. (3) Considering that the feature dimensions are now too high, can the dimensionality re-
duction process also be considered as an optimization process? Instead of just using NN to reduce
dimensionality [26].
In apparently scale-free protein-protein networks, and hence have substantially expanded our knowledge on the protein interaction space or "interactome" networks, most proteins interact with few partners.

In apparently scale-free protein-protein interaction networks, or "interactome" networks, most proteins interact with few partners.

Evidence for dynamically organized modularity in the yeast protein-protein interaction network has a special role to play in understanding interactions that are of central importance in postgenomic molecular biology.

## Article I

| Title | Evidence for dynamically organized modularity in the yeast protein-protein interaction network |
|-------|--------------------------------------------------------------------------------------------------|
| Pred tags | ppi, topology, pin, protein_interaction, bio, interactome, modules, interaction_network, genetic, biological_networks, graph |
| True tags | network_analysis, protein_interaction, pin, graph, genome, node, interactome, biological_networks, topology, interaction_network |

* #topology

  Hit C A link between the potential scale-free topology of interactome networks and genetic robustness seems to exist

  Hit S Ordinarily, the connection topology is assumed to be either completely regular or completely random

* #interactome

  Hit C A link between the potential scale-free topology of interactome networks and genetic robustness seems to exist

  Hit S and hence have substantially expanded our knowledge on the protein interaction space or interactome of the yeast

* #interaction-network

  Hit C In apparently scale-free protein-protein interaction networks, or "interactome" networks, most proteins interact with few partners

  Hit S Interaction networks are of central importance in postgenomic molecular biology

* #graph

  Hit C None

  Hit S We develop a search algorithm for topological motifs called graph alignment

  We introduce a graph generating model aimed at representing the evolution of protein interaction networks

## Article II

| Title | Situated Learning: Legitimate Peripheral Participation (Learning in Doing: Social, Cognitive and Computational Perspectives) |
|-------|--------------------------------------------------------------------------------------------------|
| Pred tags | learning, cultural_studies, discourse, practice, e-learning, field_prelim, technology, situated, equity, digital-literate |
| True tags | learning, cultural_studies, information_behavior, bitrex-import, novice, equity, digital-literate, collaboration, practice |

* #learning

  Hit C push forward the notion of situated learning—which learning is fundamentally a social process and not solely in the learner’s head

  Hit S its very first stage is bootstrapped in a social learning process under the strong influence of culture

* #practice

  Hit C moving toward full participation in the sociocultural practices of a community

  Hit S its social organization, and the details of its implementation in actual practice aboard large ships

* #cultural_studies

  Hit C None

  Hit S From the chief scientist of Xerox Corporation and a research specialist in cultural studies at [UC-Berkeley] comes a treatise

  This truly interdisciplinary text bridges art history, film, media, and cultural studies

* #cognition

  Hit C None

  Hit S We think the theory of distributed cognition has a special role to play in understanding interactions

  In this article we propose distributed cognition as a new foundation for human-computer interaction

## Article III

| Title | How to infer gene networks from expression profiles |
|-------|--------------------------------------------------------------------------------------------------|
| Pred tags | gene, network, sybilo, gene_expression, genetic, systems, causality, biology, bayesian, statistics, regulatory_network |
| True tags | dragan, primer, grant, systems, bayesian, sybilo, biological_networks, gene_expression, regulatory_networks |

* #gene_expression

  Hit C Gene expression data from microarrays are typically used for this purpose

  Hit S Large-scale gene expression profiling generates data sets that are rich in observed features but poor in numbers of observations

* #bayesian

  Hit C None

  Hit S This framework builds on the use of Bayesian networks for representing statistical dependencies. A Bayesian network is a |

  We start by showing how Bayesian networks can describe interactions between genes

* #systems

  Hit C None

  Hit S from such measurements, gene/protein interactions and key biological features of cellular systems |

  genetic control networks12 and many other self-organizing systems

## Article IV

| Title | Analysing biological pathways in genome-wide association studies |
|-------|--------------------------------------------------------------------------------------------------|
| Pred tags | genetics, gwas, genomics, association, methods, review, bioinformatics, statistics, gene, analysis, genetic, pathway_analysis |
| True tags | gwas, methods, biology, disease, mining, networks, genomics, statistics, genomics_analysis, pathway-analysis, genetic |

