Disentangled Representation Learning for Recommendation

Xin Wang, *Member, IEEE*, Hong Chen, Yuwei Zhou, Jianxin Ma, and Wenwu Zhu, *Fellow, IEEE*

Abstract—There exist complex interactions among a large number of latent factors behind the decision making processes of different individuals, which drive the various user behavior patterns in recommender systems. These factors hidden in those diverse behaviors demonstrate highly entangled patterns, covering from high-level user intentions to low-level individual preferences. Uncovering the disentanglement of these latent factors can benefit in enhanced robustness, interpretability, and controllability during representation learning for recommendation. However, the large degree of entanglement within latent factors poses great challenges for learning representations that disentangle them, and remains largely unexplored in literature. In this paper, we present the SEMantic MACRo-mIcro Disentangled Variational Auto-Encoder (SEM-MacridVAE) model for learning disentangled representations from user behaviors, taking item semantic information into account. Our SEM-MacridVAE model achieves macro disentanglement by inferring the high-level concepts associated with user intentions (e.g., to buy a pair of shoes or a laptop) through a prototype routing mechanism, as well as capturing the individual preferences with respect to different concepts separately. The micro disentanglement is guaranteed through a micro-disentanglement regularizer stemming from an information-theoretic interpretation of VAEs, which forces each dimension of the representations to independently reflect an isolated low-level factor (e.g., the size or the color of a shirt). The semantic information including visual and categorical signals extracted from candidate items is utilized to further boost the recommendation performance of the proposed SEM-MacridVAE model. Empirical experiments demonstrate that our proposed approach is able to achieve significant improvement over the state-of-the-art baselines. We also show that the learned representations are interpretable and controllable, capable of potentially leading to a new paradigm for recommendation where users have fine-grained control over some target aspects of the recommendation candidates.

Index Terms—Disentangled representation, recommendation

1 Introduction

Learning representations that can accurately reflect users' preference, based chiefly on user behavior, has been an important research focus for recommender systems since the advent of collaborative filtering [57]. Despite the huge success in the past decade, existing user behavior-based representation learning methods, including the deep structure approaches [10], [20], [38], [40], [65], [79], generally ignore the complex interactions among the latent factors behind the users' decision-making processes. These latent factors can be highly entangled, ranging from macro concepts that govern the intention of a user in a particular behavior, to micro individual preferences at a granular level when implementing a specific intention. Existing methods fail to disentangle these latent factors, resulting in the fact that those learned representations may mistakenly preserve the confounding of the highly entangled factors, which leads to non-robustness and low interpretability.

Disentangled representation learning, which targets at learning factorized representations capable of uncovering and disentangling the latent explanatory factors hidden in the observed data [3], has recently attracted lots of attentions in the research community. Disentangled representations benefits in more *robustness*, i.e., less sensitive to the misleading correlations discovered in the limited observed training data. Besides, the enhanced *interpretability* brought by disentangled representation also finds direct application in recommendation-related tasks, such as transparent advertising [41], customer-relationship management, and explainable recommendation [19], [78] etc. Moreover, the *controllability* exhibited by many disentangled representations [7], [8], [9], [13], [21], [31] can provide users with explicit controls over their desired recommendation results and offer them more interactive experience, which has great potential in driving a new paradigm for recommendation. However, the existing literature on disentangled representation learning mainly focuses on computer vision [9], [13], [14], [21], [22], [34], [37], [52], [80] rather than recommender systems.

User behaviors in recommender systems can be driven by both *macro* intentions and *micro* preferences, where macro intentions may involve high-level user intentions such as purchasing a pair of shoes or a laptop and micro preferences may refer to low-level user preferences such as the size or color of the shoes. Therefore, given that the above

- *Xin Wang, Hong Chen, Yuwei Zhou, and Wenwu Zhu are with the Department of Computer Science and Technology, Tsinghua University, Beijing 100190, China. E-mail: {xin_wang, wwzhu}@tsinghua.edu.cn, {h-chen20, zhouyw16}@mails.tsinghua.edu.cn.*
- *Jianxin Ma is with Alibaba Group, Hangzhou 310052, China. E-mail: majx13fromthu@gmail.com.*

Manuscript received 8 September 2021; revised 1 December 2021; accepted 13 February 2022. Date of publication 23 February 2022; date of current version 5 December 2022.
This work was supported in part by the National Key Research and Development Program of China under Grant 2020AAA0106300, and in part by the National Natural Science Foundation of China under Grant 62102222.
(Corresponding authors: Xin Wang and Wenwu Zhu.)
Recommended for acceptance by X. Li.
Digital Object Identifier no. 10.1109/TPAMI.2022.3153112

discrete relational user behavior data is essentially different from continuous image data, learning disentangled representations based on user behavior data for recommendation becomes largely unexplored and poses two challenges.

- The macro intentions and micro preferences co-exist in user behaviors, requiring the disentangled representation learning to separate these two levels of factors in a way that can preserve the hierarchical relations between high-level user intentions and low-level individual preferences under the intentions.
- The observed user behavior data, such as user-item ratings and user-item consumptions, are discrete and sparse in essence, differing themselves from continuous image data, which implies that the observed user behavior data is only associated with a very small number of entries in the high-dimensional representation space. This will be especially problematic when exploring the interpretability of a particular isolated dimension through varying the value of that dimension with other dimensions fixed.

To solve the challenges, we propose the SEMantic MACRo-mIcro Disentangled Variational Auto-Encoder (SEM-MacridVAE) model for learning disentangled representations based on user behavior with item semantic information being taken into consideration in this paper. Our proposed method explicitly models the separation of macro and micro factors when performing disentanglement at each level. In particular, macro disentanglement is achieved by discovering the high-level concepts associated with user intentions through a prototype routing mechanism, and separately capturing the individual preferences of a user with respect to different concepts. Micro disentanglement is strengthened through a micro-disentanglement regularizer derived from interpreting VAEs [33], [58] in terms of an information-theoretic perspective, which aims at forcing each individual dimension to indicate an independent micro factor. The semantic information including visual and categorical signals from items is employed to further boost the model performances of the proposed SEM-MacridVAE model. To handle the conflict between sparse discrete user behavior observations and dense continuous latent representations, we propose a beam-search strategy for investigating the interpretability of each isolated dimension through finding a smooth trajectory within different representations.

We conduct extensive empirical experiments to show that our SEM-MacridVAE model can achieve significant improvement over several state-of-the-art baselines. Experimental results also demonstrate that the learned disentangled representations from SEM-MacridVAE can be interpretable and controllable, which may potentially bring a promising new paradigm for recommendation where users are given fine-grained controls over target aspects of the recommendation candidates.

To summarize, this paper makes the following contributions.

- We study the problem of disentangled representation learning for discrete and sparse relational user behavior data in recommendation.
- We propose SEMantic MACRo-mIcro Disentangled Variational Auto-Encoder (SEM-MacridVAE) model, which is able to conduct both macro and micro disentanglement simultaneously in representation learning for user behavior.
- We utilize two types of semantic information, i.e., visual and categorical signals extracted from candidate items, to further boost the recommendation performance.
- We conduct extensive experiments on several real-world datasets to verify the advantages of our SEM-MacridVAE model in terms of recommendation accuracy, interpretability and controllability.

In particular, we would like to point out that compared to the MacridVAE model [51], our proposed SEM-Macrid model has the following expansions:

1) We propose a new SEM-MacridVAE which incorporates the visual semantic and categorical semantic signals extracted from items to boost the model performance.
2) Besides the public Movielens datasets adopted by MacridVAE, we additionally include four public available Amazon datasets to enrich our experiments.
3) We conduct more extensive experiments, including more recent comparative baselines, comprehensive ablation studies as well as visualizations.

The remainder of this paper is organized as follows. We review related works in Section 2 and present our proposed SEM-MacridVAE model in Section 3. Section 4 describes details about empirical evaluations over several real-world datasets in terms of various metrics. Last but not least, we conclude the whole paper and point out research directions deserving further investigations in Section 5.

2 Related Work

In this section, we review existing works on user behavior representation learning and disentangled representation learning.

2.1 Learning Representations From User Behavior

Learning from user behavior has been a central task of recommender systems since the advent of collaborative filtering [11], [23], [56], [57], [60]. Being able to predict user preferences through uncovering complex and unexpected patterns hidden in users' past behaviors without any domain knowledge, factorization based recommendation [36] has become one of the most popular methods in recommender systems. These factorization based collaborative filtering models factorize user and item information into latent representations to approximate user preferences and item attributes, either in a deterministic way [26], [35], [42], [46], [48], [56], [69], [70], [72] or a probabilistic manner [15], [27], [47], [55], [59], [68], [74]. In addition to early factorization based attempts, the more recent deep learning methods [10], [20], [38], [40], [65], [79] achieve massive improvement by learning highly informative representations. The entanglement of the latent factors behind user behavior, however, is mostly neglected by the black-box representation learning process adopted by the majority of the existing methods. To the extent of our knowledge, we

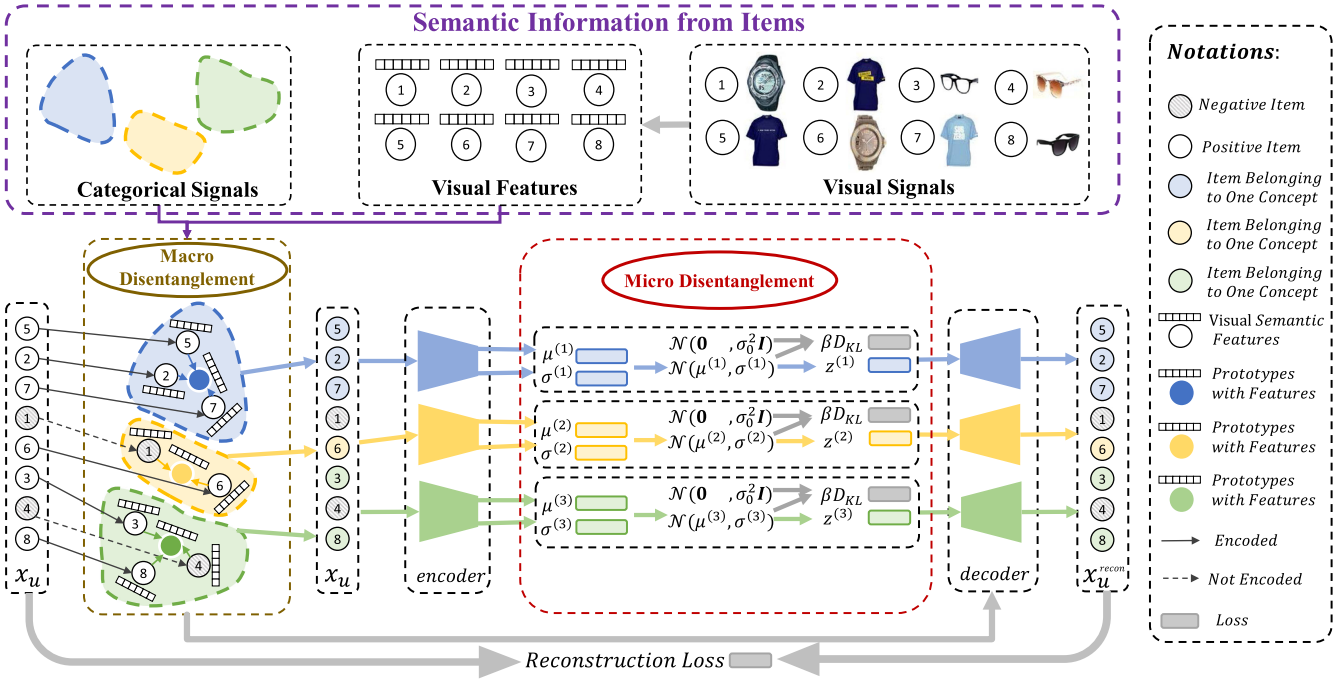


Fig. 1. The whole framework of our proposed SEM-MacridVAE model. The macro disentanglement is accomplished through learning a set of prototypes (macro concepts) based on which the user intention related with each item is inferred. Different colors (blue, yellow and green) in the figure indicate different macro concepts where each macro concept (one color) has one single independent prototype, encoder and decoder (with the same color). The micro disentanglement is achieved by capturing the preferences of the target user over different intentions separately, guaranteed by magnifying the KL divergence where a term penalizing the total correlation can be separated with a factor of β. Semantic information included categorical and visual signals extracted from candidate items is utilized to further improve model performance. In particular, visual signals are used to initialize the item factors and prototype representations, and categorical signals serve as the supervisions for learning macro concept C.

are the first to study disentangled representation learning on user behavior data.

