Hybrid Item-Item Recommendation via Semi-Parametric Embedding

Peng Hu*, Rong Du, Yao Hu and Nan Li
Alibaba Group, Hang Zhou, China
{spyr.hp, qingzhao.dr, yaoohu, nanli.ln}@alibaba-inc.com

Abstract

Nowadays, item-item recommendation plays an important role in modern recommender systems. Traditionally, this is either solved by behavior-based collaborative filtering or content-based methods. However, both kinds of methods often suffer from cold-start problems, or poor performance due to few behavior supervision; and hybrid methods which can leverage the strength of both kinds of methods are needed. In this paper, we propose a semi-parametric embedding framework for this problem. Specifically, the embedding of an item is composed of two parts, i.e., the parametric part from content information and the non-parametric part designed to encode behavior information; moreover, a deep learning algorithm is proposed to learn two parts simultaneously. Extensive experiments on real-world datasets demonstrate the effectiveness and robustness of the proposed method.

1 Introduction

Nowadays, with data explosion over internet, modern recommender systems are playing more and more important role in many domains, such as e-commerce, news articles, digital entertainment, social network, etc.

In these recommender systems, item-item (I2I) recommendation which deal with the problem of recommending related items for a given item is one of the fundamental topics. This is because, on one hand, I2I itself has direct applications scenarios, such as “you may also like” on e-commerce platforms; and on the other hand, the system can recommend to a user with similar items to those have been clicked [Sarwar et al., 2001].

In the literature, many methods have been developed for this topic, and they generally fall into two groups: collaborative filtering based methods and content based methods [Shi et al., 2014]. Generally speaking, collaborative filtering based methods are based on the assumption that similar users prefer similar items or that one user expresses similar preferences for similar items [Mnih and Salakhutdinov, 2007; Salakhutdinov and Mnih, 2008; Yu et al., 2009]. For I2I recommendation, collaborative filtering methods usually group correlations between items by their past common behavior actions (e.g., user’s click in e-commerce). When there are sufficient user behaviors, collaborative filtering methods are generally preferred due to their good performance. However, their performance can be very poor when the behavior information is sparse [Linden et al., 2003]. Moreover, they also suffer from the cold-start problem, i.e., they can not do recommendations for new items that have never seen before. Meanwhile, content-based methods [Pazzani and Billsus, 2007; Lops et al., 2011] are based on the assumption that similar items will be preferred by similar users, and they compute the similarity between items based on their content directly. Apparently, content-based methods do not depend on user behaviors and thus do not suffer from the cold-start problem. However, their performance is generally not superior to collaborative filtering based methods when there are lots of behavior information; and it is even hard to define a good similarity measure in practice.

To deal with those problems, hybrid methods [Singh and Gordon, 2008; Nickel et al., 2011; Wang and Blei, 2011; Lian et al., 2017] which combine collaborative filtering and content-based methods have been proposed in recent years. Generally speaking, those methods integrated auxiliary content information into collaborative filtering to learn effective latent factors. Especially, with deep learning based embedding methods developed last years, it is much easier to have effective representations for each item. However, existing hybrid methods either infer latent representation for items or directly use deep version of collaborative filtering with context information. Few attempts have been made to develop effective item representations which explicitly balances behavior and content information.

In this paper, we develop a semi-parametric embedding (SPE) framework for hybrid item-item recommendation. In particular, the embedding of an item is composed of two parts in this framework: the parametric part which is built from content information and the non-parametric part which is designed to encode behavior information, especially the behavior information not encoded by the content. Furthermore, a deep learning algorithm is proposed to learn two parts simultaneously, such that not only the items with rich behavior information can be well represented, the items with few behav-
ior information (i.e., cold-start items) can also be well represented. We conduct experiments on real-world datasets, and experimental results show the effectiveness of the proposed method, especially for the cold-start items. Main contributions of this paper can be summarized as follows:

- We propose a semi-parametric embedding framework (named as SPE) for hybrid item-item recommendation. It takes both the behavior and content information into account, and is able to capture the context similarity and implicit relationship between items.
- The proposed SPE framework is a tightly coupled hybrid model. That is, behavior information can guide the learning of context-based feature mapping, and also the content representation can further improve the performance of behavior based collaborative filtering, especially on cold-start items.
- SPE explicitly constructs the representation of each item by two component vectors and balances the two components automatically to get a robust recommendation for both cold-start and behavior-rich items.
- We conduct extensive experiments on real-world datasets to demonstrate the effectiveness and robustness, and then analyze experimental results comprehensively.

