Optimal Transport Alignment of User Preferences from Ratings and Texts

Nhu-Thuat Tran[1] **Hady W. Lauw**[1]

[1]School of Computing and Information Systems, Singapore Management University

Abstract

Modeling hidden factors driving user preferences is crucial for recommendation yet challenging due to sparse rating data. While aligning preference factors from ratings and texts, as a solution, shows improvements, existing methods impose restrictive one-to-one factor correspondences and underutilize cross-modal interest signals. We propose an optimal transport (OT) approach to address these gaps. By modeling rating- and text-based preference factors as distributions, we compute an OT plan that captures their probabilistic relationships. This plan serves dual roles: 1) to regularize cross-modal preference factors without rigid correspondence assumptions, and 2) to blend preference signals across modalities through barycentric mapping. Experiments on real-world datasets validate our method's effectiveness over competitive baselines, highlighting its novel use of OT for adaptive preference factor alignment, an underexplored direction in recommender system research.

1 INTRODUCTION

User-item interactions are driven by many hidden factors. Variational Autoencoder (VAE) offers an elegant framework to discover multiple preference factors. Current studies range from disentangling user interests merely from rating data, Ma et al. [2019b], Tran and Lauw [2023, 2024] to mining interest factors from both rating data and side information such as textual content Guo et al. [2022], visual content Wang et al. [2023a], social relationships Wang et al. [2023b], multi-modal data Avas et al. [2024].

Preference signals extracted from side information, such as textual content, could complement those derived from user ratings. Since rating data merely contains user and item IDs, which lack semantic depth, incorporating semantic textual content results in more expressive user and item representations. This method is especially beneficial for users with limited interactions, as textual content offers additional insights into their preferences. Moreover, text-based interest factors naturally offers interpretability of user preferences as humans can understand their meaning.

Tran and Lauw [2022] pioneered aligning cross-modal interest factors for text-aware recommendation, later extended to multi-modal settings by Zhou and Miao [2024]. However, these works impose a fixed one-to-one correspondence between rating- and text-based preference factors, leading to two key limitations. First, the rigid alignment of interest factors, i.e., one-to-one correspondence, is shared across all users, which ignores user-specific variations, e.g., some users may exhibit many-to-one or one-to-many interest correlations. Second, the uniform treatment of modalities treats rating- and text-based factors equally (e.g., simple averaging), assuming a universal importance of both modalities. However, users vary in how much they rely on textual versus rating signals when interacting with items. These shortcomings hinder their ability to capture nuanced, adaptive interest transference across modalities.

We propose BANDVAE, short for Barycentric AligNment of Mutually Disentangled Interest Factors with Variational AutoEncoder, a novel VAE framework that leverages optimal transport (OT) to address these gaps. BANDVAE learns soft, user-dependent alignments via an OT-enabled method, allowing more flexible and personalized cross-modal interactions; and adapts the fusion weights per user, capturing this personalized modality preference. First, BANDVAE uncovers user preference factors from ratings and texts via unsupervised prototype learning. Second, BANDVAE reframes cross-modal preference factor alignment as an OT problem: interest factors from each modality are treated as distributions, and the Sinkhorn algorithm computes a probabilistic transport plan, i.e., the alignment matrix. This matrix serves dual roles: 1) to compute a regularization term that aligns cross-modal preference factors, avoiding rigid correspondence assumptions, and 2) to adaptively transfer

Proceedings of the 41st Conference on Uncertainty in Artificial Intelligence (UAI 2025), PMLR 244:4294–4308.

interest signals across modalities via barycentric mapping. By integrating OT, BANDVAE effectively transfers preference signals, addressing personalization variability.

Contributions. Our contributions are threefold. First, we bridge the gap in text-aware recommendation by the novel use of optimal transport (OT) for preference alignment. Second, we propose BANDVAE: 1) leverages OT to adaptively aligning rating and text interest factors, and 2) utilizes barycentric mapping and OT-guided regularization for cross-modal interest transference. Third, we validate BANDVAE's effectiveness through extensive experiments on real-world datasets, demonstrating its superiority over existing models. In addition, we provide qualitative analysis to offer insight into the inner workings of our proposed optimal transport-based alignment of preference factors.

2 RELATED WORK

VAE-based disentangled representation learning aims at uncovering latent explanatory factors, enabling robust modeling of complex data patterns Bengio et al. [2013]. Early works Higgins et al. [2017], Burgess et al. [2018], Kim and Mnih [2018], Chen et al. [2018], Locatello et al. [2019] focused on disentangling each dimension of representation vector to encodes a distinct feature. Recent advances extend this to disentangle user preference factors at both dimension and intention levels Ma et al. [2019b], Tran and Lauw [2023, 2024], Guo et al. [2024]. To enhance disentanglement, researchers have incorporated auxiliary data, e.g., textual content Guo et al. [2022], visual information Wang et al. [2023a], multi-modal features Avas et al. [2024], and social relationships Wang et al. [2023b]. While these works share our goal of disentangling user preferences, our key distinction lies in leveraging optimal transport (OT) to align rating and text interest factors probabilistically.

Text-aware recommendation improves performance by integrating item textual content via neural networks Wang and Blei [2011], Wang et al. [2015], Kim et al. [2016], Ma et al. [2019a]. VAEs have since been widely adopted, both in non-disentangled Zhu and Chen [2023], Li and She [2017], Zhu and Chen [2022] and disentangled forms Zhang et al. [2020], Tran and Lauw [2022], Guo et al. [2022]. Our work differs by introducing optimal transport to probabilistically align rating and text interest factors. Hou et al. [2022], Rajput et al. [2023] adopted pre-trained language models (PLMs) for text-based recommendation. However, PLMs compress textual data into a single vector, overlooking its multi-faceted structure. Instead, we focus on disentangling multiple interest factors from texts. Zhou and Miao [2024], Avas et al. [2024] leverage multi-modal data (e.g., text and images), which differs from our focus on aligning ratings and texts. While related to hybrid recommendation Rendle [2010], Frolov and Oseledets [2019], Jeunen et al. [2020], Xu et al. [2023], we follow warm-start setting rather than addressing the cold-start problem, which is beyond our scope.

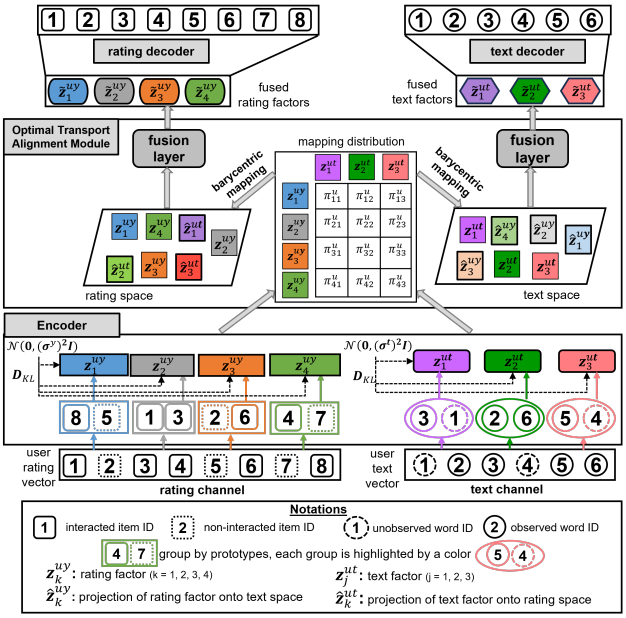


Figure 1: Illustration of our model BANDVAE. Numbers are IDs of items and words. Note that dashed items and words are not considered. BANDVAE employs optimal transport to align and fuse user interest factors from ratings and texts.

Optimal Transport (OT) offers a principled framework for measuring distances between distributions and mapping them efficiently Peyré and Cuturi [2019]. Sinkhorn algorithm Cuturi [2013], Genevay et al. [2018] have enabled OT applications in domain adaptation Courty et al. [2014, 2017], model fusion Singh and Jaggi [2020], attention Zhang et al. [2021], Sander et al. [2022], multi-modal knowledge fusion Cao et al. [2022], topic modeling Wu et al. [2023], and object-centric learning Zhang et al. [2023]. In recommender systems, OT has been applied to graph-based aggregation Chen et al. [2022] and cross-domain user correspondence Liu et al. [2024]. Our work diverges by using OT to probabilistically align cross-modal interest factors. This approach not only improves recommendation accuracy but also provides interpretable insights into user-text interactions, a novel application of OT in this domain.

3 METHODOLOGY

Preliminaries and notations. Our setting includes M users indexed by u, and N items indexed by i. For user u, let $\mathbf{y}^u \in \{0,1\}^N$ be their historical interactions with items. $\mathbf{y}_i^u = 1$ indicates an observed interaction between u and i, otherwise $\mathbf{y}_i^u = 0$. For item i, let $\mathbf{w}^i \in \mathbb{R}^W$ be the tf-idf representation of its textual content. W is the number of words in the vocabulary. Let $\mathbf{t}^u \in \mathbb{R}^W$ be textual vector of user u, obtained from their adopted items as $\mathbf{t}^u = \frac{\sum_i \mathbf{y}_i^u \mathbf{w}^i}{\sum_i \mathbf{y}_i^u}$.

Let $\mathbf{H} \in \mathbb{R}^{N \times d}$ be the embedding matrix of N items, which is the weight of decoder of rating channel in Figure 1. The encoder of rating channel is a two-layered Multilayer Perceptron (MLP). Inside the text channel in Figure 1, the weight of decoder is denoted by $\mathbf{E}^{W \times d}$, which stores W d-dimensional vectors of W words in the vocabulary. The encoder of text channel includes another two-layered Multilayer Perceptron (MLP) module. Our initial exploration leveraged a BERT-style pre-trained language model (PLM) to generate initial $\mathbf{H}$ and $\mathbf{E}$ from textual content but did not produce favorable recommendation accuracy. Thus, we do not include PLM for fair comparison with baselines and leave the integration of pre-trained models like CLIP or Large Language Models for a future study.

3.1 OVERVIEW OF BANDVAE

Figure 1 illustrates our model BANDVAE, which discovers user preferences from ratings $\mathbf{y}^u$ and texts $\mathbf{t}^u$ for a user u. Concretely, $\mathbf{z}^{uy} = \{\mathbf{z}_k^{uy}\}_{k=1}^K$ assuming K rating interest factors underlying $\mathbf{y}^u$. Similarly, $\mathbf{z}^{ut} = \{\mathbf{z}_j^{ut}\}_{j=1}^J$ consists of J text interest factors behind $\mathbf{t}^u$. Then, we align these rating and text factors via optimal transport, leveraging cross-modal interest signals to improve performance. Like previous VAE-based multi-interest modeling studies, BANDVAE includes three main components: **a) Encoder $\mathcal{E}$** derives K rating interest factors and J text interest factors for each user; **b) Alignment module $\mathcal{A}$** aligns and fuses user interest factors from ratings and texts; **c) Decoder $\mathcal{D}$** reconstructs observed user-item ratings and user associated texts. The key difference in BANDVAE lies in its novel adaptation of optimal transport for aligning and fusing cross-modal interest factors, which will be elaborated in the next section.