2.2 Disentangled Representation Learning

Disentangled representation learning aims to identify and disentangle the underlying explanatory factors [3]. Being capable of producing robust, controllable, and explainable representations, disentangled representation learning has become one of the core problems in machine learning. In general, variational methods are widely applied for disentangled representation over images. β-VAE [21] demonstrates that disentanglement can emerge once the KL divergence term in the VAE [33] objective is aggressively penalized. In particular, Kingma and Welling [33] propose to utilize Bayesian posterior inference and variational estimation to learn the controllable factors hidden in the observed data. Higgins et al. [21] propose $\beta - VAE$ by setting a weight β for the KL divergence to improve representation disentanglement learned in the observed data while sacrificing mutual information between input data and latent representations. Later approaches separate the information bottleneck term [63], [64] and the total correlation term, and achieve a greater level of disentanglement [7], [8], [31]. Other works either design an attentive architecture to learn aspect matrix for word embeddings [17] or utilize methods based on triplets to learn aspect representations from sentences where each aspect has a separate encoder [25]. Though a few existing approaches [6], [9], [12], [13], [30] do notice that a dataset can contain samples from different concepts, i.e., follow a mixture distribution, their settings are fundamentally different from ours. To be specific, these existing approaches assume that each instance is from a concept, while we assume that each instance interacts with objects from different concepts. The majority of the existing efforts are from the field of computer vision [9], [13], [14], [21], [22], [34], [37], [52], [80]. Disentangled representation learning on relational data, such as graph-structured data, was not explored until recently [49], [66], [76]. This work focuses on disentangling user behavior from both the macro intention and micro preference in recommender systems.

3 Method

In this section, we describe our SEM-MacridVAE model for learning disentangled representations from user behaviors in detail, whose whole framework is demonstrated in Fig. 1.

3.1 Notations and Problem Formulation

A user behavior dataset $\mathcal{D}$ consists of the interactions between N users and M items. The interaction between the u^{th} user and the i^{th} item is denoted by $x_{u,i} \in \{0, 1\}$, where $x_{u,i} = 1$ indicates that user u explicitly adopts item i, whereas $x_{u,i} = 0$ means there is no recorded interaction between the two. For convenience, we use $\mathbf{x}_u = \{x_{u,i} : x_{u,i} = 1\}$ to represent the items adopted by user u. The goal is to learn user representations $\{\mathbf{z}_u\}_{u=1}^{N}$ that achieves both macro and micro disentanglement. We use $\boldsymbol{\theta}$ to denote the set that contains all the trainable parameters of our model.

3.1.1 Macro Disentanglement

Users may have very diverse interests, and interact with items that belong to many high-level concepts, e.g., product categories. We aim to achieve macro disentanglement, by

learning a factorized representation of user u, namely $\mathbf{z}_u = [\mathbf{z}_u^{(1)}; \mathbf{z}_u^{(2)}; \ldots; \mathbf{z}_u^{(K)}] \in \mathbb{R}^{d'}$, where $d' = Kd$, assuming that there are K high-level concepts. The k^{th} component $\mathbf{z}_u^{(k)} \in \mathbb{R}^d$ is for capturing the user's preference regarding the k^{th} concept. Additionally, we infer a set of one-hot vectors $\mathbf{C} = \{\mathbf{c}_i\}_{i=1}^M$ for the items, where $\mathbf{c}_i = [c_{i,1}; c_{i,2}; \ldots; c_{i,K}]$. If item i belongs to concept k, then $c_{i,k} = 1$ and $c_{i,k'} = 0$ for any $k' \neq k$. We infer $\{\mathbf{z}_u\}_{u=1}^N$ in an unsupervised way, and learn $\mathbf{C}$ in a supervised manner through the categorical signals of the semantic information.

3.1.2 Micro Disentanglement

High-level concepts correspond to the intentions of a user, e.g., to buy clothes or a cellphone. We are also interested in disentangling a user's preference at a more granular level regarding the various aspects of an item. For example, we would like the different dimensions of $\mathbf{z}_u^{(k)}$ to individually capture the user's preferred sizes, colors, etc., if concept k is clothing.

3.2 Model

We start by proposing a generative model that encourages macro disentanglement. For a user u, our generative model assumes that the observed data are generated from the following distribution:

$$p_{\boldsymbol{\theta}}(\mathbf{x}_u) = \mathbb{E}_{p_{\boldsymbol{\theta}}(\mathbf{C})}\left[\int p_{\boldsymbol{\theta}}(\mathbf{x}_u \mid \mathbf{z}_u, \mathbf{C}) p_{\boldsymbol{\theta}}(\mathbf{z}_u)\, d\mathbf{z}_u\right], \tag{1}$$

and

$$p_{\boldsymbol{\theta}}(\mathbf{x}_u \mid \mathbf{z}_u, \mathbf{C}) = \prod_{x_{u,i} \in \mathbf{x}_u} p_{\boldsymbol{\theta}}(x_{u,i} \mid \mathbf{z}_u, \mathbf{C}), \tag{2}$$

where the meanings of $\mathbf{x}_u, \mathbf{z}_u, \mathbf{C}$ are described in the previous subsection. We also assume that

$$p_{\boldsymbol{\theta}}(\mathbf{z}_u) = p_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{C})$$

in Equation (1), i.e., $\mathbf{z}_u$ and $\mathbf{C}$ are generated by two independent sources. Note that $\mathbf{c}_i = [c_{i,1}; c_{i,2}; \ldots; c_{i,K}]$ is one-hot, since we assume that item i belongs to exactly one concept. We also remark that

$$p_{\boldsymbol{\theta}}(x_{u,i} \mid \mathbf{z}_u, \mathbf{C}) = Z_u^{-1} \cdot \sum_{k=1}^K c_{i,k} \cdot g_{\boldsymbol{\theta}}^{(i)}(\mathbf{z}_u^{(k)})$$

is a categorical distribution over the M items, where

$$Z_u = \sum_{i=1}^M \sum_{k=1}^K c_{i,k} \cdot g_{\boldsymbol{\theta}}^{(i)}(\mathbf{z}_u^{(k)}),$$

and $g_{\boldsymbol{\theta}}^{(i)} : \mathbb{R}^d \to \mathbb{R}_+$ is a shallow neural network that estimates how much a user with a given preference is interested in item i. We use sampled softmax [29] to estimate Z_u based on a few sampled items when M is very large.

3.2.1 Macro Disentanglement

We assume above that the user representation $\mathbf{z}_u$ is sufficient for predicting how the user will interact with the items. And we further assume that using the kth component $\mathbf{z}_u^{(k)}$ alone is already sufficient if the prediction is about an item from concept k. This design explicitly encourages $\mathbf{z}_u^{(k)}$ to capture preference regarding only the kth concept, as long as the inferred concept assignment matrix $\mathbf{C}$ is meaningful.

We will describe later the implementation details of $p_{\boldsymbol{\theta}}(\mathbf{C})$, $p_{\boldsymbol{\theta}}(\mathbf{z}_u)$ and $g_{\boldsymbol{\theta}}^{(i)}(\mathbf{z}_u^{(k)})$. Nevertheless, we note that $p_{\boldsymbol{\theta}}(\mathbf{C})$ requires careful design to prevent mode collapse, i.e., the degenerate case where almost all items are assigned to a single concept.

3.2.2 Variational Inference

We follow the variational auto-encoder (VAE) paradigm [33], [58], and optimize θ by maximizing a lower bound of $\sum_u \ln p_{\boldsymbol{\theta}}(\mathbf{x}_u)$, where $\ln p_{\boldsymbol{\theta}}(\mathbf{x}_u)$ is bounded as follows:

$$\begin{aligned}
&\ln p_{\boldsymbol{\theta}}(\mathbf{x}_u) \\
\geq &\mathbb{E}_{p_{\boldsymbol{\theta}}(\mathbf{C})}\big[\mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C})}\big[\ln p_{\boldsymbol{\theta}}(\mathbf{x}_u \mid \mathbf{z}_u, \mathbf{C})\big] \\
&- D_{\mathrm{KL}}(q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C}) \| p_{\boldsymbol{\theta}}(\mathbf{z}_u))\big].
\end{aligned} \tag{3}$$

The proof is as follows.

Proof. Given the following equation,

$$q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u) = q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C}) p_{\boldsymbol{\theta}}(\mathbf{C}),$$

then we have the following inequality,

$$\begin{aligned}
&\ln p_{\boldsymbol{\theta}}(\mathbf{x}_u) \\
&= \mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)}[\ln p_{\boldsymbol{\theta}}(\mathbf{x}_u)] \\
&= \mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)}\left[\ln \frac{p_{\boldsymbol{\theta}}(\mathbf{x}_u, \mathbf{z}_u, \mathbf{C})}{p_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)}\right] \\
&= \mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)}\left[\ln \frac{q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)}{p_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)}\right] \\
&\quad + \mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)}\left[\ln \frac{p_{\boldsymbol{\theta}}(\mathbf{x}_u, \mathbf{z}_u, \mathbf{C})}{q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)}\right] \\
&= \mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)}\left[\ln \frac{q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)}{p_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)}\right] \\
&\quad + \mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)}[\ln p_{\boldsymbol{\theta}}(\mathbf{x}_u \mid \mathbf{z}_u, \mathbf{C})] \\
&\quad + \mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)}\left[\ln \frac{p_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C})}{q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)}\right] \\
&= D_{\mathrm{KL}}(q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u) \| p_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)) \\
&\quad + \mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)}[\ln p_{\boldsymbol{\theta}}(\mathbf{x}_u \mid \mathbf{z}_u, \mathbf{C})] \\
&\quad - D_{\mathrm{KL}}(q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u) \| p_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C})) \\
&\geq \mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u)}[\ln p_{\boldsymbol{\theta}}(\mathbf{x}_u \mid \mathbf{z}_u, \mathbf{C})] \\
&\quad - D_{\mathrm{KL}}(q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u) \| p_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C})) \\
&= \mathbb{E}_{p_{\boldsymbol{\theta}}(\mathbf{C})}\big[\mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C})}\big[\ln p_{\boldsymbol{\theta}}(\mathbf{x}_u \mid \mathbf{z}_u, \mathbf{C})\big] \\
&\quad - D_{\mathrm{KL}}(q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C}) \| p_{\boldsymbol{\theta}}(\mathbf{z}_u))\big].
\end{aligned}$$

Note that in the last line above, we have used

$$\begin{aligned}
&D_{\mathrm{KL}}(q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C} \mid \mathbf{x}_u) \| p_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{C})) \\
&= D_{\mathrm{KL}}(q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C}) p_\theta(\mathbf{C}) \| p_{\boldsymbol{\theta}}(\mathbf{z}_u) p_\theta(\mathbf{C})) \\
&= \mathbb{E}_{p_{\boldsymbol{\theta}}(\mathbf{C})}\big[D_{\mathrm{KL}}(q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C}) \| p_{\boldsymbol{\theta}}(\mathbf{z}_u))\big],
\end{aligned}$$

which completes the proof. □

Here we have introduced a variational distribution $q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C})$, whose implementation also encourages macro disentanglement and will be presented later. The two expectations, i.e., $\mathbb{E}_{p_{\boldsymbol{\theta}}(\mathbf{C})}[\cdot]$ and $\mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C})}[\cdot]$, are intractable, and are therefore estimated using the Gumbel-Softmax trick [28], [54] and the Gaussian re-parameterization trick [33], respectively. Once the training procedure is finished, $q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C})$ will be an approximation of the intractable posterior distribution $p_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C})$. We use the mode of $p_{\boldsymbol{\theta}}(\mathbf{C})$ as $\mathbf{C}$, and the mode of $q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C})$ as the representation of user u.