2 Related Work

In the literature, there are lots of works on recommendation systems, ranging from content-based methods [Lang, 1995; Mooney and Roy, 2000; Pazzani and Billsus, 2007; Lops et al., 2011] to CF-based ones [Mnih and Salakhutdinov, 2007; Salakhutdinov and Mnih, 2008; Yu et al., 2009]. While both methods have their own advantages, individually they fail to provide good recommendations in many situations [Melville et al., 2002; Kim et al., 2006]. Content-based methods utilize item’s content information for recommendation and thus get rid of cold-start problem. However, it’s often cost-expensive and impractical to design and collect powerful context information feature to achieve a good result. On the other hand, CF-based methods often present more impressive recommender result, especially for behavior-rich items, but their performance drops rapidly when coming with cold-start problem due to the strong dependency on intersections with items.

Hence hybrid methods have gained popularity in recent years, which integrate context with behavior information to deal with those problems. Hybrid methods can be further divided into two categories: loosely and tightly coupled methods. Loosely coupled methods [Melville et al., 2002; Sevil et al., 2010] make the use of auxiliary information to provide features for CF but the rating information cannot guide the learning of features. On the contrary, tightly coupled methods [Wang and Blei, 2011; Wang et al., 2015] provide two-way interaction between them and often outperform loosely coupled ones. However, these existed hybrid methods are often not effective especially when the rating matrix and side information are very sparse [Agarwal et al., 2011].

Recently, deep learning may be the most powerful methods to learn effective representations [Hinton and Salakhutdinov, 2006; Hinton et al., 2006]. Thereby, some researches make use of deep learning to get powerful representation for recommendation systems and outperform the traditional ones. [Salakhutdinov et al., 2007] employs Restricted Boltzmann Machines (RBM) to perform collaborative filtering. [Van den Oord et al., 2013] and [Wang and Wang, 2014] are content-based methods and directly use convolutional neural network (CNN) or deep belief network (DBN) to obtain latent factors for content information. Combining deep learning with hybrid methods, [Wang et al., 2015] present a bayesian deep learning stacked Denoising AutoEncoder (sDAE) model to couple content information and ratings (feedback) matrix. [Dong et al., 2017] develops sDAE to a new deep learning model additional-sDAE (aSDAE) and integrate side information into latent factors to recommend. Most of them just extract a deep feature representation by mixing content and rating matrix to capture the similarity and implicit relationship between items at the same time. Very few attempts have been made to balance context and behavior information for different items, varying from cold-start to behavior-rich ones.

3 Preliminaries

In this section, we start with hybrid I2I recommendation formulation of the task discussed above, and then briefly review two technical methods used in our paper, matrix factorization and stacked denoising autoencoders.

3.1 Problem Definition

Similar to some existed hybrid methods, this paper takes implicit feedback as training and testing data to complete the recommendation task. In standard I2I recommendation setting, there are n items \( \{x_i\}_{i=1}^n \), and a symmetric binary matrix \( R \in \mathbb{R}^{n \times n} \). Each entry \( r_{ij} \) of \( R \) corresponds to whether there is co-occurrence between item \( x_i \) and \( x_j \). \( R \) is partially observed which means \( r_{ij} = 0 \) can represent either a negative relationship or just an unobserved value. Particularly, \( \forall i = j, r_{ij} = 1 \). Suppose there also exists context information matrix \( C \in \mathbb{R}^{n \times p} \), where each row vector \( c_i \) with size \( p \) represents item \( x_i \)’s meta information vector. Moreover, item’s statistics feature vector \( b_i \) with size \( q \), such as action frequency that item \( x_i \) has been showed, clicked and bought, constructs matrix \( B \in \mathbb{R}^{n \times q} \). Given \( R \), \( C \) and \( B \), we aim to predict the missing ratings in \( R \) for recommendation.