* #gwas

  Hit C None

  Hit S Genome-wide association studies (GWAS) have rapidly become a standard method for disease gene discovery

  This review is written from the viewpoint that findings from the GWAS provide preliminary genetic information that

* #biology

  Hit C None

  Hit S facilitates the usage and the analysis of biological networks in standard systems biology formats (SBML, SBGN, BioPAX)

  The molecular biology revolution led to an intense focus on the study of interactions between DNA, RNA and protein biosynthesis

* #statistics

  Hit C None

  Hit S We propose number of extensions to GSEA, including the use of different statistics to describe the association between

  We review and discuss three analytic methods to combine preliminary GWAS statistics to identify genes, alleles

* #genetic

  Hit C the power to uncover the relatively small effect sizes conferred by most genetic variants

  Hit S when used in conjunction with large databases of protein-protein, protein-DNA, and genetic interactions that
REFERENCES

[1] Bahare Askari, Jaroslaw Szlichta, and Amirali Salehi-Abari. 2021. Variational autoencoders for top-k recommendation with implicit feedback. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2061–2065.

[2] Fabiano M. Belém, Jussara M. Almeida, and Marcos A. Gonçalves. 2017. A survey on tag recommendation methods. *J. Assoc. Inf. Sci. Technol.* 68, 4 (2017), 830–844.

[3] David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. 2017. Variational inference: A review for statisticians. *J. Amer. Statist. Assoc.* 112, 518 (2017), 859–877.

[4] Jian Chen, Lan Du, and Lei Yao. 2022. Discriminative mixture variational autoencoder for semisupervised classification. *IEEE Trans. Cybern.* 52, 5 (2022), 3032–3046.

[5] Kaixuan Chen, Lina Yao, Dalin Zhang, Xianzhi Wang, Xiaojun Chang, and Feiping Nie. 2020. A semisupervised recurrent convolutional attention model for human activity recognition. *IEEE Trans. Neural Netw. Learn. Syst.* 31, 5 (2020), 1747–1756.

[6] Xuewen Chen, Yonghong Yu, Fengyixin Jiang, Li Zhang, Rong Gao, and Haiyan Gao. 2020. Graph neural networks boosted personalized tag recommendation algorithm. In *International Joint Conference on Neural Networks*. 1–8.

[7] Yifan Chen and Maarten de Rijke. 2018. A collective variational autoencoder for top-n recommendation with side information. In *Workshop on Deep Learning Recommender Systems*. 3–9.

[8] Yifan Chen, Yang Wang, Xiang Zhao, Hongzhi Yin, Ilya Markov, and Maarten De Rijke. 2020. Local variational feature-based similarity models for recommending top-N new items. *ACM Trans. Inf. Syst.* 38, 2 (2020), 1046–1102.

[9] Xiaomin Fang, Rong Pan, Guoxiang Cao, Xiuxiang He, and Wenyan Dai. 2015. Personalized tag recommendation through nonlinear tensor factorization using gaussian kernel. In *AAAI Conference on Artificial Intelligence*. 439–445.

[10] Flavio Figueiredo, Henrique Pinto, Fabiano Belém, Jussara Almeida, Marcos Gonçalves, David Fernandes, and Edleno Moura. 2013. Assessing the quality of textual features in social media. *Int. J. Inf. Process. Manag.* 49 (2013), 222–247.

[11] Ido Guy, Naama Zwerdling, Inbal Ronen, David Carmel, and Erel Uziel. 2010. Social media recommendation based on people and tags. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*. 194–201.

[12] William L. Hamilton, Rex Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. In *International Conference on Advances in Neural Information Processing Systems*. 1025–1035.

[13] Peng Hao, Guangquan Zhang, Luis Martinez, and Jie Lu. 2019. Regularizing knowledge transfer in recommendation with tag-inferred correlation. *IEEE Trans. Cybern.* 49, 1 (2019), 83–96.