3.2.3 Micro Disentanglement

A natural strategy to encourage micro disentanglement is to force statistical independence between the dimensions, i.e., to force

$$q_{\boldsymbol{\theta}}(\mathbf{z}_u^{(k)} \mid \mathbf{C}) \approx \prod_{j=1}^{d} q_{\boldsymbol{\theta}}(z_{u,j}^{(k)} \mid \mathbf{C}),$$

so that each dimension describes an isolated factor, where

$$q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{C}) = \int q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C}) p_{\text{data}}(\mathbf{x}_u)\, d\mathbf{x}_u.$$

Fortunately, the Kullback–Leibler (KL) divergence term in the lower bound above does provide a way to encourage independence. Specifically, the KL term of our model can be rewritten as:

$$\begin{aligned} &\mathbb{E}_{p_{\text{data}}(\mathbf{x}_u)}\big[D_{\text{KL}}(q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C}) \| p_{\boldsymbol{\theta}}(\mathbf{z}_u))\big] \\ &= I_q(\mathbf{x}_u; \mathbf{z}_u) + D_{\text{KL}}(q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{C}) \| p_{\boldsymbol{\theta}}(\mathbf{z}_u)). \end{aligned} \tag{4}$$

The proof is as follows.

Proof.

$$\begin{aligned} &\mathbb{E}_{p_{\text{data}}(\mathbf{x}_u)}\big[D_{\text{KL}}(q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C}) \| p_{\boldsymbol{\theta}}(\mathbf{z}_u))\big] \\ &= \mathbb{E}_{p_{\text{data}}(\mathbf{x}_u)}\left[\mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C})}\left[\ln \frac{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C})}{p_{\boldsymbol{\theta}}(\mathbf{z}_u)}\right]\right] \\ &= \mathbb{E}_{p_{\text{data}}(\mathbf{x}_u)}\left[\mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C})}\left[\ln \frac{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C})}{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{C})} \frac{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{C})}{p_{\boldsymbol{\theta}}(\mathbf{z}_u)}\right]\right] \\ &= \mathbb{E}_{p_{\text{data}}(\mathbf{x}_u)}\left[\mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C})}\left[\ln \frac{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C})}{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{C})} + \ln \frac{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{C})}{p_{\boldsymbol{\theta}}(\mathbf{z}_u)}\right]\right] \\ &= \mathbb{E}_{p_{\text{data}}(\mathbf{x}_u)}\big[D_{\text{KL}}(q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C}) \| q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{C}))\big] \\ &\quad + \mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C}) p_{\text{data}}(\mathbf{x}_u)}\left[\ln \frac{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{C})}{p_{\boldsymbol{\theta}}(\mathbf{z}_u)}\right] \\ &= I_q(\mathbf{x}_u; \mathbf{z}_u) + \mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{C})}\left[\ln \frac{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{C})}{p_{\boldsymbol{\theta}}(\mathbf{z}_u)}\right] \\ &= I_q(\mathbf{x}_u; \mathbf{z}_u) + D_{\text{KL}}(q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{C}) \| p_{\boldsymbol{\theta}}(\mathbf{z}_u)). \end{aligned}$$

Note that $p_{\text{data}}(\mathbf{x}_u \mid \mathbf{C}) = p_{\text{data}}(\mathbf{x}_u)$, and the mutual information $I_q(\mathbf{x}_u; \mathbf{z}_u)$ is under the joint distribution

$$\begin{aligned} &q_{\boldsymbol{\theta}}(\mathbf{z}_u, \mathbf{x}_u \mid \mathbf{C}) \\ &= q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C}) p_{\text{data}}(\mathbf{x}_u \mid \mathbf{C}) \\ &= q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C}) p_{\text{data}}(\mathbf{x}_u), \end{aligned}$$

which completes the proof. □

Similar decomposition of the KL term has been noted for the original VAEs previously [1], [8], [31]. Penalizing the latter KL term would encourage independence between the dimensions, if we choose a prior that satisfies $p_{\boldsymbol{\theta}}(\mathbf{z}_u) = \prod_{j=1}^{d'} p_{\boldsymbol{\theta}}(z_{u,j})$. On the other hand, the former term $I_q(\mathbf{x}_u; \mathbf{z}_u)$ is the mutual information between $\mathbf{x}_u$ and $\mathbf{z}_u$ under $q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C}) \cdot p_{\text{data}}(\mathbf{x}_u)$. Penalizing $I_q(\mathbf{x}_u; \mathbf{z}_u)$ is equivalent to applying the information bottleneck principle [2], [63], which encourages $\mathbf{z}_u$ to ignore as much noise in the input as it can and to focus on merely the essential information. We therefore follow β-VAE [21], and strengthen these two regularization terms by a factor of $\beta \gg 1$, which brings us to the following training objective:

$$\begin{aligned} &\mathbb{E}_{p_{\boldsymbol{\theta}}(\mathbf{C})}\big[\mathbb{E}_{q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C})}\big[\ln p_{\boldsymbol{\theta}}(\mathbf{x}_u \mid \mathbf{z}_u, \mathbf{C})\big] \\ &- \beta \cdot D_{\text{KL}}(q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C}) \| p_{\boldsymbol{\theta}}(\mathbf{z}_u))\big]. \end{aligned} \tag{5}$$

3.3 Implementation

In this section, we describe the implementation of $p_{\boldsymbol{\theta}}(\mathbf{C})$, $p_{\boldsymbol{\theta}}(x_{u,i} \mid \mathbf{z}_u, \mathbf{C})$ (the decoder), $p_{\boldsymbol{\theta}}(\mathbf{z}_u)$ (the prior), $q_{\boldsymbol{\theta}}(\mathbf{z}_u \mid \mathbf{x}_u, \mathbf{C})$ (the encoder), and propose an efficient strategy to combat mode collapse. The parameters $\boldsymbol{\theta}$ of our implementation include: K concept prototypes $\{\mathbf{m}_k\}_{k=1}^{K} \in \mathbb{R}^{K \times d}$, M item representations $\{\mathbf{h}_i\}_{i=1}^{M} \in \mathbb{R}^{M \times d}$ used by the decoder, M context representations $\{\mathbf{t}_i\}_{i=1}^{M} \in \mathbb{R}^{M \times d}$ used by the encoder, and the parameters of a neural network $f_{\text{nn}} : \mathbb{R}^d \to \mathbb{R}^{2d}$. We optimize $\boldsymbol{\theta}$ to maximize the training objective (see Equation (6)) using Adam [32].

3.3.1 Prototype-Based Concept Assignment

A straightforward approach would be to assume $p_{\boldsymbol{\theta}}(\mathbf{C}) = \prod_{i=1}^{M} p(\mathbf{c}_i)$ and parameterize each categorical distribution $p(\mathbf{c}_i)$ with its own set of $K - 1$ parameters. This approach, however, would result in over-parameterization and low sample efficiency. We instead propose a prototype-based implementation. To be specific, we introduce K concept prototypes $\{\mathbf{m}_k\}_{k=1}^{K}$ and reuse the item representations $\{\mathbf{h}_i\}_{i=1}^{M}$ from the decoder. We then assume $\mathbf{c}_i$ is a one-hot vector drawn from the following categorical distribution $p_{\boldsymbol{\theta}}(\mathbf{c}_i)$:

$$\begin{aligned} &\mathbf{c}_i \sim \textsc{Categorical}\big(Softmax([s_{i,1}; s_{i,2}; \ldots; s_{i,K}])\big), \\ &s_{i,k} = \textsc{Cosine}(\mathbf{h}_i, \mathbf{m}_k)/\tau, \end{aligned} \tag{6}$$

where $\textsc{Cosine}(\mathbf{a}, \mathbf{b}) = \mathbf{a}^\top \mathbf{b} / (\|\mathbf{a}\|_2 \, \|\mathbf{b}\|_2)$ is the cosine similarity, and τ is a hyper-parameter that scales the similarity from $[-1, 1]$ to $[-\frac{1}{\tau}, \frac{1}{\tau}]$. We set $\tau = 0.1$ to obtain a more skewed distribution.

3.3.2 Preventing Mode Collapse

We use cosine similarity, instead of the inner product similarity adopted by most existing deep learning methods [20], [38], [40]. This choice is crucial for preventing mode collapse, which can be a severe issue with a mixture model [24], [73] such as ours if no special treatment is applied, especially when neural networks are involved [62]. In fact, with inner product, the majority of the items are highly likely to be assigned to a single concept $\mathbf{m}_{k'}$ that has an extremely large norm, i.e., $\|\mathbf{m}_{k'}\|_2 \to \infty$, even when the items $\{\mathbf{h}_i\}_{i=1}^{M}$ correctly form K clusters in the high-dimensional Euclidean

space. And we observe empirically that this phenomenon does occur frequently with inner product (see Figs. 3c and 4c). In contrast, cosine similarity avoids this degenerate case due to the normalization. Moreover, cosine similarity is related with the Euclidean distance on the unit hypersphere, and the Euclidean distance is a proper metric that is more suitable for inferring the cluster structure, compared to inner product.

3.3.3 Decoder

The decoder predicts which item out of the M ones is mostly likely to be clicked by a user, when given the user's representation $\mathbf{z}_u = [\mathbf{z}_u^{(1)}; \mathbf{z}_u^{(2)}; \ldots; \mathbf{z}_u^{(K)}]$ and the one-hot concept assignments $\{\mathbf{c}_i\}_{i=1}^M$. We assume that

$$p_{\boldsymbol{\theta}}(x_{u,i} \,|\, \mathbf{z}_u, \mathbf{C}) \propto \sum_{k=1}^{K} c_{i,k} \cdot g_{\boldsymbol{\theta}}^{(i)}(\mathbf{z}_u^{(k)}) \tag{7}$$

is a categorical distribution over the M items, and define

$$g_{\boldsymbol{\theta}}^{(i)}(\mathbf{z}_u^{(k)}) = \exp(\textsc{Cosine}(\mathbf{z}_u^{(k)}, \mathbf{h}_i)/\tau). \tag{8}$$

This design implies that $\{\mathbf{h}_i\}_{i=1}^M$ will be micro-disentangled if $\{\mathbf{z}_u^{(k)}\}_{u=1}^N$ is micro-disentangled, as the two's dimensions are aligned.

3.3.4 Prior & Encoder

The prior $p_{\boldsymbol{\theta}}(\mathbf{z}_u)$ needs to be factorized in order to achieve micro disentanglement. We therefore set $p_{\boldsymbol{\theta}}(\mathbf{z}_u)$ to $\mathcal{N}(\mathbf{0}, \sigma_0^2\mathbf{I})$. The encoder $q_{\boldsymbol{\theta}}(\mathbf{z}_u \,|\, \mathbf{x}_u, \mathbf{C})$ is for computing the representation of a user when given the user's behavior data $\mathbf{x}_u$, which approximates the posterior. The encoder maintains an additional set of context representations $\{\mathbf{t}_i\}_{i=1}^M$, rather than reusing the item representations $\{\mathbf{h}_i\}_{i=1}^M$ from the decoder, which is a common practice in the literature [40]. We assume that

$$q_{\boldsymbol{\theta}}(\mathbf{z}_u \,|\, \mathbf{x}_u, \mathbf{C}) = \prod_{k=1}^{K} q_{\boldsymbol{\theta}}(\mathbf{z}_u^{(k)} \,|\, \mathbf{x}_u, \mathbf{C}),$$

and represent each $q_{\boldsymbol{\theta}}(\mathbf{z}_u^{(k)} \,|\, \mathbf{x}_u, \mathbf{C})$ as a multivariate normal distribution with a diagonal covariance matrix $\mathcal{N}(\boldsymbol{\mu}_u^{(k)}, [\mathrm{diag}(\boldsymbol{\sigma}_u^{(k)})]^2)$, where the mean and the standard deviation are parameterized by a neural network $f_{\mathrm{nn}} : \mathbb{R}^d \to \mathbb{R}^{2d}$:

$$\begin{aligned}
(\mathbf{a}_u^{(k)}, \mathbf{b}_u^{(k)}) &= f_{\mathrm{nn}}\left(\frac{\sum_{i:x_{u,i}=+1} c_{i,k} \cdot \mathbf{t}_i}{\sqrt{\sum_{i:x_{u,i}=+1} c_{i,k}^2}}\right), \\
\boldsymbol{\mu}_u^{(k)} &= \frac{\mathbf{a}_u^{(k)}}{\|\mathbf{a}_u^{(k)}\|_2}, \\
\boldsymbol{\sigma}_u^{(k)} &\quad \sigma_0 \cdot \exp\left(-\frac{1}{2}\mathbf{b}_u^{(k)}\right).
\end{aligned} \tag{9}$$

The neural network $f_{\mathrm{nn}}(\cdot)$ captures nonlinearity, and is shared across the K components. We normalize the mean, so as to be consistent with the use of cosine similarity which projects the representations onto a unit hypersphere. Note that σ_0 should be set to a small value, e.g., around 0.1, since the learned representations are now normalized.