3.2 Matrix Factorization

Matrix Factorization(MF) [Koren et al., 2009] has been popular in Collaborative Filtering based recommender system due to its performance and scalability. Generally, MF methods represent user and item with dense low-dimension vectors in a shared latent subspace, noted as \( u_i \), \( v_i \) respectively. Under low-rank rating matrix assumption, the rate is modeled as inner product of the latent factor vectors \( u_i \) and \( v_i \) in traditional user-item recommender systems,

\[
   r_{ij} = u_i^T v_j. \tag{1}
\]

Bias can be added to enrich the model [Koren et al., 2009].
The objective function of matrix factorization is to minimize a regularized squared error loss,
\[ L = \arg \min_{U,V} \sum_{i,j} \| r_{ij} - u_i^T v_j \|^2 + \lambda (\| u_i \|^2 + \| v_j \|^2), \]
where \( \| \cdot \| \) is an indicator function that equals to 1 if \( r_{ij} \) is observed, otherwise 0, \( \lambda \) is the regularization parameter to alleviate model overfitting.

### 3.3 Stacked Denoising Autoencoders
Autoencoder is a type of deep learning model used to encode data in an unsupervised manner. Denoising autoencoder replaces autoencoder’s input with a corrupted version by introducing noise to reconstruct the original clean input and is usually more robust. Due to its characteristic of feature extraction, there exist researchs [Wang et al., 2015; Dong et al., 2017] that employ stacked Denoising Autoencoder(sDAE) [Vincent et al., 2010] to encode auxiliary side information for better performance of recommendation due to its characteristic of feature extraction.

Figure 1(a) illustrates a simple sDAE with layers \( L = 4 \). Usually the middle hidden layer of sDAE is designed as a bottleneck for the purpose of feature compression and extraction. To update the parameters, sDAE usually solves the following optimization problem:

\[ L = \arg \min_{W} \| X_e - X_L \|_F^2 + \lambda \sum_{l} \| W_l \|_F^2, \]

where \( X_e \) is the noise corrupted version of original feature input \( X_L \), \( \lambda \) is regularization parameter and \( W \) is weight parameters of sDAE.

### 4 Semi-Parametric Embedding
To deal with cold-start items with good performance on behavior-rich items, we introduce deep learning and present a tightly coupled hybrid methods which model item’s representation as integration of parametric and non-parametric components. In this section, details about semi-parametric embedding model are described, then we apply it to the I2I recommendation problem.

#### 4.1 The Framework
Item’s id \( i \) is ignored for simplicity here. In this paper, taking item context information into consideration for dealing with cold-start problem, we model item latent factor \( v \in \mathbb{R}^k \) as integration of behavior and context components, noted as \( z \) and \( e \) respectively,

\[ v = \delta z + (1 - \delta) e, \]

where \( e \) is the parametric encoded context representation from side information \( e \), which means,

\[ e = g(c), \]

multilayer perceptron with nonlinear activation function is often selected to achieve nonlinear generalization ability for \( g \). \( \delta \) is a self-learned weight that controls the proportion between non-parametric part \( z \) and parametric part \( e \). \( \delta \) determines how much confidence our model pay on behavior information \( e \) and can be got with a mapping function from item’s statistical feature \( b \). Here we adopt linear regression for simplicity,

\[ \delta = \sigma (w \cdot b), \]

where \( \sigma \) is a sigmoid function that make sure \( \delta \) value ranges from 0 to 1.

\( z \in \mathbb{R}^k \) is a unique vector for each item \( x \), it is the agnostic non-parametric part of item representation; \( e \in \mathbb{R}^k \) is transferred from \( c \) with a mapping function \( g : \mathbb{R}^p \to \mathbb{R}^k \) and \( e \) is the parametric part. We refer the integration of non-parametric part \( z \) and parametric part \( e \) as semi-parametric embedding framework.