3.2 USER INTEREST LEARNING

Rating encoder $\mathcal{E}^y$. To model multiple user interests, we aim at uncovering the structure of their interacted items. Inspired by Tran and Lauw [2023], we employ prototype-based clustering to group user's interacted items into clusters, each capturing one user interests. To implement, $\mathcal{E}^y$ employs a set of K prototypes $\mathbf{m}^y \in \mathbb{R}^{K \times d}$, which are equivalent to cluster centroids. The clustering process runs iteratively for L^y iterations (indexed by l). Each iteration $l = 1, 2, ..., L^y$ computes item-cluster assignment matrix $\mathbf{A}_l^{uy} \in \mathbb{R}^{N \times K}$ then updates K prototypes (indexed by k) as

$$\mathbf{A}_l^{uy} = \eta\left(\frac{\mathbf{H} \cdot (\mathbf{m}_l^{uy})^T}{\tau \cdot ||\mathbf{H}||_2 \cdot ||\mathbf{m}_l^{uy}||_2}\right) \Longrightarrow \mathbf{m}_{lk}^{uy} = \sum_i \mathbf{y}_i^u (\mathbf{A}_l^{uy})_{ik} \mathbf{H}_i \quad (1)$$

$\mathbf{m}_l^{uy} = \{\mathbf{m}_{lk}^{uy}\}_{k=1}^K$ and $\mathbf{m}_1^{uy} = \mathbf{m}^y$. We implement η as the widely adopted Gumbel-Softmax Jang et al. [2017], Maddison et al. [2017] to ensure fair comparison with baselines. For an item i, its assignment score towards K clusters satisfies: $\sum_{k=1}^K (\mathbf{A}_l^{uy})_{ik} = 1$ and $(\mathbf{A}_l^{uy})_{ik} \geq 0$. $\mathbf{A}_l^{uy}$ is based on cosine similarity between $\mathbf{H}$ and $\mathbf{m}_l^{uy}$. τ is a small number to concentrate weights on the most probable prototype. While iteratively updating $\mathbf{m}_l^{uy}$ in Equation 1 leads to more informative prototypes than randomly initialized $\mathbf{m}^y$, it creates a recurrent network, which is difficult to train. Thus, we apply Implicit Differentiation by stopping gradient (sg) update to prototypes after $L^y - 1$ iterations, i.e., $\mathbf{m}_{L^y-1}^{uy} = \text{sg}(\mathbf{m}_{L^y-1}^{uy})$, then obtain assignment matrix $\mathbf{A}_{L^y}^{uy}$ as in Equation 1. For simplicity, we omit index L^y hereafter.

Next, we estimate the parameters of Gaussian distribution for each interest factor k via rating encoder's MLP

$$(\mathbf{r}_k^{uy}, \mathbf{o}_k^{uy}) = \mathbf{W}_2 tanh(\mathbf{W}_1 norm(\mathbf{A}_{:,k}^{uy} \odot \mathbf{y}^u) + \mathbf{b}_1) + \mathbf{b}_2 \quad (2)$$

where $\odot$ is element-wise multiplication. $norm(\mathbf{x}) = \mathbf{x}/||\mathbf{x}||_2$ normalizes input to unit-length vector. $\mathbf{W}_1 \in \mathbb{R}^{N \times D}, \mathbf{b}_1 \in \mathbb{R}^D, \mathbf{W}_2 \in \mathbb{R}^{D \times 2d}, \mathbf{b}_2 \in \mathbb{R}^{2d}$ are weight matrices and bias vectors. Finally, the k-th rating interest factor is sampled as $\mathbf{z}_k^{uy} \sim \mathcal{N}(\boldsymbol{\mu}_k^{uy}, [diag(\boldsymbol{\sigma}_k^{uy})]^2)$. $\boldsymbol{\mu}_k^{uy} = \frac{\mathbf{r}_k^{uy}}{||\mathbf{r}_k^{uy}||_2}$; $\boldsymbol{\sigma}_k^{uy} = \sigma^y \cdot exp(-\frac{1}{2}\mathbf{o}_k^{uy})$ and σ^y is around 0.1 Ma et al. [2019b]. Assuming the independence between rating factors of user u, we have $q(\mathbf{z}^{uy}|\mathbf{y}^u, \mathbf{A}^{uy}) = \prod_{k=1}^K \mathcal{N}(\boldsymbol{\mu}_k^{uy}, [diag(\boldsymbol{\sigma}_k^{uy})]^2)$, as variation distribution, which is aligned with prior distribution $p(\mathbf{z}^{uy}) = \mathcal{N}(\mathbf{0}, (\sigma^y)^2\mathbf{I})$ via Kullback-Leibler divergence D_{KL}^y. Following the common practice in prior studies Ma et al. [2019b], Guo et al. [2022], we omit the VAE prior during evaluation for stability and comparability.

In summary, rating encoder $\mathcal{E}^y$ produces K rating interest factors $\mathbf{z}^{uy} = \{\mathbf{z}_k^{uy}\}_{k=1}^K$, assignment matrix $\mathbf{A}^{uy}$ and regularization term $D_{KL}^y(q(\mathbf{z}^{uy}|\mathbf{y}^u, \mathbf{A}^{uy})||p(\mathbf{z}^{uy}))$.

Text encoder $\mathcal{E}^t$ clusters words into J groups, each representing one user interest from texts. $\mathcal{E}^t$ functions similarly to rating encoder $\mathcal{E}^y$, but accepts different inputs: user u's textual content $\mathbf{t}^u$, prototypes $\mathbf{m}^t \in \mathbb{R}^{J \times d}$, word embedding $\mathbf{E} \in \mathbb{R}^{W \times d}$, the number of clustering iterations L^t. To save space, we present the details in the appendix.

In summary, $\mathcal{E}^t$ produces J text interest factors $\mathbf{z}^{ut} = \{\mathbf{z}_j^{ut}\}_{j=1}^J$, assignment matrix $\mathbf{A}^{ut}$, regularization term $D_{KL}^t(q(\mathbf{z}^{ut}|\mathbf{t}^u, \mathbf{A}^{ut})||p(\mathbf{z}^{ut}))$ with $p(\mathbf{z}^{ut}) = \mathcal{N}(\mathbf{0}, (\sigma^t)^2\mathbf{I})$.

3.3 INTEREST FACTOR ALIGNMENT

Our goal is to align and fuse user interest factors derived from ratings and texts to enhance recommendation accuracy. For instance, the encoder extracts the headphone interest from ratings and the alignment module aligns this headphone interest to its counterpart from texts. Similarly, a user's interest in phone cases inferred from ratings should be aligned with the corresponding interest mined from texts. However, such alignments are unavailable in advance, requiring a data-driven approach. To address this, we frame the alignment as an optimal transport (OT) problem, treating rating and text interest factors as discrete distributions. This

formulation enables to adaptively learn probabilistic correspondences between rating and text interest factors, avoiding rigid one-to-one mappings that risk suboptimal performance. The OT-derived alignment also enables mutual transference of interest signals between modalities, refining user interest representations. Beyond improving accuracy, this approach provides interpretable insights into the relationship between user ratings and textual content.

3.3.1 Optimal Transport-derived Alignment Matrix

Following OT setting, we regard rating factors $\{\mathbf{z}_k^{uy}\}_{k=1}^K$ and text factors $\{\mathbf{z}_j^{ut}\}_{j=1}^J$ as two discrete distributions. Each factor has probability weight p_k^y and p_j^t. These weights form two probability simplexes, i.e., $\sum_{k=1}^K p_k^y = 1$ and $\sum_{j=1}^J p_j^t = 1$. As the true distribution of $\mathbf{z}^{uy}$ (and $\mathbf{z}^{ut}$) is not available, we assume uniform distribution by setting weights equally $p_k^y = 1/K\ \forall k$ and $p_j^t = 1/J\ \forall j$. Let π^u be alignment matrix between rating and text factors, defined by $\mathcal{P}^u = \{\pi^u \in \mathbb{R}_+^{K\times J} | \pi^u \mathbf{1}_J = p^y, (\pi^u)^T \mathbf{1}_K = p^t\}$, $\mathbf{1}_K, \mathbf{1}_J$ are K- and J-dimensional one vectors. We solve the tractable regularized optimal transport problem Cuturi [2013] for π^u

$$\pi^u = \underset{\pi^u \in \mathcal{P}^u}{\arg\min} \langle \pi^u, \mathbf{S}^u \rangle_F - \epsilon \cdot \mathrm{Entropy}(\pi^u) \tag{3}$$

The goal of Equation 3 is to minimize the total transporting cost from rating factors to text factors of user u, resulting in optimal alignment matrix π^u. The first term is the Frobenius dot product between π^u and the cost matrix $\mathbf{S}^u \in \mathbb{R}^{K\times J}$, $\mathbf{S}_{kj}^u = ||\mathbf{z}_k^{uy} - \mathbf{z}_j^{ut}||_2^2$ and $\langle \pi^u, \mathbf{S}^u \rangle_F = \sum_{k,j} \pi_{kj}^u \mathbf{S}_{kj}^u$. The second term $\mathrm{Entropy}(\pi^u) = \sum_{k,j} -\pi_{kj}^u log(\pi_{kj}^u)$ is the entropy of π^u, which is added to make the problem tractable. ϵ is a hyper-parameter. Small ϵ results in skewed distribution while large ϵ leads to relatively uniform distribution in π^u.

To efficiently solve Equation 3 for π^u, we employ Sinkhorn algorithm Cuturi [2013] that alternatively calculates two scaling vectors $\mathbf{u}$ and $\mathbf{v}$ until convergence as presented in Algorithm 1. This approach is efficient as it is differentiable and is highly supported on GPU for matrix multiplication. Since Sinkhorn algorithm is theoretically proven to converge to the optimal transport plan Peyré and Cuturi [2019], we therefore stop gradient update to π^u after obtained from Algorithm 1 to improve efficiency. Empirically, we found that this practice speeds up training while preserving accuracy.

3.3.2 Transference between interest factors

To enable cross-modal interest transfer and enhance user representations, we propose two approaches. First, we introduce an alignment probability-guided regularization term, where the OT-derived alignment matrix π^u guides the learning of connections between rating and text interest factors. Second, we employ a barycentric mapping strategy, projecting rating factors into the text space (and vice versa). This facilitates bidirectional interest transfer, refining interest representations and improving recommendation accuracy.

Algorithm 1 Alignment matrix between interest factors

Input: $\{\mathbf{z}_k^{uy}\}_{k=1}^K, \{\mathbf{z}_j^{uy}\}_{j=1}^J, \epsilon$
Output: π^u

$\mathbf{S}_{kj}^u = ||\mathbf{z}_k^{uy} - \mathbf{z}_j^{ut}||_2^2, \mathbf{S}^u \in \mathbb{R}^{K\times J}$
$\mathbf{B}^u = exp(-\mathbf{S}^u/\epsilon)$
initialize $\mathbf{v} \leftarrow \mathbf{1}_J$
while not converged **do**
  $\mathbf{u} \leftarrow \frac{1}{K} \frac{\mathbf{1}_K}{\mathbf{B}^u \mathbf{v}}; \mathbf{v} \leftarrow \frac{1}{J} \frac{\mathbf{1}_J}{(\mathbf{B}^u)^T \mathbf{u}}$
end while
return $\pi^u = diag(\mathbf{u}) \mathbf{B}^u diag(\mathbf{v})$

Alignment probability-guided regularization optimizes a regularization term guided by π^u as following

$$\mathcal{L}_u^{OT} = \sum_{k=1}^K \sum_{j=1}^J \pi_{kj}^u \cdot ||\mathbf{z}_k^{uy} - \mathbf{z}_j^{ut}||_2^2 \tag{4}$$

Thanks to π^u, the optimization will focus on transferring interest between most probably aligned factors. Note that while regularization-based interest transfer has been explored in Wang et al. [2015], Li and She [2017], Tran and Lauw [2022], both in non-disentangled and disentangled fashions, none of these is guided by alignment probabilities.