Algorithm 1. The Training Procedure. We add 10^{-8} to Prevent Division-by-Zero Wherever Appropriate.

input: $\mathbf{x}_u = \{x_{u,i} : \text{user } u \text{ clicks item } i, \text{i.e.}, x_{u,i} = 1\}$.
parameters:
  Concept prototypes $\mathbf{m}_k \in \mathbb{R}^d$ for $k = 1, 2, \ldots, K$;
  Item representations $\mathbf{h}_i \in \mathbb{R}^d$ for $i = 1, 2, \ldots, M$;
  Context representations $\mathbf{t}_i \in \mathbb{R}^d$ for $i = 1, 2, \ldots, M$;
  Parameters of a neural network $f_{\mathrm{nn}} : \mathbb{R}^d \to \mathbb{R}^{2d}$;
  Item categories: $\hat{\mathbf{c}}_i \in \mathbb{R}^K$ for $i = 1, 2, \ldots, M$;
    ▷ All these parameters are collectively denoted as $\boldsymbol{\theta}$.
function Initialization With Semantics
  $\{\mathbf{v}_i\}_{i=1}^M = AlexNet(\cdot)$.
  for $i = 1, 2, \ldots, M$ **do**
    $\mathbf{v}_i = \frac{1}{M}\sum_{i=1}^M \mathbf{v}_i$.
    $V = \frac{1}{M}\sum_{i=1}^M (\mathbf{v}_i - \mathbf{v}_i)(\mathbf{v}_i - \mathbf{v}_i)^T$.
    $P = Q^T[:d]$, where $V = Q\Lambda Q^T$.
    $\{\mathbf{h}_i\}_{i=1}^M = \{P\mathbf{v}_i\}_{i=1}^M$.
    $\{\mathbf{m}_k\}_{k=1}^K = Kmeans(\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_M)$.
  return $\{\mathbf{h}_i\}_{i=1}^M, \{\mathbf{m}_k\}_{k=1}^K$
function PrototypeClustering
  for $i = 1, 2, \ldots, M$ **do**
    $s_{i,k} \leftarrow \mathbf{h}_i^\top \mathbf{m}_k / (\tau \cdot \|\mathbf{h}_i\|_2 \cdot \|\mathbf{m}_k\|_2)$,
      where $k = 1, 2, \ldots, K$.
    $\mathbf{c}_i \sim$Gumbel-Softmax$([s_{i,1}; s_{i,2}; \ldots; s_{i,K}])$.
      ▷ At test time, $\mathbf{c}_i$ is set to the mode.
  return $\{\mathbf{c}_i\}_{i=1}^M$
function Encoder$\mathbf{x}_u$, $\{\mathbf{c}_i\}_{i=1}^M$
  for $k = 1, 2, \ldots, K$ **do**
    $(\mathbf{a}_k, \mathbf{b}_k) \leftarrow f_{\mathrm{nn}}\left(\frac{\sum_{i:x_{u,i}=+1} c_{i,k} \cdot \mathbf{t}_i}{\sqrt{\sum_{i:x_{u,i}=+1} c_{i,k}^2}}\right)$,
    $\boldsymbol{\mu}^{(k)} \quad \mathbf{a}_k / \|\mathbf{a}_k\|_2$,
    $\boldsymbol{\sigma}^{(k)} \quad \sigma_0 \cdot \exp\left(-\frac{1}{2}\mathbf{b}_k\right)$.
  $\boldsymbol{\mu}_u \quad [\boldsymbol{\mu}^{(1)}; \boldsymbol{\mu}^{(2)}; \ldots; \boldsymbol{\mu}^{(K)}]$,
  $\boldsymbol{\sigma}_u \quad [\boldsymbol{\sigma}^{(1)}; \boldsymbol{\sigma}^{(2)}; \ldots; \boldsymbol{\sigma}^{(K)}]$,
  $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$.
  $\mathbf{z}_u = \boldsymbol{\mu}_u + \boldsymbol{\epsilon} \circ \boldsymbol{\sigma}_u$.
    ▷ $\mathbf{z}_u$ is set to $\boldsymbol{\mu}_u$ at test time. "$\circ$" stands for element-wise multiplication.
  return$\mathbf{z}_u$, $D_{KL}(\mathcal{N}(\boldsymbol{\mu}_u, \mathrm{diag}(\boldsymbol{\sigma}_u)) \| \mathcal{N}(\mathbf{0}, \sigma_0 \cdot \mathbf{I}))$
function Decoder$\mathbf{z}_u$, $\{\mathbf{c}_i\}_{i=1}^M$
  $p_{u,i} \quad \sum_{k=1}^K c_{i,k} \cdot \exp(\mathbf{z}_u^{(k)\top} \mathbf{h}_i / (\tau \cdot \|\mathbf{z}_u^{(k)}\|_2 \cdot \|\mathbf{h}_i\|_2))$,
    where $i = 1, 2, \ldots, M$.
  $[p_{u,1}; p_{u,2}; \ldots; p_{u,M}]$
    $Softmax([\ln p_{u,1}; \ln p_{u,2}; \ldots; \ln p_{u,M}])$.
    ▷ We replace the $Softmax(\cdot)$ above with $Sampled-Softmax(\cdot)$, and compute $p_{u,i}$ only if $x_{u,i} = 1$ or item i is sampled, when M is very large.
  return $\{p_{u,i}\}_{i=1}^M$
$\{\mathbf{c}_i\}_{i=1}^M$ PrototypeClustering( ).
$\mathbf{z}_u$, D_{KL} Encoder($\mathbf{x}_u$, $\{\mathbf{c}_i\}_{i=1}^M$).
$\{p_{u,i}\}_{i=1}^M$ Decoder($\mathbf{z}_u$, $\{\mathbf{c}_i\}_{i=1}^M$).
$L = -\beta \cdot D_{\mathrm{KL}} + \sum_{i:x_{u,i}=1} \ln p_{u,i}$
  $+ \sum_{i=1}^M Cross\_Entropy(c_i, \hat{c}_i)$.
$\boldsymbol{\theta}$ Update $\boldsymbol{\theta}$ to maximize L, using the gradient $\nabla_{\boldsymbol{\theta}} L$.

3.4 Incorporating Semantic Information

In this section, we discuss the incorporation of semantic information extracted from items to further boost the model performance. Specifically, we consider two types of semantic information, i.e., visual signals and categorical signals.

Incorporating Visual Signals. The two key elements, i.e., concept prototypes $\{\mathbf{m}_k\}_{k=1}^{K}$ and item representations $\{\mathbf{h}_i\}_{i=1}^{M}$, in the prototype mechanism which has a crucial influence on both encoder and decoder, are so far initialized randomly without taking any semantic information from items into consideration.

Therefore, to further improve the model performances, we encode visual semantic information through a pre-trained AlexNet over the raw item image to obtain visual feature v_i for each item i. To match the dimension of v_i with that of the item embedding, we conduct Principal Component Analysis (PCA) on v_i. Then we initialize $\mathbf{h}_i$ with the low-dimensional visual feature conduct initialization for $\mathbf{m}_k$ by calculating the cluster center (obtained from K-means) of item representations belonging to concept k. Concretely, the visual features are obtained from the output of the last second fully-connected layers of the AlexNet, which has five convolutional layers followed by three fully-connected layers and is pre-trained on the ImageNet dataset with semantic categorical labels. Assuming the visual features output from AlexNet is denoted as $\{\mathbf{v}_i\}_{i=1}^{M} = AlexNet(\cdot)$, then the process of initializing $\{\mathbf{h}_i\}_{i=1}^{M}$ and $\{\mathbf{m}_k\}_{k=1}^{K}$ can be formulated as follows,

$$\begin{aligned}
\mathbf{v}_i &= \frac{1}{M}\sum_{i=1}^{M}\mathbf{v}_i,\\
V &= \frac{1}{M}\sum_{i=1}^{M}(\mathbf{v}_i-\mathbf{v}_i)(\mathbf{v}_i-\mathbf{v}_i)^T,\\
V &= Q\Lambda Q^T,\\
P &= Q^T[:d],\\
\{\mathbf{h}_i\}_{i=1}^{M} &= \{P\mathbf{v}_i\}_{i=1}^{M},\\
\{\mathbf{m}_k\}_{k=1}^{K} &= Kmeans(\mathbf{h}_1,\mathbf{h}_2,\ldots,\mathbf{h}_M),
\end{aligned} \tag{10}$$

where each column of Q represents an eigenvector of V and P contains d eigenvectors corresponding to the largest d eigenvalues.

Incorporating Categorical Signals. The number of macro concepts, i.e., K, are so far preset by human experience, followed by macro disentanglement in an unsupervised manner, which may run the risk of misalignment between the macro concepts and actual categories of items despite massive cost on trying and testing. Therefore, we utilize the categorical semantic information to achieve better macro disentanglement through supervised categorical signals in the following:

$$min\sum_{i=1}^{M} Cross\_Entropy(c_i,\hat{c_i}), \tag{11}$$

where $\hat{c_i}$ is one-hot vector that reflects the ground-truth category of the i^{th} item and $Cross\_Entropy(c_i,\hat{c_i})$ denotes the binary classification loss between the learned category and true category of the item i.

Empirical results in our experiments later show that by taking semantic information, i.e., visual and categorical signals, into account, the proposed SEM-MacridVAE model is able to outperform MacridVAE which initializes item representations and concept prototypes in a random manner. The incorporation of item semantic information is illustrated in the upper part of Fig. 1. Algorithm 1 presents the implementation details of the whole procedure.

3.5 User-Controllable Recommendation

The controllability enabled by the disentangled representations can bring a new paradigm for recommendation. It allows a user to interactively search for items that are similar to an initial item except for some controlled aspects, or to explicitly adjust the disentangled representation of his/her preference, learned by the system from his/her past behaviors, to actually match the current preference. Here, we formalize the task of user-controllable recommendation, and illustrate a possible solution.

3.5.1 Task Definition

Let $\mathbf{h}_* \in \mathbb{R}^d$ be the representation to be altered, which can be initialized as either an item representation or a component of a user representation. The task is to gradually alter its j^{th} dimension $h_{*,j}$, while retrieving items whose representations are similar to the altered representation. This task is nontrivial, since usually no item will have exactly the same representation as the altered one, especially when we want the transition to be smooth, monotonic, and thus human-understandable.

3.5.2 Solution

Here we illustrate our approach to this task. We first probe the suitable range (a,b) for $h_{*,j}$. Let us assume that prototype k_* is the prototype closest to $\mathbf{h}_*$. The range (a,b) is decided such that: prototype k_* remains the prototype closest to $\mathbf{h}_*$ if and only if $h_{*,j} \in (a,b)$. We can decide each endpoint of the range using binary search. We then divide the range (a,b) into B subranges, $a = a_0 < a_1 < a_2 \ldots < a_B = b$. We ensure that the subranges contain roughly the same number of items from concept k_* when dividing (a,b) . Finally, we aim to retrieve B items $\{i_t\}_{t=1}^{B} \in \{1,2,\ldots,M\}^B$ that belong to concept k_*, each from one of the B subranges, i.e., $h_{i_t,j} \in (a_{t-1},a_t]$. We thus decide the B items by maximizing

$$\sum_{1\le t\le B} e^{\frac{\text{Cosine}(\mathbf{h}_{i_t,-j},\mathbf{h}_{*,-j})}{\tau}} + \gamma\cdot\sum_{1\le t<t'\le B} e^{\frac{\text{Cosine}(\mathbf{h}_{i_t,-j},\mathbf{h}_{i_{t'},-j})}{\tau}}, \tag{12}$$

where $\mathbf{h}_{i,-j} = [h_{i,1};h_{i,2};\ldots;h_{i,j-1};h_{i,j+1};\ldots;h_{i,d}] \in \mathbb{R}^{d-1}$ and γ is a hyper-parameter. We approximately solve this maximization problem sequentially using beam search [45].