As stated above, Figure 1(b) is a brief illustration of SPE. For items with sufficient behaviors, the non-parametric component dominates the semi-parametric representation with a high weight by the confidence weight value \( \delta \); for cold-start items, the non-parametric embedding is unreliable and the parametric component helps to depict the item with help of the meta-context information. Particularly for completely new coming items, \( \delta = 0 \) and item’s embedding \( v \) degenerates to \( e \).

#### 4.2 SPE for Hybrid I2I
Details about instantiation of SPE model in I2I recommender system are described in this part.

As most distance metric researches, similarity score between item \( v_i \) and \( v_j \) can be defined as

\[ r_{ij} = s(v_i, v_j), \]

where \( s \) is usually a symmetric distance metric.

We choose the similarity score function \( s(\cdot) \) in Eq. 7 as inner product following most existed recommender approaches for fair, and then we rescale similarity between items by mapping into range of \([0, 1]\) with a sigmoid function, that is,

\[ s(v_i, v_j) = \frac{1}{1 + e^{-v_i \cdot v_j}}. \]

For the characteristic of implicit feedback in I2I problems, we follow the previous work [He et al., 2017] and use cross-entropy loss to evaluate empirical risk. Specifically, we label all \( r_{ij} = 1 \) as positive samples and sample randomly from \( r_{ij} = 0 \) which denotes negative or unobserved rates in \( \mathbb{R} \) uniformly as negative samples. The positive and negative examples are denoted as \( \mathcal{P} \) and \( \mathcal{N} \) respectively, and the negative sampling ratio is set to 3 in the verification experiments, which means \(|\mathcal{N}| = 3|\mathcal{P}|\).
Optimization
As stated above, semi-parametric embedding model in I2I recommender system can be solved by minimizing the objective function as follow:

\[
\arg \min_{(i,j) \in \{P,N\}} \mathcal{L}(r_{ij}, s(v_i, v_j)) + \lambda_z \|z\|^2 + \lambda_g \|g(\cdot)\|^2,
\]

where \(s(v_i, v_j)\) is from Eq. 8, \(\mathcal{L}\) is defined as the binary cross entropy loss in our experiments, and \(\lambda_z, \lambda_g\) are trade-off weights for regularization of non-parametric embedding \(z\) and the encoding function \(g(\cdot)\) respectively. Best value for \(\lambda_z, \lambda_g\) can be found by grid search.

Prediction
After latent factor vectors for behavior information and context information are learned, the weight \(\delta\) and thus trade off the two components. Similarity rating between items can be calculated by Eq. 4 & 8 and a list of ranked items is generated for each item based on these similarity scores. For completely new coming items, their behavior vector \(z\) is unreliable and thus the item’s representation embedding degenerates as \(v = e\).

5 Experiments
The experimental section is organized as follows. We first describe datasets, evaluation metric and baseline algorithms used to verify the superiority of SPE for I2I problems. Then detail results are summarized and that is followed by a section to get deeper insights into the idea about how SPE model balances the parametric and non-parametric part adaptively. Finally, we show robust results of the extended model.

5.1 Datasets
The algorithms are evaluated on three real-world datasets: Amazon product data [McAuley et al., 2015] (Specifically the musical instruments subset), Yelp\(^1\) and a private dataset collected from Alibaba’s second-hand e-commerce platform. For Alibaba private dataset, we collect items from one of the most popular categories for the convenience of calculation. As Amazon product data and Yelp originally contain user-item interactions data, we follow [Grbovic et al., 2015] and convert the dataset into relationships between items for I2I recommendation. We assume that those items associated with a same user are related,

\[
r_{ij} = \max_k \mathbb{I}_k(v_i, v_j)
\]

where \(\mathbb{I}_k(\cdot)\) is an indicator function. \(\mathbb{I}_k(v_i, v_j) = 1\) if both of \(v_i\) and \(v_j\) have been rated, commented or clicked by user \(u_k\), otherwise \(\mathbb{I}_k(v_i, v_j) = 0\).