Mapping and fusing. To capture interest signals across modalities, we fuse rating and text factors via barycentric mapping Perrot et al. [2016], Courty et al. [2017].

Barycentric Mapping. Note each entry in the alignment matrix π_{kj}^u indicates how much of the probability mass from a rating factor, $\mathbf{z}_k^{uy}$, should be transferred to the corresponding text factor, $\mathbf{z}_j^{ut}$. Thus, using π^u, we can map rating factors onto text space via solving $\hat{\mathbf{z}}_k^{uy} = \arg\min_{\mathbf{s}^t \in \mathbb{R}^d} \sum_j \pi_{kj}^u c(\mathbf{s}^t, \mathbf{z}_j^{ut})$, where $\hat{\mathbf{z}}_k^{uy}$ is the transformation of $\mathbf{z}_k^{uy}$ in text space and $c(\cdot,\cdot)$ is the cost function. Following Courty et al. [2017], the solution for $\hat{\mathbf{z}}_k^{uy}$ is

$$\hat{\mathbf{z}}_k^{uy} = diag(\pi_k^u \mathbf{1}_J)^{-1} \pi_k^u \mathbf{z}^{ut} \tag{5}$$

where $\mathbf{z}^{ut} = \{\mathbf{z}_j^{ut}\}_{j=1}^J \in \mathbb{R}^{J\times d}$. We repeat Equation 5 $\forall k = 1, 2, ..., K$ to obtain $\{\hat{\mathbf{z}}_k^{uy}\}_{k=1}^K$. Similarly, we compute $\hat{\mathbf{z}}_j^{ut}$, the transformation of text factor $\mathbf{z}_j^{ut}$ onto rating space

$$\hat{\mathbf{z}}_j^{ut} = diag((\pi^u)_j^T \mathbf{1}_K)^{-1} (\pi^u)_j^T \mathbf{z}^{uy} \tag{6}$$

where $\mathbf{z}^{uy} = \{\mathbf{z}_k^{uy}\}_{k=1}^K \in \mathbb{R}^{K\times d}$. We obtain $\{\hat{\mathbf{z}}_j^{ut}\}_{j=1}^J$ by applying Equation 6 for $\forall j = 1, 2, ..., J$.

Adaptively Fusing. We fuse $\{\mathbf{z}_k^{uy}\}_{k=1}^K$ with their transformed versions $\{\hat{\mathbf{z}}_k^{uy}\}_{k=1}^K$ to create input for rating decoder, enabling transferring rating signals explicitly to text space via $\{\hat{\mathbf{z}}_k^{uy}\}_{k=1}^K$. As each user's decision bases individually on ratings and texts, we design an adaptive fusion layer as

$$\tilde{\mathbf{z}}_k^{uy} = \mathbf{z}_k^{uy} + \rho_k^{uy} \cdot \hat{\mathbf{z}}_k^{uy}, \quad \forall k = 1, 2, ..., K \tag{7}$$

$\rho_k^{uy} = log(1 + exp(\zeta([\mathbf{z}_k^{uy}; \hat{\mathbf{z}}_k^{uy}])))$ is the fusion weight and ; is concatenation. $\zeta : \mathbb{R}^{2d} \rightarrow \mathbb{R}^1$ is a neural network. Similarly, a fusion layer is applied for text factors

$$\tilde{\mathbf{z}}_j^{ut} = \mathbf{z}_j^{ut} + \rho_j^{ut} \cdot \hat{\mathbf{z}}_j^{ut}, \quad \forall j = 1, 2, ..., J \tag{8}$$

$\rho_j^{ut} = log(1+exp(\zeta([\mathbf{z}_j^{ut}; \hat{\mathbf{z}}_j^{ut}]))). \zeta$ here is the same as one in rating fusion. By this design, ρ^{uy} and ρ^{ut} are dynamically learned for each individual user. Then, $\tilde{\mathbf{z}}^{uy} = \{\tilde{\mathbf{z}}_k^{uy}\}_{k=1}^K$ and $\tilde{\mathbf{z}}^{ut} = \{\tilde{\mathbf{z}}_j^{ut}\}_{j=1}^J$ go to rating and text decoders, respectively.

3.4 DECODER

Rating decoder $\mathcal{D}^y$ of rating channel accepts user u's fused rating factors $\tilde{\mathbf{z}}^{uy} = \{\tilde{\mathbf{z}}_k^{uy}\}_{k=1}^K$ as input. $\mathcal{D}^y$ predicts the probability of an interaction between a user u and an item i as the weighted sum of rating factors' predictions

$$p(\mathbf{y}_i^u) = \frac{\sum_{k=1}^K \mathbf{A}_{ik}^{uy} \cdot exp(s(\tilde{\mathbf{z}}_k^{uy}, \mathbf{H}_i)/\tau)}{\sum_{i'=1}^N \sum_{k=1}^K \mathbf{A}_{ik}^{uy} \cdot exp(s(\tilde{\mathbf{z}}_k^{uy}, \mathbf{H}_{i'})/\tau)} \tag{9}$$

$s(\cdot, \cdot)$ is cosine similarity. The learning objective includes cross-entropy loss to match the predicted interaction probabilities $p(\mathbf{y}^u)$ with observed interactions $\mathbf{y}^u$ and KL divergence term (controlled by β^y) from rating encoder $\boldsymbol{\mathcal{E}}^y$.

$$\mathcal{L}_u^y = \sum_{i=1}^N -\mathbf{y}_i^u ln\ p(\mathbf{y}_i^u) + \beta^y \cdot D_{KL}^y(q(\mathbf{z}^{uy}|\mathbf{y}^u, \mathbf{A}^{uy})||p(\mathbf{z}^{uy})) \tag{10}$$

Text decoder $\mathcal{D}^t$ of text channel has user u's fused text factors $\tilde{\mathbf{z}}^{ut} = \{\tilde{\mathbf{z}}_j^{ut}\}_{j=1}^J$ as input. $\mathcal{D}^t$ predicts the probability of a word w appearing in textual content associated with user u as the weighted sum of text factors' predictions

$$p(\mathbf{t}_w^u) = \frac{\sum_{j=1}^J \mathbf{A}_{wj}^{ut} \cdot exp(s(\tilde{\mathbf{z}}_j^{ut}, \mathbf{E}_w)/\tau)}{\sum_{w'=1}^W \sum_{j=1}^J \mathbf{A}_{wj}^{ut} \cdot exp(s(\tilde{\mathbf{z}}_j^{ut}, \mathbf{E}_{w'})/\tau)} \tag{11}$$

$s(\cdot, \cdot)$ is cosine similarity. Similarly, the learning objective includes cross-entropy term to match predicted probability $p(\mathbf{t}^u)$ with observed textual information $\mathbf{t}^u$ and KL divergence term derived from text encoder $\boldsymbol{\mathcal{E}}^t$, controlled by β^t.

$$\mathcal{L}_u^t = \sum_{w=1}^W -\mathbf{t}_w^u ln\ p(\mathbf{t}_w^u) + \beta^t \cdot D_{KL}^t(q(\mathbf{z}^{ut}|\mathbf{t}^u, \mathbf{A}^{ut})||p(\mathbf{z}^{ut})) \tag{12}$$

Final learning objective. Given a batch of user $\mathcal{B}$, BANDVAE minimizes $\mathcal{L} = \frac{1}{||\mathcal{B}||} \sum_{u \in \mathcal{B}} \mathcal{L}_u^y + \lambda_t \cdot \mathcal{L}_u^t + \lambda_r \cdot \mathcal{L}_u^{OT}$. λ_t and λ_r are hyper-parameters. Algorithm 2 presents the training procedure of BANDVAE.

3.5 EXTENSION

Our method, while focuses on two modalities, can be easily extended to multiple modalities. Suppose there is a set of user-associated modalities $\mathcal{T}$ (e.g., text, image, audio) in addition to rating modality y. For each modality $m \in \mathcal{T} \cup \{y\}$, an encoder $\mathcal{E}^m$ (Section 3.2) is employed to discover K^m interest factors $\{\mathbf{z}_k^{um}\}_{k=1}^{K^m}$ for each user u. Then, the OT-based alignment module $\mathcal{A}$ (Section 3.3) fuses interest factors from m with those from y to obtain $\tilde{\mathbf{z}}^{um}$. Each modality m has a decoder $\mathcal{D}^m$ to reconstruct the respective input, where $\mathcal{D}^m$ accepts $\tilde{\mathbf{z}}^{um}$ as input. The learning objective becomes $\mathcal{L} = \mathcal{L}_y^{recon} + \sum_m^{\mathcal{T}}(\lambda_m \cdot \mathcal{L}_m^{recon} + \lambda_{rm}\mathcal{L}_{ym}^{OT})$, where $\mathcal{L}^{recon}$ is the reconstruction loss as Equation 10 and 12 while $\mathcal{L}_{ym}^{OT}$ regularizes interest factors of two modalities m and y as Equation 4. These losses are controlled by λ_m and λ_{rm}.

Algorithm 2 Training procedure of BANDVAE

Input:

- Rating and text vectors of M users $\{\mathbf{y}^u\}_{u=1}^M$ and $\{\mathbf{t}^u\}_{u=1}^M$.
- Rating channel's parameters Θ^y: item matrix in decoder $\mathbf{H} \in \mathbb{R}^{N \times d}$; prototype representations $\mathbf{m}^y \in \mathbb{R}^{K \times d}$; MLP's parameters: $\mathbf{W}_1 \in \mathbb{R}^{N \times D}, \mathbf{b}_1 \in \mathbb{R}^D, \mathbf{W}_2 \in \mathbb{R}^{D \times 2d}, \mathbf{b}_2 \in \mathbb{R}^{2d}$
- Text channel's parameters Θ^t: decoder weight matrix $\mathbf{E} \in \mathbb{R}^{W \times d}$; prototype representations $\mathbf{m}^t \in \mathbb{R}^{J \times d}$; MLP's parameters: $\mathbf{W}_1' \in \mathbb{R}^{W \times D}, \mathbf{b}_1' \in \mathbb{R}^D, \mathbf{W}_2' \in \mathbb{R}^{D \times 2d}, \mathbf{b}_2' \in \mathbb{R}^{2d}$
- Parameters of fusion layer $\zeta : \mathbb{R}^{2d} \rightarrow \mathbb{R}^1$
- Hyper-parameters $\tau, \epsilon, \sigma^y, \sigma^t, L^y, L^t$