Intuitively, selecting items from the B subranges ensures that the items change monotonously in terms of the j^{th} dimension. On the other hand, the first term in the maximization problem forces the retrieved items to be similar with the initial item in terms of the dimensions other than j, while the second term encourages any two retrieved items to be similar in terms of the dimensions other than j.

TABLE 1
Statistics of the Datasets

Dataset	Users	Item	Ratings	Density
ML-latest-small	531	4807	46217	1.8106%
Movies&TV	15187	46234	623239	0.0888%
Musical Instruments	655	10377	15540	0.2286%
Home&Kitchen	6640	52900	154622	0.0440%
Clothing&shoes&Jewelry	9575	130742	197708	0.0158%

We highlight in Figs. 6, 7, and 8 some example cases that we found using this approach.

4 Empirical Experiments

In this section, we demonstrate that our learned disentangled representations are not only effective for recommendation, but also interpretable and controllable.

4.1 Experimental Setup

4.1.1 Datasets

We conduct extensive experiments on five public real-world datasets, including a MovieLens dataset (i.e., ML-latest-small) [16] and four Amazon product datasets [18] of different meta categories (i.e., Movies&TV, Musical Instruments, Home&Kitchen and Clothing&Shoes&Jewelry[1]). We follow MultiVAE [40], and binarize these five datasets by labeling ratings of four or higher as 1, and keeping users who have at least fifteen rating actions. Each item in MovieLens and Amazon datasets is associated with its corresponding image, and we utilize an AlexNet pre-trained on the ImageNet dataset to obtain 4096-dimension visual features which will then be transformed to a d-dimension latent factor to initialize item representation h of this item. Given that the ImageNet dataset contains semantic guidance for visual features through providing ground-truth labels for various types of images, in this way we are able to incorporate semantic information of each item into our learning process. All datasets are preprocessed using the script provided by MultiVAE. Half of the held-out users are used for validation, while the other half of the held-out users are for testing. Table 1 summarizes the basic statistics of the above datasets.

4.1.2 Baselines

We compare our approach with four baselines, MultiDAE [40], MultiVAE [40], MacridVAE [51] and DGCF [66]. MultiDAE [40] and MultiVAE [40] are the two state-of-the-art methods for collaborative filtering. In particular, MultiVAE is similar to β-VAE [21], and has a hyper-parameter β that controls the strength of disentanglement. However, MultiVAE does not learn disentangled representations, because it requires $\beta \ll 1$ to perform well. MacridVAE [51] can be treated as a variant of SEM-MacridVAE which conducts random initialization without considering any semantic information from items, and we compare it with the proposed SEM-MacridVAE model to further verify the improvement brought by incorporating item semantic information. Besides, DGCF [66] is chosen as the comparative baseline given that it is one of the most recent works focusing on disentangled collaborative filtering.

We would also like to point out that there are also several works related to disentangled representation for recommendation [39], [61], [67], [71], [77]. However, we find that some of these works require multimodal [67], [77] or heterogeneous [71] information as input, some [39] in essence can be regarded as a β-VAE model with varying β, and some other work [61] utilizes social network information for social recommendation. These works are orthogonal to our focus in this paper and therefore are not included for comparisons in the experiments.

4.1.3 Hyper-Parameters

We constrain the number of learnable parameters to be around $2Md$ for each method so as to ensure fair comparison, which is equivalent to using d-dimensional representations for the M items. Note that all the methods under investigation use two sets of item representations, and we do not constrain the dimension of user representations since they are not parameters. We treat K as a hyper-parameter to be tuned and do not directly set K to the ground truth when [67], evaluating its performance on recommendation tasks, so as to ensure a fair comparison with the baselines. We set $d = 200$ and fix τ to 0.1. The neural network $f_{\text{nn}}(\cdot)$ in our model is a multilayer perceptron (MLP), whose input and output are constrained to be d-dimensional and $2d$-dimensional, respectively. We use the tanh activation function. We apply dropout before every layers, except the last layer. The model is trained using Adam. We then tune the other hyper-parameters of both our approach's and our baselines' automatically using the TPE method [5] implemented by Hyepropt [4]. We let Hyperopt conduct 200 trials to search for the optimal hyper-parameter configuration for each method on the validation of each dataset. The hyper-parameter search space is specified as follows:

- The standard deviation of the prior $\sigma_0 \in [0.075, 0.5]$.
- The strength of micro disentanglement $\beta \in [0, 100]$.
- The number of macro factors $K \in \{1, 2, 3, \ldots, 20\}$.
- The learning rate $\in [10^{-8}, 1]$.
- L2 regularization $\in [10^{-12}, 1]$.
- Dropout rate $\in [0.05, 1]$.
- The number of hidden layers in a neural network $\in \{0, 1, 2, 3\}$.
- The number of neurons in a hidden layer $\in \{50, 100, 150, \ldots, 700\}$.

4.1.4 Number of Macro Factors

Our initial implementation adaptively adjusts the number of macro factors K during training. To be specific, we set K as a sufficiently large value at the beginning and shrink its value after every training epoch if the Jensen–Shannon (JS) divergence between $\{p_{i|k}\}_{i=1}^{M}$ and $\{p_{i|k'}\}_{i=1}^{M}$ for some $k \neq k'$ is negligible compared to a predefined threshold, where $p_{i|k} := p_{\boldsymbol{\theta}}(c_{i,k} = 1) / \sum_{i'} p_{\boldsymbol{\theta}}(c_{i',k} = 1)$. We, however, do not find this adaptive strategy to be significantly better than the naïve strategy that treats K as a hyper-parameter to be

1. http://jmcauley.ucsd.edu/data/amazon/links.html

TABLE 2
Results of Recommendation Performance, Where Bold Font Denotes the Winner

Dataset	Method	Metrics		
		NDCG@100	recall@20	recall@50
ML-latest -small	MultiDAE	0.31930(±0.02657)	0.27885(±0.03689)	0.37373(±0.03658)
	MultiVAE	0.33233(±0.03031)	0.28539(±0.03659)	0.38718(±0.03774)
	MacridVAE	0.34180(±0.03190)	0.29844(±0.03711)	0.38994(±0.03696)
	DGCF	0.34709(±0.02971)	0.29185(±0.03433)	0.39353(±0.03836)
	SEM-MacridVAE	**0.35129(±0.03204)**	**0.30293(±0.03591)**	**0.40026(±0.03724)**
Movies &TV	MultiDAE	0.09774(±0.00336)	0.08342(±0.00382)	0.13936(±0.00498)
	MultiVAE	0.09953(±0.00338)	0.08431(±0.00387)	0.14004(±0.00496)
	MacirdVAE	0.11619(±0.00365)	0.10397(±0.00426)	0.16011(±0.00527)
	DGCF	0.10060(±0.00328)	0.08541(±0.00380)	0.14666(±0.00498)
	SEM-MacridVAE	**0.11674(±0.00367)**	**0.10466(±0.00423)**	**0.16101(±0.00515)**
Musical Instruments	MultiDAE	0.04508(±0.01194)	0.03171(±0.01767)	0.09709(±0.03196)
	MultiVAE	0.04420(±0.01156)	0.03436(±0.01781)	0.09590(±0.03022)
	MacridVAE	0.05706(±0.01871)	0.04034(±0.01885)	0.09419(±0.03199)
	DGCF	0.06109(±0.01657)	0.08352(±0.03310)	0.11870(±0.03754)
	SEM-MacridVAE	**0.06450(±0.01415)**	**0.08479(±0.03132)**	**0.13436(±0.03829)**
Home &Kitchen	MultiDAE	0.03577(±0.00401)	0.03488(±0.00499)	0.06190(±0.00674)
	MultiVAE	0.03761(±0.00420)	0.03607(±0.00514)	0.06094(±0.00671)
	MacridVAE	0.04271(±0.00456)	0.03641(±0.00510)	0.06737(±0.00659)
	DGCF	0.04370(±0.00404)	0.03853(±0.00494)	0.07699(±0.00708)
	SEM-MacridVAE	**0.04463(±0.00434)**	**0.04669(±0.00557)**	**0.07913(±0.00727)**
Clothing &Shoes &Jewelry	MultiDAE	0.01123(±0.00213)	0.01156(±0.00297)	0.01725(±0.00353)
	MultiVAE	0.01107(±0.00182)	0.01278(±0.00296)	0.02507(±0.00432)
	MacridVAE	0.01785(±0.00265)	0.01540(±0.00313)	0.03009(±0.00458)
	DGCF	0.01833(±0.00296)	0.02293(±0.00462)	0.03691(±0.00558)
	SEM-MacridVAE	**0.01853(±0.00240)**	**0.02491(±0.00434)**	**0.03720(±0.00517)**

We note that all models are constrained to have around $2Md$ parameters, where M is the number of items and d is the dimension of each item representation. The experiments show our proposed SEM-MacridVAE model is able to beat all comparative baselines.

tuned by Hyperopt, since the adaptive strategy introduces extra computational cost as well as a new hyper-parameter.

4.1.5 Experimental Environment

We implement our model with Tensorflow, and conduct our experiments with:

- CPU: Intel(R) Xeon(R) CPU E5-2699 v4 @ 2.20GHz.
- RAM: DDR4 1TB.
- GPU: 8x GeForce GTX 1080 Ti.
- Operating system: Ubuntu 18.04 LTS.
- Software: Python 3.6; NumPy 1.15.4; SciPy 1.2.0; scikit-learn 0.20.0; TensorFlow 1.12.

4.2 Model Performance

We evaluate the performance of our approach on the task of collaborative filtering for implicit feedback datasets [23], one of the most common settings for recommendation. We follow the experiment protocol established by the previous work [40] strictly, as well as use the same preprocessing procedure and evaluation metrics. The results on the five datasets are shown in Table 2.

We observe that our SEM-MacridVAE model outperforms the baseline methods significantly in all but one case for all datasets. The improvement is likely due to two desirable properties of our approach. Firstly, macro disentanglement not only allows us to accurately represent the diverse interests of a user using the different components, but also alleviates data sparsity by allowing a rarely visited item to borrow information from other items of the same category, which is the motivation behind many hierarchical methods [50], [75]. Secondly, as we will later show in Section 4.4 that the dimensions of the representations learned by our approach are highly disentangled, i.e., independent, which should take credits from the micro disentanglement regularizer leading to more robust performances. This second property implies that our approach can be more robust to the scenario where multiple factors are co-influencing the data generating process, especially when there is a limited amount of available data [3]. For example, it would not overreact to the preference for bag size when making a prediction that is only related with the preference for bag color.

SEM-MacridVAE versus MacridVAE. The comparisons between SEM-MacridVAE with item semantic information and MacridVAE without semantic information in Table 2 further validate the benefit of considering semantics in boosting model performances. Indeed, incorporating semantic meanings has been regarded as one effective way to improve both model accuracy and explainability of machine learning algorithms in the community.

With Macro & Micro Disentanglement v.s. Without Macro & Micro Disentanglement. Ablation studies (*Without Macro* and *Without Micro*) in Table 3 confirm the benefit of conducting both *Macro* and *Micro* disentanglement when making recommendations.