Other details for each dataset are listed as follows. Product title words for Amazon product data are collected as its Bag-Of-Words (BOW) features. User-generated comments tags of each business are used as its BOW features for Yelp. For Alibaba private second-hand dataset, segmented title words of item are grouped as BOW features. It is worth mentioning that the cold-start problem is more critical on the second-hand trading platform than traditional e-commerce platform because of the difficulty of behavior logs accumulation. On the other hand, amateur sellers on second-hand platform hardly have as strong intentions and skills to lead a deal as professional sellers. Thus item’s context information published on Alibaba second-hand dataset is often missed or even corrupted.

The basic statistics of each dataset are shown in Table 2. For more comprehensive verification, we construct additional experiments on corrupt versions of each dataset by randomly setting entities of item’s BOW features to 0 with a probability of \(\alpha\), named as “corrupt-rate” in our experiments. if not specified, \(\alpha\) is set to 0.3 by default.

5.2 Evaluation Metric
Given an item for recommendation, we follow the evaluation strategy of [He et al., 2017] by mixing the ground-truth items with 100 randomly sampled items those have no observed relationship with it, then we rank the ground-truth rec-items along with the 100 items and measure Hit Ratio(HR) and Normalized Discounted Cumulative Gain(NDCG) for the top \(K\). As the results for different \(K\) are consistent, we only report the result of \(K = 10\) due to space limitations.

We follow [Wang and Blei, 2011] and measure both in-matrix and out-of-matrix prediction performance. The in-matrix prediction refers to make recommendations for items those have been observed at least one time in training data, while the out-of-matrix prediction makes recommendations for cold-start items those have never been observed in training data. As the description above, CF-based methods cannot make recommendations for those out-of-matrix items because there is no behavior information.

To sum up, four metrics are reported in our experiments: In-matrix HR@10, In-matrix NDCG@10, Out-of-matrix HR@10 and Out-of-matrix NDCG@10.

5.3 Baselines
We compare our proposed SPE model with the following representatives of three types of recommendation algorithms.

- **Content-KNN.** Content \(K\)-Nearest Neighbor is a classical content-based approach. For a pair of items, their similarity is usually only measured by a distance metric between their context side information. Cosine similarity is selected in our experiments.

- **NeuMF.** Neural Matrix Factorization [He et al., 2017] is a popular collaborative filtering based methods. NeuMF combines generalized matrix factorization model and deep learning to explore behavior information.

- **CDL.** Collaborative Deep Learning [Wang et al., 2015] is one of state-of-the-art hybrid models. It jointly learns deep feature representations for items and models interactions. We do some minor modifications and also apply the stacked denoising autoencoder (sDAE) for the user side in I2I recommendation. Compared to the unmodified algorithms, the modified version achieves bet-

---

\(^1\)https://www.yelp.com/dataset
Detailed experimental results are shown in Table 1. For each dataset, the best result for each evaluation metric on original and corrupt version (\(\alpha = 0.3\)) are marked with boldface.

From Table 1, in detail, CF-based NeuMF cannot do out-of-matrix recommendation and hybrid models, CDL and SPE, obviously outperform the CF-based model (NeuMF) and content-based model (Content-KNN) for both the in-matrix and out-of-matrix metrics on original and corrupted version dataset. In particular, better performance than Content-KNN for the four out-of-matrix prediction task, especially on the corrupted version dataset, means that the tightly coupled hybrid model can indeed learn a better representation with the guide of behavior supervision information.

Moreover, our proposed SPE model significantly outperforms CDL on two of the biggest datasets and CDL achieves slightly better performance on out-of-matrix metric than SPE with Amazon dataset. It’s probably because simple multi-layer perceptron used in SPE can’t well generalize the high