Output: Updated Θ^y and Θ^t

for *each batch of user* $\mathcal{B}$ **do**
  for *user* $u \in \mathcal{B}$ **do**
    $\{\mathbf{z}_k^{uy}\}_{k=1}^K \leftarrow \boldsymbol{\mathcal{E}}^y(\mathbf{y}^u, \mathbf{m}^y, \mathbf{H}, \tau, \sigma^y, L^y)$ `// Rating encoder`
    $\{\mathbf{z}_j^{ut}\}_{j=1}^J \leftarrow \boldsymbol{\mathcal{E}}^t(\mathbf{t}^u, \mathbf{m}^t, \mathbf{E}, \tau, \sigma^t, L^t)$ `// Text encoder`
    $\pi^u \leftarrow$ Sinkhorn algorithm$(\{\mathbf{z}_k^{uy}\}_{k=1}^K, \{\mathbf{z}_j^{ut}\}_{j=1}^J, \epsilon)$ `// Alignment matrix in Algorithm 1`
    $\{\hat{\mathbf{z}}_k^{uy}\}_{k=1}^K \leftarrow$ Barycentric mapping$(\pi^u, \{\mathbf{z}_j^{ut}\}_{j=1}^J)$ `// Equation 5`
    $\{\hat{\mathbf{z}}_j^{ut}\}_{j=1}^J \leftarrow$ Barycentric mapping$(\pi^u, \{\mathbf{z}_k^{uy}\}_{k=1}^K)$ `// Equation 6`
    $\tilde{\mathbf{z}}^{uy} \leftarrow$ Fuse$(\{\mathbf{z}_k^{uy}\}_{k=1}^K, \{\hat{\mathbf{z}}_k^{uy}\}_{k=1}^K)$ `// Equation 7`
    $\tilde{\mathbf{z}}^{ut} \leftarrow$ Fuse$(\{\mathbf{z}_j^{ut}\}_{j=1}^J, \{\hat{\mathbf{z}}_j^{ut}\}_{j=1}^J)$ `// Equation 8`
    $\mathcal{L}_u^y \leftarrow$ Rating channel loss `// Equation 10`
    $\mathcal{L}_u^t \leftarrow$ Text channel loss `// Equation 12`
    $\mathcal{L}_u^{OT} \leftarrow$ Regularization term `// Equation 4`
  Calculate loss $\mathcal{L} = \frac{1}{||\mathcal{B}||} \sum_{u \in \mathcal{B}} \mathcal{L}_u^y + \lambda_t \cdot \mathcal{L}_u^t + \lambda_r \cdot \mathcal{L}_u^{OT}$
  Update Θ^y, Θ^t to minimize $\mathcal{L}$

4 EXPERIMENTS

Datasets. We use four publicly available datasets as shown in Table 1: **CiteULike-a**[1] contains interactions between users and scientific articles; **MovieLens**[2] includes users' ratings on movies; **Cell Phones** and **Video Games** contain user' reviews on Cell Phones & Accessories and Video Games categories of *Amazon dataset*[3].

[1] http://wanghao.in/CDL.htm

[2] https://grouplens.org/datasets/movielens/

[3] https://nijianmo.github.io/amazon/index.html

Table 1: Statistics of datasets used in our paper.

Dataset	# users	# items	# interactions	# words
CiteULike-a	5,551	16,980	204,986	8,000
MovieLens	15,000	7,892	1,005,820	8,000
Cell Phones	25,500	17,989	285,047	8,000
Video Games	52,387	16,598	473,148	8,000

For CiteULike-a, Cell Phones and Video Games datasets, we use the accompanying textual content, i.e., title & abstract for CiteULike-a and item descriptions for Amazon categories. For Cell Phones, we retain users with at least 8 interactions and items with at least 5 interactions and for Video Games, these numbers are 5 and 5, respectively. For MovieLens, we follow Zhu and Chen [2022] to extract a subset of users from ML-10M version. We keep user ratings larger than 3 as interactions Ma et al. [2019b] and collect item textual content from IMDB [4]. For all datasets, we remove stop words and only keep words with frequency higher than 3 and appearing in less than 60% of item texts and retain top $8k$ words with highest frequency as in Zhu and Chen [2022]. These strategies help ensure that even short or noisy item descriptions contribute meaningful information. Moreover, these steps are employed across baselines, ensuring fair comparison. We keep these pre-processing steps at minimal complexity so that the performance gain is attributed to our proposed aligning mechanism. Employing advanced methods to generate clean text would potentially enhance our proposed framework.

We adopt *strong generalization* setting as in Ma et al. [2019b] to construct training, validation and test sets by randomly choosing 80% of users for training and 10% of users for each validation and test sets. For validation and test sets, 20% of a user interactions is kept as the ground truth. To keep the quality of datasets, we only retain items with at least 5 words in their textual content so that the textual content brings semantic information. All cold-start items, i.e., those do no appear in training set, are discarded since there is no parameters associating with them, following the common practice in the field.

Baselines. We compare BANDVAE against state-of-the-art models, including models only utilizing ratings **MacridVAE** Ma et al. [2019b], **RecVAE** Shenbin et al. [2020], **ELSA** Vančura et al. [2022], **VALID** Tran and Lauw [2023], **FacetVAE** Tran and Lauw [2024] and models using both ratings and texts **MDCVAE** Zhu and Chen [2022], **TopicVAE**Guo et al. [2022], **ADDVAE** Tran and Lauw [2022] and **SEM-MacridVAE** Wang et al. [2023a]. Among these, **RecVAE, ELSA** and **MDCVAE** are single-interest modeling models while **MacridVAE, TopicVAE, ADDVAE, SEM-MacridVAE, VALID, FacetVAE** are multi-interest modeling models.

[4]https://datasets.imdbws.com/

- **MacridVAE** Ma et al. [2019b] introduces macro- and micro-disentanglement of user preferences via multi-prototype representation and independence regularization.
- **RecVAE** Shenbin et al. [2020] proposes composite prior, rescaling regularization term and an alternative training into a novel VAE-based recommendation model.
- **MDCVAE** Zhu and Chen [2022] regularizes decoder weights of the user-oriented autoencoder by latent embeddings inferred from textual content.
- **TopicVAE** Guo et al. [2022] improves disentangling user preferences by designing attention-based topic extraction from textual content, topic-guided contrastive loss and heuristic method to set value of regularization term.
- **ADDVAE** Tran and Lauw [2022] leverages two disentangled networks to model user's ratings and user associated texts then aligns disentangled factors from these two modalities using compositional de-attention and regularization.
- **ELSA** Vančura et al. [2022] improves SOTA linear autoencoder by factorizing hidden space into a low-rank plus sparse structure.
- **SEM-MacridVAE** Wang et al. [2023a] exploits semantic knowledge from side information to improve VAE-based disentangled recommendation models. We use tf-idf item-word matrix, i.e., $\mathbf{W} = \{\mathbf{w}^i\}_{i=1}^N$, as side information for fair comparison.
- **VALID** Tran and Lauw [2023] improves VAE-based disentangling user interests by iterative latent attention and implicit differentiation.
- **FacetVAE** Tran and Lauw [2024] disentangles multi-faceted item space and derive compositional user interests via bi-directional binding.

We follow the strong generalization setting in Ma et al. [2019b], i.e., validation and test sets include unseen users. Thus, we only involve baselines capable of predicting interactions for unseen users. While models SLIM, EASE, SimpleX are capable, they have been already outperformed by other baselines RecVAE, ELSA and VALID. Thus, we only retain state-of-the-art models as our baselines.

Implementation. For all models, we choose the hyper-parameters based on performance on validation set. Then, we retrain and report performance on test set, which is averaged over ten runs on NVIDIA RTX 2080 Ti GPU machine. Pertaining to baselines, we follow their original papers to choose hyper-parameters by performing grid search in the same range described in those papers. Regarding BANDVAE, the default settings are $D = 300$ for MovieLens and Cell Phones and $D = 600$ for CiteULike-a and Video Games after tuning from {100, 200, 300, 500, 600}; embedding size $d = 100$ for all datasets; dropout rate applied for $\mathbf{A}^{uy}$ and $\mathbf{A}^{ut}$ is 0.5; number of rating and text factors are $K = 4$ and $J = 4$, respectively (more values of K and

J are analyzed in subsequenct sections; β^y and β^t follow annealing process $min(\beta_0, \frac{update}{T})$ where $\beta_0 = 1$ for rating channel and $\beta_0 = 0.2$ for text channel, T is chosen from $\{1k, 5k, 10k, 20k\}$, and $update$ is the number parameter updates; σ^y and σ^t are chosen from $\{0.05, 0.075, 0.1\}$; the search space of λ_t and λ_r is $\{0.1, 0.2, 0.5, 1, 2, 5\}$; $\epsilon \in \{0.2, 0.5, 1\}$ in Sinkhorn algorithm. Archiecture of fusion network $\zeta : 2d \rightarrow d/2 \rightarrow 1$. The number of prototype update steps L^y in rating encoder are chosen from $\{2, 3, 4\}$ while $L^t = 1$. We train BANDVAE using Adam optimizer with learning rate 0.001 on NVIDIA RTX 2080 Ti GPU machine. Training stops after 30 epochs without improving performance on validation set. We report Recall and NDCG at top 10 and 50 with full-ranking strategy Zhao et al. [2020], i.e., test item is ranked against all items to avoid sampling bias.

4.1 RECOMMENDATION PERFORMANCE

Table 2 reports the recommendation performance. *First*, BANDVAE achieves significantly higher accuracy than baselines using textual content for multi-interest modeling TopicVAE, SemMacridVAE, and ADDVAE on CiteULike-a, Cell Phones, and Video Games, demonstrating the advantage of its optimal transport-based alignment and fusion. *Second*, BANDVAE also outperforms multi-interest models that do not use textual content MacridVAE, VALID, and FacetVAE, underscoring the value of aligning rating and text factors. Additionally, BANDVAE surpasses single-interest models MD-CVAE, RecVAE, and ELSA, highlighting the importance of capturing multiple interests. *Third*, on MovieLens, while BANDVAE performs consistently across metrics, some baselines excel only at specific metrics. For example, RecVAE's composite prior aids Recall@10, but BANDVAE achieves notably higher Recall@50 and NDCG@50. SEMMacridVAE and TopicVAE learns item representations from texts, attaining comparable accuracy with BANDVAE w.r.t. only top 10 metrics. In contrast, BANDVAE is evidently better than these two w.r.t. top 50 metrics.

4.2 MODEL ANALYSIS

We conduct experiments to gain insights into BANDVAE's inner working. We present more ablative studies in the appendix to further understand BANDVAE.

Alignment method. In Figure 2, we analyze three alternatives to understand the derivation of π^u.

- *Sinkhorn (Sink)* is Sinkhorn algorithm in Algorithm 1.
- *Normalization (Norm)* generates π^u by normalizing negative distance between disentangled factors from two modalities, i.e., $\pi^u_{kj} = \frac{exp(-||\mathbf{z}^{uy}_k - \mathbf{z}^{ut}_j||^2_2/\epsilon)}{\sum_{k=1}^{K}\sum_{j=1}^{J} exp(-||\mathbf{z}^{uy}_k - \mathbf{z}^{ut}_j||^2_2/\epsilon)}$.
- *Diagonal (Diag)* assumes k^{th} rating factor aligned with k^{th} text factor, i.e., $\pi^u_{kj} = 1/K$ if $k = j$, otherwise $\pi^u_{kj} = 0$. This approach is only applicable when $K = J$.