With Visual & Categorical Signals v.s. Without Visual & Categorical Signals. Similarly, comparisons for *Without Visual*, *Without Categorical* and *SEM-MacridVAE (Full Model)* in

TABLE 3
Ablation Studies on Macro & Micro Disentanglement and Visual & Categorical Signals, Where Bold Font Denotes the Winner

Dataset	Method	Metrics		
		NDCG@100	recall@20	recall@50
ML-latest -small	Without Macro	0.34571(±0.03125)	0.29806(±0.03481)	0.38994(±0.03898)
	Without Micro	0.33872(±0.02977)	0.29140(±0.03482)	0.38253(±0.03828)
	Without Visual	0.34568(±0.03222)	0.29999(±0.03729)	0.38504(±0.03750)
	Without Categorical	0.34469(±0.03130)	0.30278(±0.03492)	0.39415(±0.03824)
	SEM-MacridVAE (Full Model)	**0.35129(±0.03204)**	**0.30293(±0.03591)**	**0.40026(±0.03724)**
Movies &TV	Without Macro	0.11309(±0.00359)	0.10294(±0.00422)	0.15935(±0.00521)
	Without Micro	0.11064(±0.00345)	0.10210(±0.00424)	0.16062(±0.00522)
	Without Visual	0.11524(±0.00365)	0.10360(±0.00417)	0.16025(±0.00521)
	Without Categorical	0.11559(±0.00365)	0.10336(±0.00424)	0.16012(±0.00515)
	SEM-MacridVAE (Full Model)	**0.11674(±0.00367)**	**0.10466(±0.00423)**	**0.16101(±0.00515)**
Musical Instruments	Without Macro	0.06389(±0.01775)	0.07111(±0.02798)	0.11726(±0.03532)
	Without Micro	0.06206(±0.01383)	0.07111(±0.02798)	0.13733(±0.03819)
	Without Visual	0.05801(±0.01923)	0.04496(±0.02238)	0.10265(±0.03253)
	Without Categorical	0.06355(±0.01426)	0.08265(±0.02881)	0.13656(±0.03621)
	SEM-MacridVAE (Full Model)	**0.06450(±0.01415)**	**0.08479(±0.03132)**	**0.13436(±0.03829)**
Home &Kitchen	Without Macro	0.04377(±0.00404)	0.04412(±0.00556)	0.07332(±0.00713)
	Without Micro	0.04365(±0.00429)	0.04243(±0.00535)	0.07166(±0.00690)
	Without Visual	0.04264(±0.00437)	0.04066(±0.00525)	0.07030(±0.00689)
	Without Categorical	0.04440(±0.00426)	0.04250(±0.00520)	0.07484(±0.00692)
	SEM-MacridVAE (Full Model)	**0.04463(±0.00434)**	**0.04669(±0.00557)**	**0.07913(±0.00727)**
Clothing &Shoes &Jewelry	Without Macro	0.01817(±0.00254)	0.02233(±0.00398)	0.03055(±0.00470)
	Without Micro	0.01805(±0.00257)	0.02219(±0.00398)	0.03660(±0.00511)
	Without Visual	0.01645(±0.00246)	0.01963(±0.00369)	0.03107(±0.00469)
	Without Categorical	0.01852(±0.00250)	0.02248(±0.00406)	0.03328(±0.00476)
	SEM-MacridVAE (Full Model)	**0.01853(±0.00240)**	**0.02491(±0.00434)**	**0.03720(±0.00517)**

The experimental results demonstrate our proposed model with full functionality (i.e., SEM-MacridVAE) achieve the best performance.

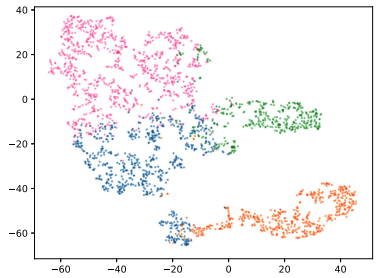
Table 3 further validate the necessity of incorporating semantic information to boost the model performance.

Discussions. The ablation studies on the two-level (macro and micro) disentanglement and the semantic (visual and categorical) signals show the improvement of the model performance brought by these core components in our proposed SEM-MacridVAE model. On the one hand, the macro disentanglement helps to capture user high-level intentions from a diversity of potential interests, while the micro disentanglement targets at learning the user low-level preferences in a more fine-grained way. Taking both levels of intentions into consideration enables the proposed SEM-MacridVAE model to more accurately infer user interests, thus improving the final recommendation performance. On the other hand, incorporating the categorical signals can more accurately align the learned macro disentangled intentions (i.e., the prototype concepts) to the ground-truth item categories, thus leading to better representation learning. Moreover, employing the visual signals to initialize the item embeddings is able to ease the process of learning detailed user visual preferences (e.g., the color of a bag) for our SEM-MacridVAE model. These two types of semantic supervision are taken into account to improve both the disentanglement and explainability of SEM-MacridVAE with human prior, which is illustrated by the ablation studies in Table 2 as we expect.

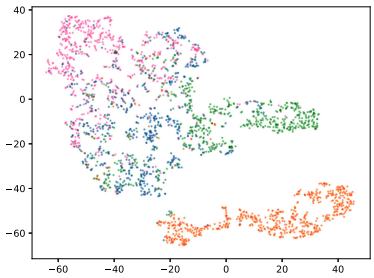
4.3 Macro Disentanglement

In order that we can qualitatively examine to which degree our proposed SEM-MacridVAE model is able to achieve macro disentanglement, the high-dimensional representations learned by our approach are visualized on three Amazon datasets, i.e., *Amazon Musical Instruments*, *Amazon Home&Kitchen* and *Amazon Clothing&Shoes&Jewelry*. We pick subsets from these three Amazon datasets respectively such that every item only belongs to one category, and the number of items in every category is balanced to be closed to each other. Concretely, we set K to 4 for *Amazon Musical Instruments*, 5 for *Amazon Home&Kitchen* and 3 for *Amazon Clothing&Shoes&Jewelry* i.e., the number of ground-truth categories, when training our model. We then match each learned prototype to a ground truth category by greedily minimizing the distance between the prototype and the center of the items from that category. We visualize the item representations and the user representations together using t-SNE [53], where we treat the K components of a user as K individual points and keep only the two components that have the highest confidence levels. The confidence of component k is defined as $\sum_{i:x_{u,i}>0} c_{i,k}$, where $c_{i,k}$ is the value inferred by our SEM-MacridVAE model rather than taken from the ground-truth.

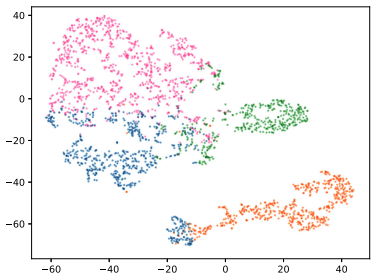
Figs. 2, 3, and 4 depict the visualization results on *Amazon Musical Instruments*, *Amazon Home&Kitchen* and *Amazon Clothing&Shoes&Jewelry* respectively. Specifically, item i is colored according to $\arg\max_k c_{i,k}$, i.e., the inferred category. The discovered clusters of items (see Figs. 2a, 3a, and 4a), learned in an unsupervised manner, align well with the ground-truth categories (see Figs. 2b, 3b, and 4b, where the color order is chosen such that the connections between the ground-truth categories and the learned clusters are easy to verify). Figs. 2c, 3c, and 4c highlight the importance of using



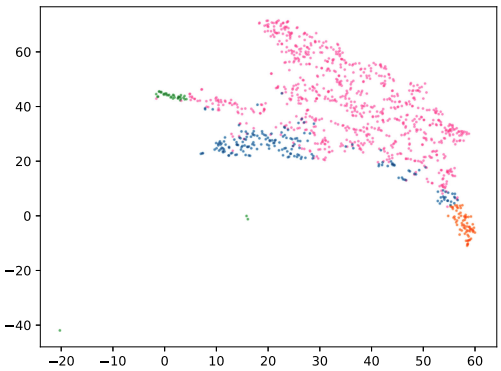
(a) Items in *Amazon Musical Instruments*, colored based on the *predicted* categories.



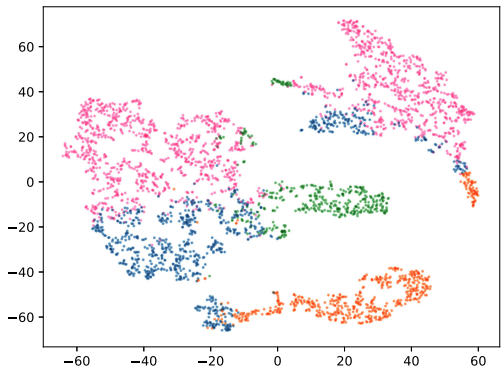
(b) Items in *Amazon Musical Instruments*, colored based on the *ground-truth* categories.



(c) Items obtained by training a new model using inner product instead of cosine.

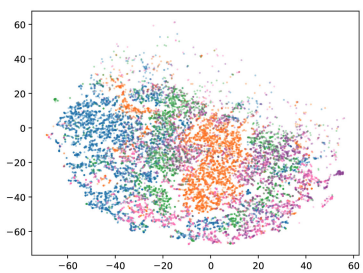


(d) Users in *Amazon Musical Instruments*, colored based on the *predicted* categories.

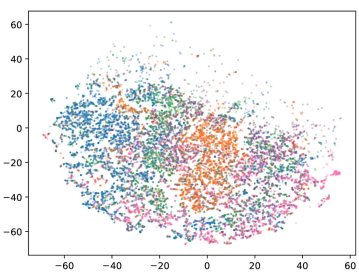


(e) Items and Users in *Amazon Musical Instruments*, colored based on the *predicted* categories.

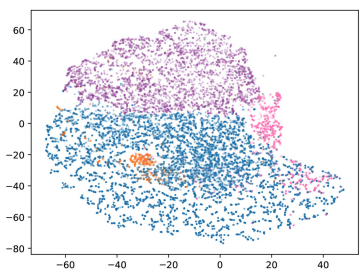
Fig. 2. Visualization of macro disentanglement for *Amazon Musical Instruments* with $K = 4$, where item i is colored according to $\arg\max_k c_{i,k}$, i.e., the inferred category. The discovered clusters of items (see Fig. 2a) align well with the ground-truth categories (see Fig. 2b, where the color order is chosen such that the connections between the ground-truth categories and the learned clusters are easy to verify). Fig. 2c highlights the importance of using cosine similarity rather than inner product to combat mode collapse, where items are obtained by training a new model that uses inner product instead of cosine, colored according to the value of $\arg\max_k c_{i,k}$.



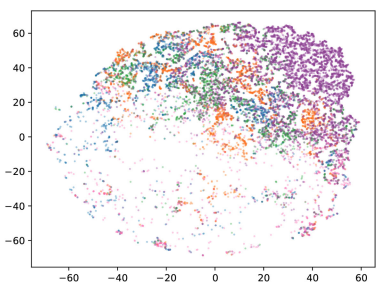
(a) Items in *Amazon Home&Kitchen*, colored based on the *predicted* categories.



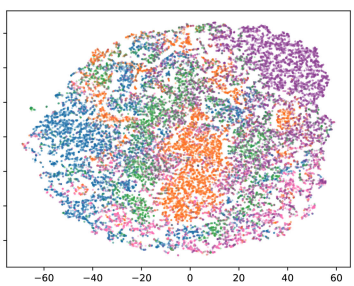
(b) Items in *Amazon Home&Kitchen*, colored based on the *ground-truth* categories.



(c) Items obtained by training a new model using inner product instead of cosine.

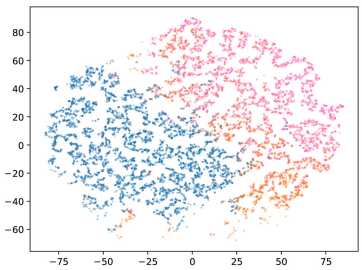


(d) Users in *Amazon Home&Kitchen*, colored based on the *predicted* categories.

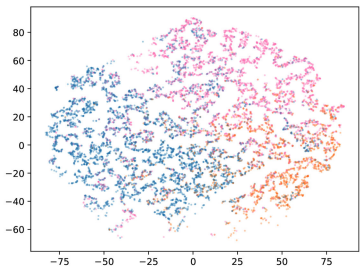


(e) Items and Users in *Amazon Home&Kitchen*, colored based on the *predicted* categories.

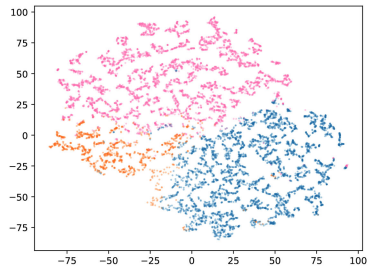
Fig. 3. Visualization of macro disentanglement for *Amazon Home&Kitchen* with $K = 5$, in the same way as Fig. 2.



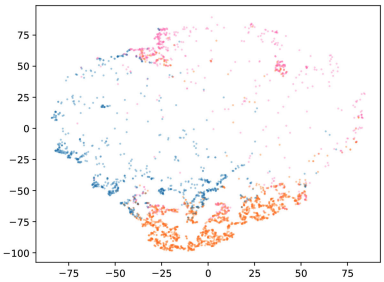
(a) Items in *Amazon Clothing&Shoes*, colored based on the *predicted* categories.