[14] Hebatallah A. Mohamed Hassan, Giuseppe Sansonetti, Fabio Gasparetti, and Alessandro Micarelli. 2018. Semantic-based tag recommendation in scientific bookmarking systems. In *ACM Conference on Recommender Systems*. 465–469.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Identity mappings in deep residual networks. In *European Conference on Computer Vision*. 630–645.

[16] I. Higgins, L. Matthey, A. Pal, Christopher P. Burgess, Xavier Glorot, M. Botvinick, S. Mohamed, and Alexander Lerchner. 2017. Beta-VAE: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations*.

[17] Navid Khezrian, Jafar Habibi, and Issa Annamoradnejad. 2020. Tag recommendation for online Q&A communities based on BERT pre-training technique. *arXiv preprint arXiv:2010.04971* (2020).

[18] Diederik P. Kingma and Max Welling. 2014. Auto-encoding variational Bayes. In *International Conference on Learning Representations*.

[19] Thomas N. Kipf and Max Welling. 2016. Variational graph auto-encoders. In *International Conference on Advances in Neural Information Processing Systems Workshop*.

[20] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Comput.* 42, 8 (2009), 30–37.

[21] Wonsung Lee, Kyungwoo Song, and II-Chul Moon. 2017. Augmented variational autoencoders for collaborative filtering with auxiliary information. In *ACM Conference on Information Knowledge Management*. 1139–1148.

[22] Jia Li, Hua Xu, Xingwei He, Junhui Deng, and Xiaomin Sun. 2016. Tweet modeling with LSTM recurrent neural networks for hashtag recommendation. In *International Joint Conference on Neural Networks*. 1570–1577.

[23] Xin Li, Lei Guo, and Yihong Eric Zhao. 2008. Tag-based social interest discovery. In *International Conference on World Wide Web*. 675–684.

[24] Xiaopeng Li and James She. 2017. Collaborative variational autoencoder for recommender systems. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 305–314.
[25] Dawen Liang, Rahul G. Krishnan, Matthew D. Hoffman, and Tony Jebara. 2018. Variational autoencoders for collaborative filtering. In International Conference on World Wide Web. 689–698.

[26] Minnan Luo, Xiaojun Chang, Liqiang Nie, Yi Yang, Alexander G. Hauptmann, and Qinghua Zheng. 2018. An adaptive semisupervised feature analysis for video semantic recognition. IEEE Trans. Cybern. 48, 2 (2018), 648–660.

[27] Hao Ma, Wenbo Gong, José Miguel Hernández-Lobato, Noam Koenigstein, Sebastian Nowozin, and Cheng Zhang. 2018. Partial VAE for hybrid recommender system. In NIPS Workshop on Bayesian Deep Learning.

[28] Suman Kalyan Maity, Abhishek Panigrahi, Sayan Ghosh, Arundhati Banerjee, Pawan Goyal, and Animesh Mukherjee. 2019. DeepTagRec: A content-cum-user based tag recommendation framework for stack overflow. In European Conference on Information Retrieval. 125–131.

[29] Liqiang Nie, Yongqi Li, Fuli Feng, Xuemeng Song, Meng Wang, and Yinglong Wang. 2020. Large-scale question tagging via joint question-topic embedding learning. ACM Trans. Inf. Syst. 38, 2 (2020), 1046–8188.

[30] Liqiang Nie, Yi-Liang Zhao, Xiangyu Wang, Jialie Shen, and Tat-Seng Chua. 2014. Learning to recommend descriptive tags for questions in social forums. ACM Trans. Inf. Syst. 32, 1 (2014), 1046–8188.

[31] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. 2014. DeepWalk: Online learning of social representations. In ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 701–710.

[32] Daniel Ramage, David Hall, Ramesh Nallapati, and Christopher D. Manning. 2009. Labeled LDA: A supervised topic model for credit attribution in multi-labeled corpora. In Conference on Empirical Methods in Natural Language Processing. 248–256.

[33] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In Conference on Uncertainty in Artificial Intelligence. 452–461.

[34] Steffen Rendle and Lars Schmidt-Thieme. 2009. Factor models for tag recommendation in bibleon. In International Conference on ECML PKDD-Discovery Challenge. 235–242.

[35] Steffen Rendle and Lars Schmidt-Thieme. 2010. Pairwise interaction tensor factorization for personalized tag recommendation. In ACM International Conference on Web Search and Data Mining. 81–90.