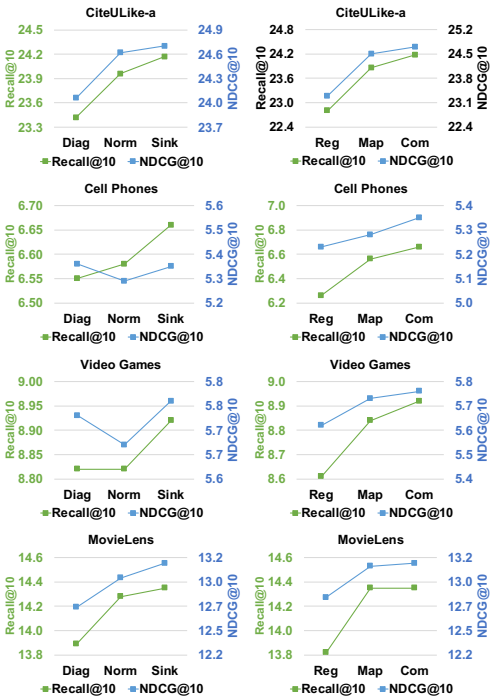


Figure 2: In each row, the left figure compares methods to derive π^u, i.e., *Diag*, *Norm*, *Sink*. The right figure contrasts methods, *Reg*, *Map*, *Def*, for transferring user interests.

First, Sinkhorn outperforms normalization across all datasets, as it converges to the optimal transport solution Peyré and Cuturi [2019], avoiding skewed alignment matrices that over-concentrate probabilities on highly similar pairs. Second, normalization generally matches or exceeds diagonal's accuracy, except for NDCG@10 on Cell Phones and Video Games, highlighting the importance of capturing pairwise alignments. In contrast, the diagonal approach's rigid one-to-one assumption leads to suboptimal performance, underscoring the value of probabilistic alignment.

Interest transfer method. Figure 2 reports recommendation accuracy w.r.t. three interest transfer methods.

- *Combination (Com)* includes both regularization and mapping & fusing inside BANDVAE.
- *Mapping & Fusing (Map)* only includes mapping and fusing for interest transfer (no regularization).
- *Regularization (Reg)* only involves regularization for interest transfer (no mapping and fusing).

First, regularization and mapping & fusing complement each other, with their combination achieving higher accuracy than either alone. Second, mapping & fusing has a stronger impact than regularization, highlighting the importance of bidirectional interest transfer between ratings and

Table 2: Recommendation performance comparison. The highest results are boldfaced while the runners-up are underlined. Units of reported numbers, which are averaged over ten runs, are percentage. * denotes statistical significance between the boldfaced and the underlined on a paired t-test with p-value $< 5 \times 10^{-2}$.

Model	CiteULike-a		Cell Phones		Video Games		MovieLens	
	R@10	N@10	R@10	N@10	R@10	N@10	R@10	N@10
MDCVAE	$22.43^{+7.8\%}$	$20.97^{+17.8\%}$	$4.34^{+53.5\%}$	$3.38^{+58.3\%}$	$5.90^{+51.2\%}$	$3.84^{+50.0\%}$	$14.03^{+2.4\%}$	$11.98^{+9.7\%}$
TopicVAE	$17.00^{+42.2\%}$	$17.54^{+40.8\%}$	$5.31^{+25.4\%}$	$4.23^{+26.5\%}$	$6.87^{+29.8\%}$	$4.37^{+31.8\%}$	$14.27^{+0.6\%}$	$13.01^{+1.0\%}$
RecVAE	$21.46^{+12.6\%}$	$22.43^{+10.1\%}$	$3.77^{+76.7\%}$	$2.92^{+83.2\%}$	$6.93^{+28.7\%}$	$4.33^{+33.0\%}$	**14.45**$^{-0.6\%}$	$13.02^{+0.9\%}$
MacridVAE	$21.92^{+10.3\%}$	$22.95^{+7.6\%}$	$5.82^{+14.4\%}$	$4.84^{+10.5\%}$	$7.95^{+12.2\%}$	$5.14^{+12.1\%}$	$14.25^{+0.8\%}$	$12.74^{+3.1\%}$
SEM-MacridVAE	$22.91^{+5.5\%}$	$23.85^{+3.6\%}$	$5.39^{+23.6\%}$	$4.32^{+23.8\%}$	$7.61^{+17.2\%}$	$4.87^{+18.3\%}$	$14.17^{+1.3\%}$	**13.36**$^{-1.6\%}$
ADDVAE	$23.44^{+3.1\%}$	$24.12^{+2.4\%}$	$5.76^{+15.6\%}$	$4.90^{+9.2\%}$	$8.09^{+10.3\%}$	$5.21^{+10.6\%}$	$14.01^{+2.5\%}$	$12.62^{+4.1\%}$
ELSA	$21.23^{+13.8\%}$	$22.25^{+11.0\%}$	$\underline{6.21}^{+7.2\%}$	$4.75^{+12.6\%}$	$7.39^{+20.7\%}$	$4.63^{+24.4\%}$	$13.35^{+7.6\%}$	$12.26^{+7.2\%}$
VALID	$22.50^{+7.4\%}$	$23.24^{+6.3\%}$	$6.18^{+7.8\%}$	$\underline{5.21}^{+2.7\%}$	$\underline{8.48}^{+5.2\%}$	$\underline{5.39}^{+6.9\%}$	$14.22^{+1.0\%}$	$12.99^{+1.2\%}$
FacetVAE	$\underline{23.53}^{+2.7\%}$	$\underline{24.68}^{+0.1\%}$	$5.52^{+20.7\%}$	$4.53^{+18.1\%}$	$7.64^{+16.8\%}$	$4.92^{+17.1\%}$	$13.97^{+2.8\%}$	$12.68^{+3.6\%}$
BANDVAE	**24.17***	**24.70**	**6.66***	**5.35***	**8.92***	**5.76***	$\underline{14.36}$	$\underline{13.14}$
p-value	2.2×10^{-4}	7.7×10^{-1}	6.6×10^{-5}	3.2×10^{-2}	6.6×10^{-4}	1.1×10^{-4}	4.4×10^{-1}	5.9×10^{-2}

Model	CiteULike-a		Cell Phones		Video Games		MovieLens	
	R@50	N@50	R@50	N@50	R@50	N@50	R@50	N@50
MDCVAE	$38.72^{+17.1\%}$	$26.45^{+17.0\%}$	$9.60^{+42.4\%}$	$4.87^{+50.9\%}$	$14.21^{+51.9\%}$	$5.98^{+50.5\%}$	$29.75^{+11.5\%}$	$18.13^{+10.8\%}$
TopicVAE	$37.78^{+20.1\%}$	$23.84^{+29.8\%}$	$11.59^{+17.9\%}$	$6.00^{+22.5\%}$	$17.94^{+20.3\%}$	$7.22^{+24.7\%}$	$31.90^{+4.0\%}$	$19.54^{+2.8\%}$
RecVAE	$38.39^{+18.2\%}$	$27.27^{+13.5\%}$	$8.79^{+55.5\%}$	$4.34^{+69.4\%}$	$18.02^{+19.8\%}$	$7.15^{+25.9\%}$	$32.78^{+1.2\%}$	$\underline{19.80}^{+1.5\%}$
MacridVAE	$43.00^{+5.5\%}$	$29.21^{+5.9\%}$	$11.96^{+14.3\%}$	$6.58^{+11.7\%}$	$20.02^{+7.8\%}$	$8.22^{+9.5\%}$	$32.28^{+2.8\%}$	$19.49^{+3.1\%}$
SEM-MacridVAE	$43.14^{+5.1\%}$	$29.87^{+3.6\%}$	$11.75^{+16.3\%}$	$6.12^{+20.1\%}$	$19.01^{+13.5\%}$	$7.80^{+15.4\%}$	$31.59^{+5.0\%}$	$19.77^{+1.6\%}$
ADDVAE	$\underline{43.89}^{+3.3\%}$	$30.23^{+2.3\%}$	$11.96^{+14.3\%}$	$6.65^{+10.5\%}$	$20.11^{+7.3\%}$	$8.30^{+8.4\%}$	$\underline{32.95}^{+0.7\%}$	$19.63^{+2.3\%}$
ELSA	$41.20^{+10.1\%}$	$28.32^{+9.3\%}$	$\underline{13.05}^{+4.8\%}$	$6.72^{+9.4\%}$	$19.91^{+8.4\%}$	$7.81^{+15.2\%}$	$31.95^{+3.8\%}$	$19.13^{+5.0\%}$
VALID	$43.44^{+4.4\%}$	$29.43^{+5.1\%}$	$12.61^{+8.4\%}$	$\underline{7.01}^{+4.9\%}$	$\underline{20.61}^{+4.7\%}$	$\underline{8.56}^{+5.10\%}$	$31.91^{+4.0\%}$	$19.60^{+2.5\%}$
FacetVAE	$43.85^{+3.4\%}$	$\underline{30.43}^{+1.7\%}$	$11.60^{+17.8\%}$	$6.25^{+17.6\%}$	$19.50^{+10.7\%}$	$7.96^{+13.1\%}$	$31.89^{+4.0\%}$	$19.33^{+3.9\%}$
BANDVAE	**45.36***	**30.94***	**13.67***	**7.35***	**21.58***	**9.00***	**33.18***	**20.09***
p-value	2.5×10^{-3}	7.8×10^{-3}	2.8×10^{-3}	2.3×10^{-4}	2.8×10^{-6}	1.1×10^{-8}	3.8×10^{-2}	3.2×10^{-2}

texts. Third, excluding regularization reduces BANDVAE's accuracy, confirming its role in enhancing performance.

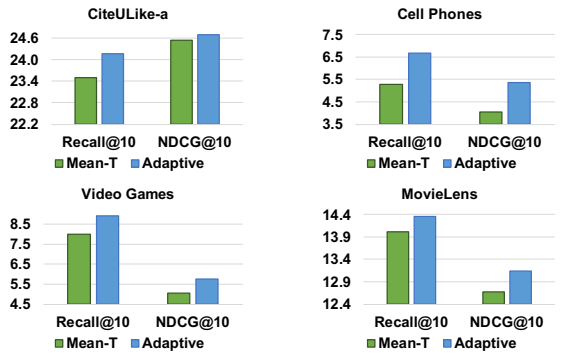


Figure 3: Fusion method comparison *Mean-T* vs. *Adaptive*.

Fusion method. Figure 3 compares our *adaptive fusion* method against *Mean-T* employed in Tran and Lauw [2022].

- *Adaptive* learns the adaptive fusion weight ρ^{uy} (and ρ^{ut}) for each user u as in Equations 7 and Equation 8.
- *Mean-T* (only applicable when $K = J$) computes the average[5] of two *transformed versions* of interest factors and sharing the final factors for both channels, i.e., $\tilde{\mathbf{z}}_k^{uy} = \frac{1}{2}(\hat{\mathbf{z}}_k^{uy} + \hat{\mathbf{z}}_k^{ut}) = \tilde{\mathbf{z}}_k^{ut}$ (k in place of j for text factors).