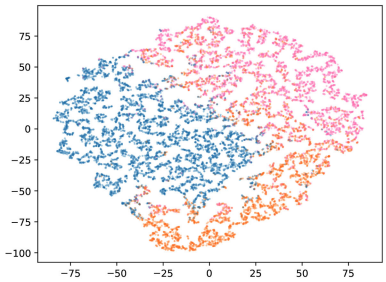
(b) Items in *Amazon Clothing&Shoes*, colored based on the *ground-truth* categories.



(c) Items obtained by training a new model using inner product instead of cosine.



(d) Users in *Amazon Clothing&Shoes&Jewelry*, colored based on the *predicted* categories.



(e) Items and Users in *Amazon Clothing&Shoes&Jewelry*, colored based on the *predicted* categories.

Fig. 4. Visualization of macro disentanglement for *Amazon Clothing&Shoes&Jewelry* with $K = 3$, in the same way as Fig. 2.

cosine similarity rather than inner product to combat mode collapse, where items are obtained by training a new model that uses inner product instead of cosine, colored according to the value of $\arg\max_k c_{i,k}$.

4.3.1 Interpretability

Again, we take the Amazon datasets as instances. Figs. 2a, 3a, and 4a show the clusters inferred based on the prototypes on *Amazon Musical Instruments*, *Amazon Home&Kitchen* and *Amazon Clothing&Shoes&Jewelry* respectively, which are similar to Figs. 2b, 3b, and 4b showing the ground-truth categories respectively. Given that the proposed SEM-MacridVAE model is trained without the ground-truth category labels, we believe that our approach is able to discover and disentangle the macro structures underlying the user behavior data in an interpretable manner.

4.3.2 Cosine versus Inner Product

To further present the necessity of adopting cosine similarity instead of the widely used inner product similarity, we train an additional model using inner product rather than cosine to calculate similarity. The item representations obtained from this additional model on *Amazon Musical Instruments*, *Amazon Home&Kitchen* and *Amazon Clothing &Shoes&Jewelry* are visualized in Figs. 2c, 3c, and 4c respectively.

We observe that by adopting inner product as the similarity measure, the clustering results may vary for different datasets. The prototype assignments are similar on *Amazon Musical Instruments* (see Fig. 2c), while the majority of the items are assigned to the same prototype on *Amazon Home&Kitchen* (see Fig. 3c) or assigned to wrong prototypes different from the ground-truth prototypes on *Amazon Clothing&Shoes&Jewelry* (see Fig. 4c). On the other hand, each of the prototypes learned by the cosine-based model is assigned quite a significant number of items, being consistent with the ground-truth categories (see Figs. 2a, 3a, and 4a).

These results support our claim that an appropriate metric space such as the one defined through the cosine similarity will play an important role in preventing the mode collapse problem.

4.4 Micro Disentanglement

In addition to macro disentanglement, it is also necessary to examine the capability of our proposed SEM-MacridVAE in achieving micro disentanglement.

4.4.1 Independence

One important motivation of disentangled representation learning is to achieve robust performance by letting the dimensions capture the underlying explanatory factors in a statistically independent way.

To gain further insight, we vary the hyper-parameters related with micro disentanglement, i.e., β for our proposed SEM-MacridVAE, MacridVAE and MultiVAE. In Fig. 5, we plot the relationships between the level of independence (micro disentanglement) achieved and the corresponding recommendation performance. Each method is evaluated on *ML-latest-small*, *Amazon Musical Instruments*, *Amazon Movies&TV*, *Amazon Home&Kitchen* and *Amazon Clothing&-Shoes&Jewelry*. We quantify the level of independence achieved by a set of d-dimensional representations using $1 - \frac{2}{d(d-1)} \sum_{1 \le i < j \le d} |\text{corr}_{i,j}|$, where $\text{corr}_{i,j}$ is the correlation between dimension i and j. Fig. 5 indicates that high

Fig. 5. Micro disentanglement vs. recommendation performance. By varying the hyper-parameters β, we compare the micro disentanglement and recommendation performance. It is observed that SEM-MacridVAE overall outperforms both MacridVAE and multiVAE in terms of both micro disentanglement and recommendation performance under recall@20.

performance is in general associated with a relatively high level of independence (micro disentanglement) and SEM-MacridVAE achieves a higher level of micro disentanglement than MultiVAE.

4.4.2 Interpretability

We train our SEM-MacridVAE model with $d = 10$, $\beta = 50$ and $\sigma_0 = 0.3$ on Amazon datasets, and investigate the interpretability of the dimensions using strategies introduced in Section 3.5.

In Figs. 6, 7, and 8, we retrieve some representative dimensions that have human-understandable semantics on *Amazon Musical Instruments*, *Amazon Home&Kitchen* and *Amazon Clothing&Shoes&Jewelry* respectively. The examples from these three datasets suggest that our SEM-MacridVAE model has the potential to offer users fine-grained controls over targeted aspects of the candidate items in recommendation lists. However, we note that not all dimensions are human-understandable.

Moreover, as is pointed out by Locatello *et al.* [43], well-trained interpretable models can only be reliably identified with the help of external knowledge, e.g., item attributes. Therefore, we encourage future efforts in investigating more semi-supervised methods [44] for disentangled representation learning.

4.5 Model Complexity Analysis

In addition to the performance illustration and interpretability visualization of our proposed SEM-MacridVAE model, we further provide the model complexity analysis.

Space Complexity. As mentioned before, the space complexity, i.e., the number of parameters used by SEM-MacridVAE, is $2Md$ where M is the number of items and d is the dimension of latent factors.

Time Complexity. We analyze the time complexity according to the sequential execution pipeline of the proposed algorithm by calculating the times of element multiplication. Assuming that there are N users and M items, the *Prototype Clustering* process requires $O(MdK)$ times of

(a) Shape (b) Color (c) Style

(d) Shape, the same dimension as 6a. (e) Color, the same dimension as 6b. (f) Style, the same dimension as 6c.

(g) Shape, the same dimension as 6a. (h) Color, the same dimension as 6b. (i) Style, the same dimension as 6c.

Fig. 6. Starting from an item representation, we gradually change the value of a target dimension and retrieve the items having similar representations with the changed representations, as is described in Section 3.5. Here we present the retrieved items in *Amazon Musical Instruments* dataset when varying the target dimension and fixing others.

(a) Shape (b) Color (c) Style

(d) Shape, the same dimension as 7a. (e) Color, the same dimension as 7b. (f) Style, the same dimension as 7c.

(g) Shape, the same dimension as 7a. (h) Color, the same dimension as 7b. (i) Style, the same dimension as 7c.

Fig. 7. Starting from an item representation, we gradually change the value of a target dimension and retrieve the items having similar representations with the changed representations, as is described in Section 3.5. Here we present the retrieved items in *Amazon Home&Kitchen* dataset when varying the target dimension and fixing others.

(a) Shape (b) Color (c) Style

(d) Shape, the same dimension as 8a. (e) Color, the same dimension as 8b. (f) Style, the same dimension as 8c.

(g) Shape, the same dimension as 8a. (h) Color, the same dimension as 8b. (i) Style, the same dimension as 8c.

Fig. 8. Starting from an item representation, we gradually change the value of a target dimension and retrieve the items having similar representations with the changed representations, as is described in Section 3.5. Here we present the retrieved items in *Amazon Clothing&Shoes&Jewelry* dataset when varying the target dimension and fixing others.

multiplications. Both the *Encoding* process and *Decoding* process need $O(NMdK)$ times of multiplications. The incorporation of visual signals is pre-trained and does not consume extra running time during the training process. The incorporation of categorical signals needs $O(MK)$ times of multiplications. Summing up all the above operations, the time complexity of our SEM-MacridVAE model is $O(NMdK + NMdK + MdK + MK) = O(NMdK)$ times of element multiplications.

5 Conclusion

In this paper, we study the problem of learning disentangled representations from user behaviors, and propose our SEM-MacridVAE model capable of performing disentanglement at both macro and micro levels. We relate macro factors to high-level concepts associated with user intentions (buy a pair of shoes or a laptop) and micro factors to low-level individual user preferences (the size or the color of a shirt). Extra semantic item information, including visual semantic and categorical semantic, are further taken into consideration to boost recommendation performance. Empirical results including both quantitative and qualitative experiments over five real-world datasets demonstrate the effectiveness of our approach in learning disentangled representations that are robust, interpretable, and controllable.

As for future work, it will be an interesting and promising research direction for future investigation to explore novel applications that can benefit in the interpretability and controllability brought by the disentangled representations.

Acknowledgments

Hong Chen, Yuwei Zhou and Jianxin Ma have contributed equally to this work.

References

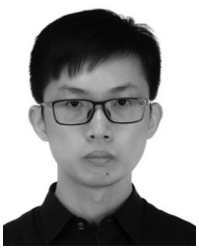
[1] A. Achille and S. Soatto, "Information dropout: Learning optimal representations through noisy computation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 12, pp. 2897–2905, Dec. 2018.
[2] A. A. Alemi, I. Fischer, Joshua V Dillon, and K. Murphy, "Deep variational information bottleneck," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–6.
[3] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.
[4] J. Bergstra, D. Yamins, and D. D. Cox, "Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms," in *Proc. 12th Python Sci. Conf.*, 2013, pp. 13–20.
[5] J. S. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 2546–2554.
[6] D. Bouchacourt, R. Tomioka, and S. Nowozin, "Multi-level variational autoencoder: Learning disentangled representations from grouped observations," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 2095–2102.
[7] C. P. Burgess *et al.*, "Understanding disentangling in $beta$-vae," 2018, *arXiv:1804.03599*.
[8] T. Q. Chen, X. Li, R. B. Grosse, and D. K. Duvenaud, "Isolating sources of disentanglement in variational autoencoders," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 2610–2620.
[9] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "Infogan: Interpretable representation learning by information maximizing generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 2172–2180.
[10] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for youtube recommendations," in *Proc. 10th ACM Conf. Recommender Syst.*, 2016, pp. 191–198.
[11] M. Deshpande and G. Karypis, "Item-based top-n recommendation algorithms," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 143–177, 2004.
[12] N. Dilokthanakul *et al.*, "Deep unsupervised clustering with gaussian mixture variational autoencoders," 2016, *arXiv:1611.02648*.
[13] E. Dupont, "Learning disentangled joint continuous and discrete representations," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 710–720.
[14] S. M. A. Eslami *et al.*, "Neural scene representation and rendering," *Sci.*, vol. 360, no. 6394, pp. 1204–1210, 2018.
[15] G. Guo, J. Zhang, and N. Yorke-Smith, "A novel bayesian similarity measure for recommender systems," in *Proc. 23rd Int. Joint Conf. Artif. Intell.*, 2013, pp. 2619–2625.
[16] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *ACM Trans. Interactive Intell. Syst.*, vol. 5, no. 4, 2016, Art. no. 19.
[17] R. He, Wee S. Lee, Hwee T. Ng, and D. Dahlmeier, "An unsupervised neural attention model for aspect extraction," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, 2017, pp. 388–397.
[18] R. He and J. McAuley, "UPS and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering," in *Proc. 25th Int. Conf. World Wide Web*, 2016, pp. 507–517.
[19] X. He, T. Chen, Min-Y. Kan, and X. Chen, "Trirank: Review-aware explainable recommendation by modeling aspects," in *Proc. 24th ACM Int. Conf. Inf. Knowl. Manage.*, 2015, pp. 1661–1670.
[20] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proc. 26th Int. Conf. World Wide Web*, 2017, pp. 173–182.
[21] I. Higgins *et al.*, "beta-VAE: Learning basic visual concepts with a constrained variational framework," in *Proc. Int. Conf. Learn. Representations*, 2017, pp. 1–6.
[22] J.-T. Hsieh, B. Liu, D.-A. Huang, L. F. Fei-Fei, and J. C. Niebles, "Learning to decompose and disentangle representations for video prediction," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 517–526.
[23] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in *Proc. 8th IEEE Int. Conf. Data Mining*, 2008, pp. 263–272.
[24] R. A. Jacobs *et al.*, "Adaptive mixtures of local experts," *Neural Computation*, vol. 3, no. 1, pp. 79–87, 1991.
[25] S. Jain, E. Banner, J.-W. van de Meent, I. J. Marshall, and B. C. Wallace, "Learning disentangled representations of texts with application to biomedical abstracts," 2018, *arXiv:1804.07212*.
[26] M. Jamali and M. Ester, "Trustwalker: A random walk model for combining trust-based and item-based recommendation," in *Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2009, pp. 397–406.
[27] M. Jamali and M. Ester, "A matrix factorization technique with trust propagation for recommendation in social networks," in *Proc. 4th ACM Conf. Recommender Syst.*, 2010, pp. 135–142.
[28] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," 2016, *arXiv:1611.01144*.
[29] S. Jean, K. Cho, R. Memisevic, and Y. Bengio, "On using very large target vocabulary for neural machine translation," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics*, 2015, pp. 1–10.
[30] Z. Jiang, Y. Zheng, H. Tan, B. Tang, and H. Zhou, "Variational deep embedding: an unsupervised and generative approach to clustering," in *Proc. 26th Int. Joint Conf. Artif. Int.*, 2017, pp. 1965–1972.
[31] H. Kim and A. Mnih, "Disentangling by factorising," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 2654–2663.
[32] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2014, pp. 1–6.
[33] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," 2013, *arXiv:1312.6114*.
[34] N. Komodakis and S. Gidaris, "Unsupervised representation learning by predicting image rotations," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–6.
[35] Y. Koren, "Factorization meets the neighborhood: A multifaceted collaborative filtering model," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2008, pp. 426–434.
[36] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Comput.*, vol. 42, no. 8, pp. 30–37, 2009.
[37] A. Kosiorek, H. Kim, Y. W. Teh, and I. Posner, "Sequential attend, infer, repeat: Generative modelling of moving objects," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 8606–8616.