[36] H. Kiyok, H. Sohn, Honglak Lee, and Xinchen Yan. 2015. Learning structured output representation using deep conditional generative models. In International Conference on Advances in Neural Information Processing Systems. 3483–3491.

[37] Yang Song, Lu Zhang, and C. Lee Giles. 2011. Automatic tag recommendation algorithms for social recommender systems. ACM Trans. Web 5, 1 (2011), 1–31.

[38] Jianshen Sun, Mingyue Zhu, Yuanchun Jiang, Yezheng Liu, and Le Wu. 2021. Hierarchical attention model for personalized tag recommendation. J. Assoc. Inf. Sci. Technol. 72, 2 (2021), 173–189.

[39] Masahiro Suzuki, Kotaro Nakayama, and Yutaka Matsuo. 2017. Joint multimodal learning with deep generative models. In International Conference on Learning Representations.

[40] Panagiotis Symeonidis, Alexandros Nanopoulos, and Yannis Manolopoulos. 2008. Tag recommendations based on tensor dimensionality reduction. In ACM Conference on Recommeder Systems. 43–50.

[41] Shijie Tang, Yuan Yao, Suwei Zhang, Feng Xu, Tianxiao Gu, Hanghang Tong, Xiaohui Yan, and Jian Lu. 2019. An integral tag recommendation model for textual content. In AAAI Conference on Artificial Intelligence. 5109–5116.

[42] Jesse Vig, Shilad Sen, and John Riedl. 2012. The tag genome: Encoding community knowledge to support novel interaction. ACM Trans. Interact. Intell. Syst. 2, 3 (2012), 1–44.

[43] Chong Wang and David M. Blei. 2011. Collaborative topic modeling for recommending scientific articles. In ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 448–456.

[44] Chaojie Wang, Bo Chen, Sucheng Xiao, Zhengjue Wang, Hao Zhang, Penghui Wang, Ning Han, and Mingyuan Zhou. 2021. Multimodal Weibull variational autoencoder for jointly modeling image-text data. IEEE Trans. Cybern. 52, 10 (2022), 11156–11171.

[45] Chao Wang, Hengshu Zhu, Peng Wang, Chen Zhu, Xi Zhang, Enhong Chen, and Hui Xiong. 2021. Personalized and explainable employee training course recommendations: A Bayesian variational approach. ACM Trans. Inf. Syst. 40, 4 (2021), 1046–8188.

[46] Hao Wang, Binyi Chen, and Wu-Jian Li. 2013. Collaborative topic regression with social regularization for tag recommendation. In International Joint Conference on Artificial Intelligence. 2719–2725.

[47] Hao Wang, Naiyan Wang, and Dit-Yan Yeung. 2015. Collaborative deep learning for recommender systems. In ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 1235–1244.

[48] Yong Wu, Shenggu Xi, Yuan Yao, Feng Xu, Hanghang Tong, and Jian Lu. 2018. Guiding supervised topic modeling for content based tag recommendation. Neurocomputing 314 (2018), 479–489.

[49] Ping Yang, Yan Song, and Yang Ji. 2015. Tag-based user interest discovery though keywords extraction in social network. In International Conference on Big Data Computing and Communications. 363–372.

[50] Jiakao Yuan, Yuanyuan Jin, Wenyan Liu, and Xiaoling Wang. 2019. Attention-based neural tag recommendation. In International Conference on Database Systems and Advanced Applications. 350–365.
[51] Dalin Zhang, Lina Yao, Kaixuan Chen, Sen Wang, Xiaojun Chang, and Yunhao Liu. 2020. Making sense of spatio-temporal preserving representations for EEG-based human intention recognition. *IEEE Trans. Cybern.*, 50, 7 (2020), 3033–3044.

[52] Haijun Zhang, Yanfang Sun, Mingbo Zhao, Tommy W. S. Chow, and Q. M. Jonathan Wu. 2020. Bridging user interest to item content for recommender systems: An optimization model. *IEEE Trans. Cybern.*, 50, 10 (2020), 4268–4280.

Received 28 April 2022; revised 5 December 2022; accepted 20 December 2022