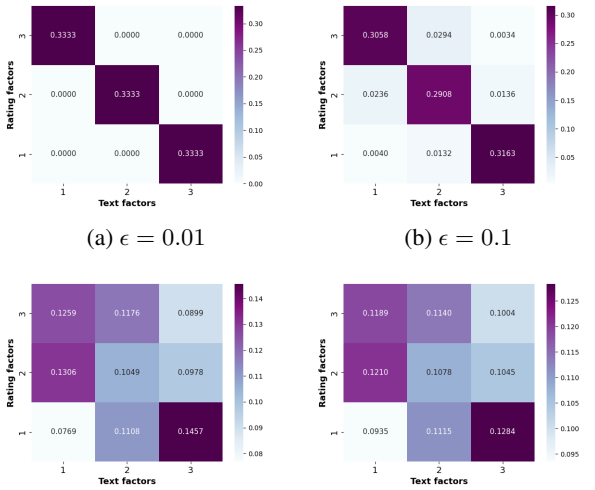
Our adaptive fusion outperforms Mean-T, demonstrating two key advantages. First, personalized fusion weights better capture user-specific preferences, as equal weights (Mean-T) fail to account for variability in how users weigh ratings versus texts. Second, Mean-T's shared representations are overly restrictive, while BANDVAE's flexible fusion effectively models preferences across modalities.

Table 3: BANDVAE's performance w.r.t. ϵ in Equation 3.

ϵ	CiteULike-a		Cell Phones		Video Games		MovieLens	
	R@10	N@10	R@10	N@10	R@10	N@10	R@10	N@10
0.01	23.78	24.36	6.57	5.31	**8.93**	**5.77**	13.90	12.62
0.02	23.79	24.47	6.52	5.26	8.89	5.75	13.92	12.58
0.1	**24.17**	**24.70**	6.60	5.28	8.88	5.68	14.10	12.74
0.2	24.09	24.66	6.50	5.23	8.81	5.64	14.13	12.83
1	24.12	24.70	6.61	5.28	8.79	5.65	14.35	13.14
2	24.11	24.65	**6.66**	**5.35**	8.79	5.66	**14.37**	**13.14**

Effect of ϵ in Algorithm 1. Table 3 shows the results. The effect of ϵ is data-dependent. On CiteULike-a, Cell Phones and MovieLens, $\epsilon \geq 0.1$ leads to higher recommendation

[5]In Tran and Lauw [2022], *sum* is used. We empirically found that *sum* and *mean* lead to similar results.



(a) $\epsilon = 0.01$ (b) $\epsilon = 0.1$

(c) $\epsilon = 1$ (d) $\epsilon = 2$

Figure 4: Example of alignment matrix π^u on Cell Phones.

Figure 5: Illustration of aligning interest factors. For rating factors, we show top 3 items with highest predicted scores, described by images and short descriptions. For text factors, we visualize top 10 words with highest predicted scores.

accuracy than smaller ones $\epsilon < 0.1$. Pertaining to Video Games, $\epsilon < 0.1$ generally results in higher accuracy. These observations imply that ϵ should be chosen carefully to produce favorable recommendation accuracy on each dataset.

Effect of ϵ on alignment matrix π^u. Theoretically, small ϵ results in sparse π^u while large ϵ leads to roughly uniform π^u. Figure 4 shows an illustrative example produced by BANDVAE, evidently confirming the theoretical influence of ϵ. To further verify, we report the entropy of $\pi^u \in \mathbb{R}^{K \times J}$

Table 4: Entropy of π^u w.r.t. ϵ in Equation 3.

Dataset	ϵ						
	0.01	0.02	0.1	0.2	1	2	5
CiteULike-a	**1.9156**	1.9743	1.9775	1.9812	1.9993	1.9998	2.0000
Cellphones	1.6391	1.7578	**1.6251**	1.7769	1.9854	1.9965	1.9995
Video Games	**1.5987**	1.8430	1.9931	1.9974	1.9999	2.0000	2.0000
MovieLens	**1.2336**	1.8749	1.8447	1.9059	1.9964	1.9991	1.9999

in Table 4. The reported numbers are averaged of row-wise and column-wise entropy of π^u. Evidently, small ϵ results in lower entropy values, indicating sparser alignment matrices.

Case study. To better understand the alignment process in BANDVAE, Figure 5 visualizes an example of decoder outputs corresponding to alignment matrix in Figure 4(a). The alignment matrix in Figure 4(a) reveals a staggered correspondence between interest factors from two modalities. For instance, rating interest factor 3 aligns with text interest factor 1. This observation is consistent with Figure 5, where top three items predicted by rating interest factor 3 include VR products, while text interest factor 1 includes relevant terms like *virtual, reality, vr, glasses*. Similar interpretations can be made for other factors. This showcases BANDVAE's ability to semantically align and interpret rating factors.

Table 5: Similarities between text factors w.r.t. λ_t.

Dataset	λ_t						
	0	0.1	0.2	0.5	1	2	5
CiteULike-a	0.7993	0.8878	0.9371	0.7089	0.5340	**0.3559**	0.4586
Cellphones	0.8690	0.4746	0.4799	0.4703	0.3466	0.0533	**0.0094**
Video Games	0.7873	0.8334	0.8402	0.8546	0.8266	0.2928	**0.1210**
MovieLens	0.4487	0.4431	0.4442	0.4535	0.4140	**0.1126**	0.4093

Effect of λ_t on interpretability. The text decoder's output, particularly the distinctiveness of top words for each text interest factor, enhances interpretability of the relationship between ratings and texts. Distinct words make it easier to infer the meaning behind each factor, providing insights into user preferences. To quantify this, we measure similarity between text interest factors as the fraction of shared words in their top 10 terms. For each user, we compute pairwise similarities and report the average over all users in Table 5. Table 5 shows that large λ_t results in low similarities between text interest factors, which facilitates the understanding of user interests with more distinctive top predicted words.

5 CONCLUSION

We propose a novel model BANDVAE to align disentangled user interests from ratings and texts. By treating interest factors as distributions, we frame their alignment as an optimal transport problem, enabling data-driven discovery of probabilistic correspondences. BANDVAE's novelty features two key mechanisms: (1) an alignment probability-guided regularization term and (2) a barycentric mapping strategy. These enable BANDVAE to integrate cross-modal interest signals, improving textual content-aware recommendation accuracy and offering interpretable alignment of user preferences.

Acknowledgements

Hady W. Lauw gratefully acknowledges the support by the Lee Kong Chian Fellowship awarded by Singapore Management University.

References

Ignacio Avas, Liesbeth Allein, Katrien Laenen, and Marie-Francine Moens. Align macridvae: Multimodal alignment for disentangled recommendations. In *European Conference on Information Retrieval*, pages 73–89, 2024.

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35:1798–1828, 2013.

Christopher P. Burgess, Irina Higgins, Arka Pal, Loïc Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in β-vae. *CoRR*, 2018. URL `http://arxiv.org/abs/1804.03599`.

Zongsheng Cao, Qianqian Xu, Zhiyong Yang, Yuan He, Xiaochun Cao, and Qingming Huang. OTKGE: Multimodal knowledge graph embeddings via optimal transport. In *NeurIPS*, pages 39090–39102, 2022.

Huiyuan Chen, Chin-Chia Michael Yeh, Fei Wang, and Hao Yang. Graph neural transport networks with non-local attentions for recommender systems. In *Proceedings of the ACM Web Conference 2022*, page 1955–1964, 2022.

Tian Qi Chen, Xuechen Li, Roger B. Grosse, and David Duvenaud. Isolating sources of disentanglement in variational autoencoders. *CoRR*, 2018. URL `http://arxiv.org/abs/1802.04942`.

Nicolas Courty, Rémi Flamary, and Devis Tuia. Domain adaptation with regularized optimal transport. In *Machine Learning and Knowledge Discovery in Databases*, page 274–289, 2014.

Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(9):1853–1865, 2017.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *NeurIPS*, volume 26, 2013.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171–4186, 2019.

Evgeny Frolov and Ivan Oseledets. Hybridsvd: when collaborative information is not enough. In *RecSys*, page 331–339, 2019.

Aude Genevay, Gabriel Peyre, and Marco Cuturi. Learning generative models with sinkhorn divergences. In *AISTATS*, pages 1608–1617, 2018.

Zhiqiang Guo, Guohui Li, Jianjun Li, and Huaicong Chen. Topicvae: Topic-aware disentanglement representation learning for enhanced recommendation. In *Proceedings of the 30th ACM MM*, page 511–520, 2022.

Zhiqiang Guo, Guohui Li, Jianjun Li, Chaoyang Wang, and Si Shi. Dualvae: Dual disentangled variational autoencoder for recommendation. In *SDM*, pages 571–579, 2024.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2017.

Yupeng Hou, Shanlei Mu, Wayne Xin Zhao, Yaliang Li, Bolin Ding, and Ji-Rong Wen. Towards universal sequence representation learning for recommender systems. In *KDD*, 2022.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *ICLR*, 2017.

Olivier Jeunen, Jan Van Balen, and Bart Goethals. Closed-form models for collaborative filtering with side-information. In *RecSys*, page 651–656, 2020.

Dong Hyun Kim, Chanyoung Park, Jinoh Oh, Sungyoung Lee, and Hwanjo Yu. Convolutional matrix factorization for document context-aware recommendation. In *RecSys*, pages 233–240, 2016.

Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In *ICML*, pages 2654–2663, 2018.

Xiaopeng Li and James She. Collaborative variational autoencoder for recommender systems. In *KDD*, pages 305–314, 2017.

Weiming Liu, Chaochao Chen, Xinting Liao, Mengling Hu, Jiajie Su, Yanchao Tan, and Fan Wang. User distribution mapping modelling with collaborative filtering for cross domain recommendation. In *The Web Conference*, page 334–343, 2024.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *ICML*, pages 4114–4124, 2019.

Chen Ma, Peng Kang, Bin Wu, Qinglong Wang, and Xue Liu. Gated attentive-autoencoder for content-aware recommendation. In *WSDM*, pages 519–527, 2019a.

Jianxin Ma, Chang Zhou, Peng Cui, Hongxia Yang, and Wenwu Zhu. Learning disentangled representations for recommendation. In *Advances in Neural Information Processing Systems*, volume 32, pages 5712–5723, 2019b.

Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. In *ICLR*, 2017.

Michaël Perrot, Nicolas Courty, Rémi Flamary, and Amaury Habrard. Mapping estimation for discrete optimal transport. In *NeurIPS*, pages 4197–4205, 2016.

Gabriel Peyré and Marco Cuturi. Computational optimal transport. *Found. Trends Mach. Learn.*, 11(5-6):355–607, 2019.

Shashank Rajput, Nikhil Mehta, Anima Singh, Raghunandan Hulikal Keshavan, Trung Vu, Lukasz Heldt, Lichan Hong, Yi Tay, Vinh Q. Tran, Jonah Samost, Maciej Kula, Ed H. Chi, and Mahesh Sathiamoorthy. Recommender systems with generative retrieval. In *Advances in Neural Information Processing Systems*, 2023.

Steffen Rendle. Factorization machines. In *The 10th IEEE ICDM*, pages 995–1000, 2010.

Michael E. Sander, Pierre Ablin, Mathieu Blondel, and Gabriel Peyré. Sinkformers: Transformers with doubly stochastic attention. In *AISTATS*, pages 3515–3530, 2022.

Ilya Shenbin, Anton Alekseev, Elena Tutubalina, Valentin Malykh, and Sergey I. Nikolenko. Recvae: A new variational autoencoder for top-n recommendations with implicit feedback. In *WSDM*, page 528–536, 2020.

Sidak Pal Singh and Martin Jaggi. Model fusion via optimal transport. In *NeurIPS*, 2020.

Nhu-Thuat Tran and Hady W. Lauw. Aligning dual disentangled user representations from ratings and textual content. In *KDD*, page 1798–1806, 2022.

Nhu-Thuat Tran and Hady W. Lauw. Multi-representation variational autoencoder via iterative latent attention and implicit differentiation. In *CIKM*, page 2462–2471, 2023.