[38] X. Li and J. She, "Collaborative variational autoencoder for recommender systems," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2017, pp. 305–314.
[39] Y. Li, P. Zhao, D. Wang, X. Xian, Y. Liu, and V. S. Sheng, "Learning disentangled user representation based on controllable vae for recommendation," in *Proc. Int. Conf. Database Syst. Adv. Appl.*, 2021, pp. 179–194.
[40] D. Liang, R. G. Krishnan, M. D. Hoffman, and T. Jebara, "Variational autoencoders for collaborative filtering," in *Proc. World Wide Web Conf.*, 2018, pp. 689–698.
[41] B. Liu, A. Sheth, U. Weinsberg, J. Chandrashekar, and R. Govindan, "Adreveal: Improving transparency into online targeted advertising," in *Proc. 12th ACM Workshop Hot Top. Netw.*, 2013, pp. 1–7.
[42] N. N. Liu and Q. Yang, "Eigenrank: A ranking-oriented approach to collaborative filtering," in *Proc. 31st Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2008, pp. 83–90.
[43] F. Locatello, S. Bauer, M. Lucic, S. Gelly, B. Schölkopf, and O. Bachem, "Challenging common assumptions in the unsupervised learning of disentangled representations," in *Proc. 36th Int. Conf. Mach. Learn.*, 2019, pp. 4114–4124.
[44] F. Locatello, M. Tschannen, S. Bauer, G. Rätsch, B. Schölkopf, and O. Bachem, "Disentangling factors of variation using few labels," 2019, *arXiv:1905.01258*.
[45] B. Lowere, "The harpy speech recognition system," Ph.D. thesis, Dept. Comput. Sci., Carnegie Mellon Univ., Pittsburgh, PA, 1976.
[46] H. Ma, I. King, and M. R. Lyu, "Learning to recommend with social trust ensemble," in *Proc. 32nd Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2009, pp. 203–210.
[47] H. Ma, H. Yang, M. R. Lyu, and I. King, "SoRec: Social recommendation using probabilistic matrix factorization," in *Proc. 17th ACM Conf. Inf. Knowl. Manage.*, 2008, pp. 931–940.
[48] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King, "Recommender systems with social regularization," in *Proc. 4th ACM Int. Conf. Web Search Data Mining*, 2011, pp. 287–296.
[49] J. Ma, P. Cui, K. Kuang, X. Wang, and W. Zhu, "Disentangled graph convolutional networks," in *Proc. 36th Int. Conf. Mach. Learn.*, 2019, pp. 4212–4221.
[50] J. Ma, P. Cui, X. Wang, and W. Zhu, "Hierarchical taxonomy aware network embedding," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2018, pp. 1920–1929.
[51] J. Ma, C. Zhou, P. Cui, H. Yang, and W. Zhu, "Learning disentangled representations for recommendation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 5711–5722.
[52] L. Ma, Q. Sun, S. Georgoulis, Luc Van Gool, B. Schiele, and M. Fritz, "Disentangled person image generation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 99–108.
[53] L. van der Maaten and G. Hinton, "Visualizing data using T-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, 2008.
[54] C. J. Maddison, A. Mnih, and Y. W. Teh, "The concrete distribution: A continuous relaxation of discrete random variables," 2016, *arXiv:1611.00712*.
[55] A. Mnih and R. Salakhutdinov, "Probabilistic matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 1257–1264.
[56] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: Bayesian personalized ranking from implicit feedback," in *Proc. 25th Conf. Uncertainty Artif. Intell.*, 2009, pp. 452–461.
[57] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "Grouplens: An open architecture for collaborative filtering of netnews," in *Proc. ACM Conf. Comput. Supported Cooperative Work*, 1994, pp. 175–186.
[58] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *Proc. 31st Int. Conf. Int. Conf. Mach. Learn.*, 2014, pp. II-1278–II-1286.
[59] R. Salakhutdinov and A. Mnih, "Bayesian probabilistic matrix factorization using markov chain monte carlo," in *Proc. 25th Int. Conf. Mach. Learn.*, 2008, pp. 880–887.
[60] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proc. 10th Int. Conf. World Wide Web*, 2001, pp. 285–295.
[61] X. Sha, Z. Sun, and J. Zhang, "Disentangling multi-facet social relations for recommendation," *IEEE Trans. Comput. Soc. Syst.*, early access, Sep. 03, 2021, doi: 10.1109/TCSS.2021.3108794.
[62] N. Shazeer *et al.*, "Outrageously large neural networks: The sparsely-gated mixture-of-experts layer," 2017, *arXiv:1701.06538*.
[63] N. Tishby, F. C. Pereira, and W. Bialek, "The information bottleneck method," 2000, *physics/0004057*.
[64] N. Tishby and N. Zaslavsky, "Deep learning and the information bottleneck principle," in *Proc. IEEE Inf. Theory Workshop*, 2015, pp. 1–5.
[65] H. Wang, N. Wang, and D.-Y. Yeung, "Collaborative deep learning for recommender systems," in *Proc. 21th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2015, pp. 1235–1244.
[66] X. Wang, H. Jin An Zhang, X. He, T. Xu, and Tat-S. Chua, "Disentangled graph collaborative filtering," in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2020, pp. 1001–1010.
[67] X. Wang, H. Chen, and W. Zhu, "Multimodal disentangled representation for recommendation," in *Proc. IEEE Int. Conf. Multimedia Expo*, 2021, pp. 1–6.
[68] X. Wang, S. C. Hoi, M. Ester, J. Bu, and C. Chen, "Learning personalized preference of strong and weak ties for social recommendation," in *Proc. 26th Int. Conf. World Wide Web*, 2017, pp. 1601–1610.
[69] X. Wang, W. Lu, M. Ester, C. Wang, and C. Chen, "Social recommendation with strong and weak ties," in *Proc. 25th ACM Int. Conf. Inf. Knowl. Manage.*, 2016, pp. 5–14.
[70] X. Wang, W. Zhu, and C. Liu, "Social recommendation with optimal limited attention," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2019, pp. 1518–1527.
[71] Y. Wang, S. Tang, Y. Lei, W. Song, S. Wang, and M. Zhang, "DisenHAN: Disentangled heterogeneous graph attention network for recommendation," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Manage.*, 2020, pp. 1605–1614.
[72] Y. Wu, X. Liu, M. Xie, M. Ester, and Q. Yang, "CCCF: Improving collaborative filtering via scalable user-item co-clustering," in *Proc. 9th ACM Int. Conf. Web Search Data Mining*, 2016, pp. 73–82.
[73] S. E. Yuksel, J. N. Wilson, and P. D. Gader, "Twenty years of mixture of experts," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 8, pp. 1177–1193, Aug. 2012.
[74] Y. Zhang and J. Koren, "Efficient bayesian hierarchical user modeling for recommendation system," in *Proc. 30th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2007, pp. 47–54.
[75] Y. Zhang and J. Koren, "Efficient bayesian hierarchical user modeling for recommendation system," in *Proc. 30th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2007, pp. 47–54.
[76] Y. Zhang, Z. Zhu, Y. He, and J. Caverlee, "Content-collaborative disentanglement representation learning for enhanced recommendation," in *Proc. 14th ACM Conf. Recommender Syst. Virtual Event*, 2020, pp. 43–52.
[77] Y. Zhang, Z. Zhu, Y. He, and J. Caverlee, "Content-collaborative disentanglement representation learning for enhanced recommendation," in *Proc. 14th ACM Conf. Recommender Syst.*, 2020, pp. 43–52.
[78] Y. Zhang, G. Lai, M. Zhang, Y. Zhang, Y. Liu, and S. Ma, "Explicit factor models for explainable recommendation based on phrase-level sentiment analysis," in *Proc. 37th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2014, pp. 83–92.
[79] C. Zhou *et al.*, "Atrank: An attention-based user behavior modeling framework for recommendation," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 4564–4571.
[80] J.-Y. Zhu etal, "Visual object networks: Image generation with disentangled 3D representations," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 118–129.

Xin Wang (Member, IEEE) received the BE and PhD degrees in computer science and technology from Zhejiang University, China, and the PhD degree in computing science from Simon Fraser University, Canada. He is currently an assistant professor with the Department of Computer Science and Technology, Tsinghua University. He has authored or coauthored several high-quality research papers in top journals and conferences, including *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *IEEE Transactions on Knowledge and Data Engineering*, *IEEE Transactions on Multimedia*, ICML, NeurIPS, ACM Multimedia, KDD, WWW, and SIGIR. His research interests include multimedia intelligence, machine learning and its applications in multimedia big data. He was the recipient of 2017 China Postdoctoral innovative talents supporting program. He was the recipient of ACM China Rising Star Award in 2020.

Hong Chen received the BE from the Department of Electronic Engineering, Tsinghua University, Beijing, China, degree in 2020, where he is currently working toward the PhD degree with the Department of Computer Science and Technology. His main research interests include machine learning and multimodal information processing.

Yuwei Zhou is currently working toward the undergraduation degree with the Department of Computer Science and Technology, Tsinghua University. His research interests include knowledge-driven, interpretable and disentangled representation learning.



Jianxin Ma received the BE and master's degrees from the Department of Computer Science and Technology, Tsinghua University in 2017 and 2020, respectively, under the supervision of Wenwu Zhu and Peng Cui. He has authored or coauthored several papers at prestigious conferences, such as KDD, AAAI, ICML, and NeurIPS. His research interests include machine learning, in particular representation learning, on relational data, such as graph data and user behavior data from recommender systems. He is currently doing applied research with DAMO academy, Alibaba Inc.

Wenwu Zhu (Fellow, IEEE) received the PhD degree from New York University in 1996. He is currently a professor with the Department of Computer Science and Technology, Tsinghua University, the vice dean of the National Research Center for Information Science and Technology. Prior to his current post, he was a senior researcher and research manager with Microsoft Research Asia. From 2004 to 2008, he was the chief scientist and director with Intel Research, China. During 1996–1999, he was a member of Technical Staff with Bell Labs New Jersey. He has authored or coauthored more than 350 referred papers, and is inventor or co-inventor of more than 50 patents. His current research interests include data-driven multimedia networking and cross-media big data computing. He was the recipient of eight best paper awards, including ACM Multimedia 2012 and IEEE Transactions on Circuits and Systems for Video Technology in 2001 and 2019. During 2017–2019, he was the EiC of *IEEE Transactions on Multimedia*. He was in the steering committee of *IEEE Transactions on Multimedia* during 2015–2016 and *IEEE Transactions on Mobile Computing* during 2007–2010, respectively. He is the general co-chair of ACM Multimedia 2018 and ACM CIKM 2019, respectively. He is an AAAS fellow, SPIE fellow, and a member of The Academy of Europe (Academia Europaea).