Nhu Thuat Tran and Hady W. Lauw. Learning multi-faceted prototypical user interests. In *ICLR*, 2024.

Vojtěch Vančura, Rodrigo Alves, Petr Kasalický, and Pavel Kordík. Scalable linear shallow autoencoder for collaborative filtering. In *RecSys*, page 604–609, 2022.

Chong Wang and David M. Blei. Collaborative topic modeling for recommending scientific articles. In *KDD*, pages 448–456, 2011.

Hao Wang, Naiyan Wang, and Dit-Yan Yeung. Collaborative deep learning for recommender systems. In *KDD*, pages 1235–1244, 2015.

Xin Wang, Hong Chen, Yuwei Zhou, Jianxin Ma, and Wenwu Zhu. Disentangled representation learning for recommendation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(1):408–424, 2023a.

Xin Wang, Zirui Pan, Yuwei Zhou, Hong Chen, Chendi Ge, and Wenwu Zhu. Curriculum co-disentangled representation learning across multiple environments for social recommendation. In *ICML*, pages 36174–36192, 2023b.

Xiaobao Wu, Xinshuai Dong, Thong Nguyen, and Anh Tuan Luu. Effective neural topic modeling with embedding clustering regularization. In *International Conference on Machine Learning*, pages 37335–37357, 2023.

Yang Xu, Lei Zhu, Zhiyong Cheng, Jingjing Li, Zheng Zhang, and Huaxiang Zhang. Multi-modal discrete collaborative filtering for efficient cold-start recommendation. *IEEE Trans. Knowl. Data Eng.*, 35(1):741–755, 2023.

Shujian Zhang, Xinjie Fan, Huangjie Zheng, Korawat Tanwisuth, and Mingyuan Zhou. Alignment attention by matching key and query distributions. In *Advances in Neural Information Processing Systems*, pages 13444–13457, 2021.

Yan Zhang, David W Zhang, Simon Lacoste-Julien, Gertjan J Burghouts, and Cees GM Snoek. Unlocking slot attention by changing optimal transport costs. In *ICML*, volume 202, pages 41931–41951, 2023.

Yin Zhang, Ziwei Zhu, Yun He, and James Caverlee. Content-collaborative disentanglement representation learning for enhanced recommendation. In *RecSys*, page 43–52, 2020.

Wayne Xin Zhao, Junhua Chen, Pengfei Wang, Qi Gu, and Ji-Rong Wen. Revisiting alternative experimental settings for evaluating top-n item recommendation algorithms. In *CIKM*, page 2329–2332, 2020.

Xin Zhou and Chunyan Miao. Disentangled graph variational auto-encoder for multimodal recommendation with interpretability. *IEEE Trans. Multim.*, 26:7543–7554, 2024.

Xin Zhou, Hongyu Zhou, Yong Liu, Zhiwei Zeng, Chunyan Miao, Pengwei Wang, Yuan You, and Feijun Jiang. Bootstrap latent representations for multi-modal recommendation. In *The Web Conference*, page 845–854, 2023.

Yaochen Zhu and Zhenzhong Chen. Mutually-regularized dual collaborative variational auto-encoder for recommendation systems. In *TheWebConf.*, page 2379–2387, 2022.

Yaochen Zhu and Zhenzhong Chen. Variational bandwidth auto-encoder for hybrid recommender systems. *IEEE Trans. Knowl. Data Eng.*, 35(5):5371–5385, 2023.

Optimal Transport Alignment of User Preferences from Ratings and Texts

Nhu-Thuat Tran[1] **Hady W. Lauw**[1]

[1]School of Computing and Information Systems, Singapore Management University

A EXTENDED RELATED WORK

VAE-based disentangled representation learning. Uncovering hidden explanatory factors behind data results in robust representations and enables modeling complex patterns underlying data Bengio et al. [2013]. Variational AutoEncoder or VAE is a popular method offering representation disentanglement. Early works in this direction Higgins et al. [2017], Burgess et al. [2018], Kim and Mnih [2018], Chen et al. [2018], Locatello et al. [2019] focus on dimension-level disentanglement, where each element in the representation vector captures a distinctive latent feature. Later, Ma et al. [2019b], Tran and Lauw [2023, 2024], Guo et al. [2024] extend this line by disentangling user preferences not only at dimension level but also at intention level. Follow-up works incorporate various sources of information to improve disentangling user preferences. Tran and Lauw [2022], Guo et al. [2022] employs textual content while Wang et al. [2023a] hires visual information. Zhou and Miao [2024], Avas et al. [2024] seek the rich knowledge behind multi-modal data, i.e., textual and visual features. Wang et al. [2023b] integrates social relationships between users to better disentangle user preferences. Our work follows this line of research yet is distinctive in innovatively incorporating *optimal transport* for aligning disentangled rating and text factors. While we mainly focus on rating and text data in this work, the proposed method is applicable when multi-modalities involve as elaborated in Section 3.5.

Textual content-aware recommendation. Early methods Wang and Blei [2011], Wang et al. [2015], Kim et al. [2016], Ma et al. [2019a] leverage deep neural networks to model item textual content, thereby enhancing recommendation performance. Later, VAE has been widely adopted for this task, both in non-disentangled Zhu and Chen [2023], Li and She [2017], Zhu and Chen [2022] and disentangled fashions Zhang et al. [2020], Tran and Lauw [2022], Guo et al. [2022]. What distinguishes our work from these is the introduction of an optimal transport (OT)-based approach to align and fuse interest factors from ratings and textual content, which provides a more flexible and nuanced alignment between interest factors. Recently, pre-trained language models (PLMs), e.g., Devlin et al. [2019], have been explored to generate text-based item representations for recommendation Hou et al. [2022], Zhou et al. [2023], Rajput et al. [2023]. While PLMs offer powerful text encodings, they tend to compress the entire content into a single vector, which ignores the intricate structure and multi-faceted nature of textual data. In contrast, our work focuses on disentangling multiple interest factors from textual content to capture a richer representation of user preferences. Thus, we leave the integration of PLMs into our framework as a direction for future work. Zhou and Miao [2024], Avas et al. [2024] leverage textual and visual data for recommendation tasks, which differs significantly from ours, particularly in their use of multi-modal features. As a result, this work is not not directly comparable with our model, which focuses solely on aligning ratings and textual content. Additionally, our work is related to hybrid recommender systems Rendle [2010], Frolov and Oseledets [2019], Jeunen et al. [2020], Xu et al. [2023], which aim to tackle challenges like the cold-start problem by combining multiple data sources. However, our primary objective in this paper is to discover and align multiple interest factors across modalities in a warm-start setting, where sufficient interaction data is available. Addressing the cold-start problem, though relevant, falls outside the scope of this work.

Optimal transport and its applications. Optimal Transport (OT) offers an elegant framework to measure the distance between two probability distributions and facilitates the transformation of points from one distribution to another Peyré and Cuturi [2019]. The popular method for computing optimal transport plan is Sinkhorn algorithm Cuturi [2013], Genevay et al. [2018], which offers efficient and GPU-friendly framework and thus, has enabled numerous applications across various

domains. For example, OT has been utilized in domain adaptation Courty et al. [2014, 2017], model fusion Singh and Jaggi [2020] and attention-based models Zhang et al. [2021], Sander et al. [2022]. Additionally, OT has demonstrated its effectiveness in fusing multi-modal knowledge graph data Cao et al. [2022], enhancing the coherence of topic modeling via regularization Wu et al. [2023], and improving object-centric learning Zhang et al. [2023]. In recommender systems, OT has also been widely explored, e.g., aggregating non-local information in graph-based recommendation Chen et al. [2022], or finding user correspondence in cross-domain recommendation setting Liu et al. [2024]. Our work builds on OT but adopts an orthogonal approach. Specifically, we apply OT to align and fuse mutually disentangled user interest factors derived from ratings and textual content. By leveraging OT to perform this alignment in a data-driven manner, our approach allows for a more flexible and personalized representation of user preferences across modalities, leading to higher recommendation accuracy and offering a feasible method to gain insights into the relationship between user interactions and textual content.

B TEXT ENCODER $\mathcal{E}^t$

Text encoder $\boldsymbol{\mathcal{E}}^t$ functions similarly to the rating encoder $\mathcal{E}^y$, but its input is the textual content $\mathbf{t}^u$ associated with user u. $\mathcal{E}^t$ also leverages prototypes, $\mathbf{m}^t \in \mathbb{R}^{J\times d}$, to cluster words into J groups, each representing one user interest from texts. In general, this process can also run iteratively for L^t iterations. However, we empirically found that employing iteratively clustering process inside $\mathcal{E}^t$, i.e., $L^t > 1$, does not show clear improvement. Thus, to maintain efficiency, we set $L^t = 1$. As such, there is no prototype updating inside $\mathcal{E}^t$ and the set of text prototypes $\mathbf{m}^t$ is shared among users. Then, we calculate the word-cluster assignment matrix $\mathbf{A}^{ut} \in \mathbb{R}^{W\times J}$

$$\mathbf{A}^{ut} = \eta\left(\frac{\mathbf{E}\cdot(\mathbf{m}^{ut})^T}{\tau\cdot||\mathbf{E}||_2\cdot||\mathbf{m}^{ut}||_2}\right) \tag{13}$$

Similar to rating encoder $\boldsymbol{\mathcal{E}}^y$, η is Gumbel-Softmax. Next, we estimate two parameters of Gaussian distribution for text interest factor j as $\boldsymbol{\mu}_j^{ut} = \frac{\mathbf{r}_j^{ut}}{||\mathbf{r}_j^{ut}||_2}$, $\boldsymbol{\sigma}_j^{ut} = \sigma^t \cdot exp(-\frac{1}{2}\mathbf{o}_j^{ut})$ where

$$(\mathbf{r}_j^{ut}, \mathbf{o}_j^{ut}) = \mathbf{W}_2' tanh(\mathbf{W}_1' norm(\mathbf{A}_{:j}^{ut} \odot \mathbf{t}^u) + \mathbf{b}_1') + \mathbf{b}_2' \tag{14}$$

$\odot$ and $norm(\cdot)$ are the same as in rating encoder. $\mathbf{W}_1' \in \mathbb{R}^{W\times D}, \mathbf{b}_1' \in \mathbb{R}^D, \mathbf{W}_2' \in \mathbb{R}^{D\times 2d}, \mathbf{b}_2' \in \mathbb{R}^{2d}$ are weight matrices and bias vectors of text encoder. σ^t's value is around 0.1. Then j^{th} text factor is sampled as $\mathbf{z}_j^{ut} \sim \mathcal{N}(\boldsymbol{\mu}_j^{ut}, [diag(\boldsymbol{\sigma}_j^{ut})]^2)$, which is repeated $\forall j = 1, 2, ..., J$. Assuming the independence between text factors of user u, we have $q(\mathbf{z}^{ut}|\mathbf{t}^u, \mathbf{A}^{ut}) = \prod_{j=1}^J \mathcal{N}(\boldsymbol{\mu}_j^{ut}, [diag(\boldsymbol{\sigma}_j^{ut})]^2)$ as the variational distribution, which is then aligned with prior distribution $p(\mathbf{z}^{ut}) = \mathcal{N}(\mathbf{0}, (\sigma^t)^2\mathbf{I})$ via Kullback-Leibler divergence (D_{KL}^t) to impose micro-disentanglement.

In summary, text encoder $\mathcal{E}^t$ produces J text interest factors $\mathbf{z}^{ut} = \{\mathbf{z}_j^{ut}\}_{j=1}^J$, assignment matrix $\mathbf{A}^{ut}$, and regularization term $D_{KL}^t(q(\mathbf{z}^{ut}|\mathbf{t}^u, \mathbf{A}^{ut})||p(\mathbf{z}^{ut}))$.

C ADDITIONAL RESULTS

Analysis on the Number of Interest Factors. We report recommendation accuracy w.r.t. the numbers of rating factors K and the numbers of text factors J in Table 6 and Table 7, respectively. *For rating factors*, setting $K = 3$ or $K = 4$ gives the best accuracy on CiteULike-a while $K \geq 5$ is ideal for Cell Phones. These evidences show that users have varied interests. BANDVAE generally performs best on Video Games with $K \geq 4$ while reaches its peak performance on MovieLens with $K = 4$. *Pertaining to text factors*, setting $J = 4$ results in the higher accuracy on CiteULike-a and MovieLens. In contrast, BANDVAE's performance on Cell Phones is not highly sensitive to J. For Video Games, setting $J \leq 4$ produces better accuracy than larger values.

It is worth to note that thanks to the pair-wise alignment between interest factors, BANDVAE can accommodate users' distinctive behaviors across modalities, i.e., when K and J differ, while baseline such as Tran and Lauw [2022] cannot. As a result, BANDVAE offers greater flexibility and is more applicable.

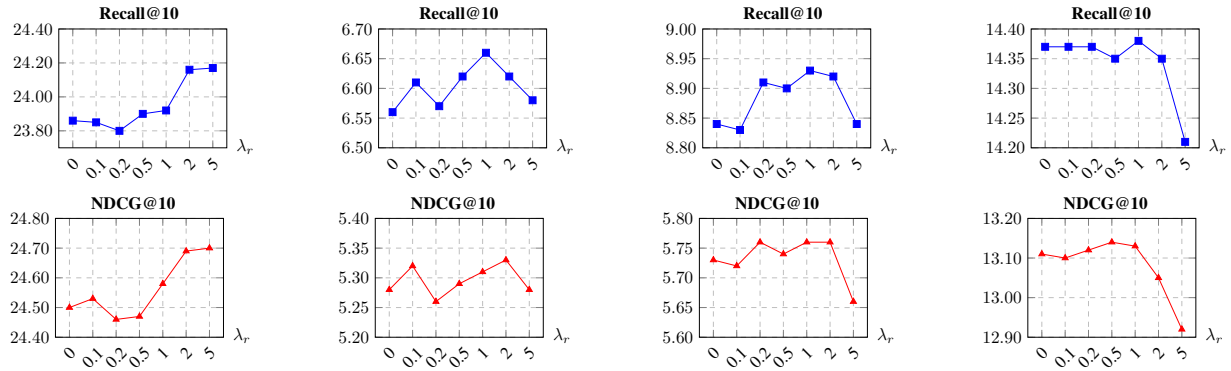
Effect of λ_r. Figure 6 presents BANDVAE's accuracy w.r.t. λ_r, which controls the effect of regularization term for interest transfer between rating and text factors. First, we observe that setting λ_r to 1 or 0.5 results in higher accuracy on chosen datasets. Second, the effect of λ_r is data-dependent, e.g., while CiteULike-a favors large λ_r, the remaining datasets requires smaller value, i.e., around 0.5 and 1. An excessive value of λ_r might cause detrimental effect.

Table 6: BANDVAE's performance w.r.t. the number of rating factors K.

K	CiteULike-a		Cell Phones		Video Games		MovieLens	
	R@10	N@10	R@10	N@10	R@10	N@10	R@10	N@10
2	23.84	24.54	5.93	4.61	8.37	5.33	14.13	12.74
3	**24.18**	**24.77**	6.35	4.97	8.74	5.58	14.08	12.82
4	24.17	24.70	6.66	5.35	8.92	5.76	**14.35**	**13.14**
5	23.77	24.37	6.69	5.51	9.02	5.80	13.97	12.62
6	23.87	24.39	6.92	5.74	**9.03**	5.83	14.11	12.83
7	23.84	24.32	**6.94**	5.86	8.96	**5.85**	13.95	12.70
8	23.86	24.29	6.85	**5.87**	8.95	5.82	14.07	12.79

Table 7: BANDVAE's performance w.r.t. the number of text factors J.

J	CiteULike-a		Cell Phones		Video Games		MovieLens	
	R@10	N@10	R@10	N@10	R@10	N@10	R@10	N@10
2	23.95	24.58	6.53	5.20	8.87	5.70	14.17	12.92
3	24.15	24.57	6.56	5.26	**8.98**	**5.77**	14.04	12.81
4	**24.17**	**24.70**	**6.66**	**5.35**	8.92	5.76	**14.35**	**13.14**
5	24.04	24.52	6.60	5.28	8.94	5.73	14.05	12.67
6	23.88	24.34	6.59	5.33	8.85	5.68	13.78	12.50
7	23.86	24.31	6.48	5.23	8.86	5.68	14.00	12.67
8	24.05	24.49	6.50	5.21	8.77	5.60	13.95	12.65



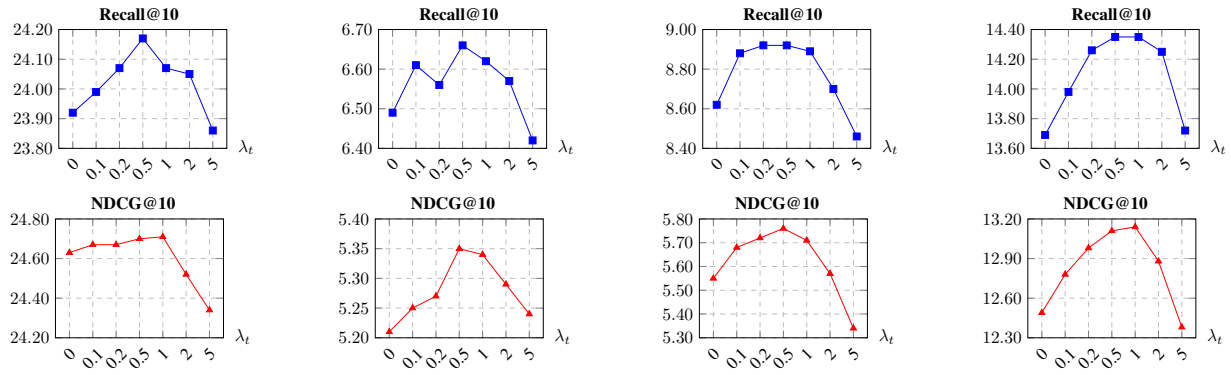
(a) Results on CiteULike-a (b) Results on Cell Phones (c) Results on Video Games (d) Results on MovieLens

Figure 6: BANDVAE's performance w.r.t. various of λ_r.

Effect of λ_t. Figure 7 presents the influence of λ_t, which controls the effect of text reconstruction objective, on BANDVAE's accuracy. First, setting $\lambda_t > 0$ leads to higher accuracy than setting $\lambda_t = 0$, underscoring the benefit of textual signals. Second, setting λ_t to a moderate value, i.e., around 0.5 or 1, results in favorable accuracy across datasets.

Efficiency Analysis. Table 8 analyzes the efficiency of BANDVAE and two strongest baselines ADDVAE and VALID. For each model, we record the training time per epoch (in second) (averaged over ten runs) and the memory required for training (in GB). There are three key takeaways. First, BANDVAE maintains a comparable efficiency level yet achieves higher recommendation accuracy than ADDVAE and VALID. Second, the training time and memory gaps between VALID and BANDVAE come from textual content modeling component, i.e., text channel, in BANDVAE yet do not appear in VALID. Third, despite both including a textual content modeling module, BANDVAE employs multiple prototype updates in rating encoder while ADDVAE does not, which results in difference in efficiency level of these two models. Though BANDVAE introduces added complexity, this is the cost of modeling richer, user-aware cross-modal relationships. As shown in Tables 2, this complexity leads to clear performance gains.

Effect of L^y. Table 9 shows BANDVAE's recommendation accuracy w.r.t. L^y, the number iterations to update rating prototypes in rating encoder $\mathcal{E}^y$. Evidently, using more than one prototype update steps leads to higher recommendation



(a) Results on CiteULike-a (b) Results on Cell Phones (c) Results on Video Games (d) Results on MovieLens

Figure 7: BANDVAE's performance w.r.t. various of λ_t.

Table 8: Efficiency comparison between BANDVAE, ADDVAE and VALID w.r.t. running time (seconds per training epoch) and memory required for training (measured in GB).

Model	CiteULike-a		MovieLens		Cell Phones		Video Games	
	Time (s)	Mem (GB)	Time (s)	Mem (GB)	Time (s)	Mem (GB)	Time (s)	Mem (GB)
ADDVAE	1.82	2.48	3.26	1.92	6.08	2.38	10.82	2.44
VALID	1.51	2.96	2.09	1.99	6.20	4.58	9.33	2.89
BANDVAE	1.95	3.39	3.68	2.52	8.38	4.89	14.68	3.38

Table 9: BANDVAE's performance w.r.t. the number iterations L^y to update rating prototypes.

L^y	CiteULike-a		Cell Phones		Video Games		MovieLens	
	R@10	N@10	R@10	N@10	R@10	N@10	R@10	N@10
1	23.62	24.33	6.19	5.03	8.41	5.42	14.12	12.84
2	24.17	**24.70**	6.39	5.12	8.92	5.76	14.09	12.75
3	**24.20**	24.57	6.51	5.22	**8.95**	**5.78**	**14.35**	**13.14**
4	23.78	24.28	**6.66**	**5.35**	8.79	5.68	14.08	12.90

accuracy in all chosen datasets, which confirms the effectiveness of updating prototypes in rating encoder. This finding is consistent with Tran and Lauw [2023]. Moreover, each data requires a specific value of L^y to achieve favorable accuracy, e.g., 2 on CiteULike-a, 4 on Cell Phones and 3 on Video Games and MovieLens.

Table 10: BANDVAE's performance w.r.t. the number iterations L^t to update text prototypes.

L^t	CiteULike-a		Cell Phones		Video Games		MovieLens	
	R@10	N@10	R@10	N@10	R@10	N@10	R@10	N@10
1	**24.17**	**24.70**	**6.66**	**5.35**	**8.92**	**5.76**	**14.35**	**13.14**
2	24.14	24.63	6.56	5.31	8.86	5.67	13.92	12.72
3	23.84	24.50	6.63	5.34	8.85	5.62	13.90	12.66
4	23.59	24.27	6.59	5.31	8.68	5.58	13.78	12.52

Effect of L^t. Table 10 presents the influence of the number iterations L^t to update text prototypes in text encoder $\boldsymbol{\mathcal{E}}^t$. The effect of L^t is contrary to that of L^y, i.e., using more prototype update steps results in a reduction in BANDVAE's performance. We conjecture that as users' comprehensions of textual content, i.e., words and phrases, are roughly the same, and thus, imposing personalization into word clustering in text encoder via setting $L^t > 1$ causes a detrimental effect. As such, we fix $L^t = 1$ for all datasets through out the paper to maintain both effectiveness and efficiency.