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Method</th>
<th>In-HR@10</th>
<th>In-NDCG@10</th>
<th>Out-HR@10</th>
<th>Out-NDCG@10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon</td>
<td>NeuMF</td>
<td>0.5004</td>
<td>0.3109</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Content-KNN</td>
<td>0.3285</td>
<td>0.2110</td>
<td>0.1606</td>
<td>0.3396</td>
</tr>
<tr>
<td></td>
<td>CDL</td>
<td>0.5958</td>
<td>0.3475</td>
<td>0.3374</td>
<td>0.4512</td>
</tr>
<tr>
<td></td>
<td>SPE</td>
<td><strong>0.5687</strong></td>
<td><strong>0.3487</strong></td>
<td><strong>0.5160</strong></td>
<td><strong>0.5062</strong></td>
</tr>
<tr>
<td></td>
<td>original</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>corrupt</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| Yelp      | NeuMF      | 0.5127   | 0.2263     | -         | -           |
|           | Content-KNN| 0.5798   | 0.3374     | 0.2512    | 0.5780      |
|           | CDL        | 0.8068   | 0.5033     | 0.4878    | 0.7663      |
|           | SPE        | **0.8390**| **0.5386** | **0.8342**| **0.7735**  |
|           | original   |          |            |           |             |
|           | corrupt    |          |            |           |             |

| Alibaba   | NeuMF      | 0.5632   | 0.3447     | -         | -           |
|           | Content-KNN| 0.6856   | 0.5287     | 0.4153    | 0.6632      |
|           | CDL        | 0.8534   | 0.4922     | 0.4691    | 0.5780      |
|           | SPE        | **0.8890**| **0.6130** | **0.5688**| **0.8503**  |
|           | original   |          |            |           |             |
|           | corrupt    |          |            |           |             |

Table 1: Performance compared with the baselines on real-world datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#features</th>
<th>#items</th>
<th>#interactions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon</td>
<td>6310</td>
<td>1824</td>
<td>214705</td>
</tr>
<tr>
<td>Yelp</td>
<td>39117</td>
<td>116490</td>
<td>4719789</td>
</tr>
<tr>
<td>Alibaba</td>
<td>214705</td>
<td>3412637</td>
<td>7874</td>
</tr>
</tbody>
</table>

Table 2: Datasets statistics

5.4 Performance Comparison with Baselines

Better performance in almost all experimental settings. We only report results of the modified version for simplicity.

All compared implementations are with the form as their respective literature reported if not mentioned. When comparing the performance of different models, common hyper-parameters like the embedding dimension \(k\) and the negative sampling ratio \(r\) are set to the same value for fairness (if not specified, \(k\) is set to 16, and the positive/negative sampling ratio is set to 1 : 3 by default). We use grid search to find the best value of the trade-off hyperparameters for NeuMF, CDL and our proposed models. Particularly, we use the same network structure as in the original work [He et al., 2017] for NeuMF and the suggested other hyperparameters of their original work [Wang et al., 2015] are adopted for CDL. For hybrid based methods, CDL and SPE, the out-of-matrix item’s latent factor representation degenerates as \(v = e\).
dimension context information space with limited supervision behavior instances while stacked denoising autoencoder adopted in CDL can learn a robust representation and alleviate this problem with less dependency on supervision behavior information.

In order to verify our main idea that how semi-parametric model adaptively adjust confidence between parametric and non-parametric components for both cold-start and behavior-rich items, we conduct an investigation on varying of confidence weight $\delta$ along with statistics vectors input $b$. Figure 2 shows $\delta$ changing along with the number of item has ever been rated on Amazon product dataset. We can observe from Figure 2 that SPE model is paying more attention to behavior representation $z$ with the accumulation of action behaviors on the item. And also, for corrupted version dataset. SPE achieve higher confidence on non-parametric embedding $z$ than the original input version due to the uncertainty brought by noise on the parametric embedding $e$.

5.5 SPE-sDAE

As we know, because of privacy concerns, it is time-consuming and cost-expensive to collect user profiles and item information in recommender systems. The lacking entities in context feature vector and thus decrease the performance of content-based and collaborative recommender systems. On the other hand, as introduced in subsection 3.3, stacked Denoising Autoencoder(sDAE) is widely applied to do feature extraction and denoise due to the robustness on learning representation. In this subsection, we introduce sDAE into SPE to learn a robust parametric embedding $e$ for alleviating the problem when item’s side information is missing or even corrupted.

We bring reconstruction error of Eq. 3 into SPE’s objective function and update Eq. 9 as

$$
\arg \min \sum_{(i,j) \in \{P,N\}} \mathcal{L}(r_{ij}, s(v_i, v_j)) + \lambda_2 \|z\|^2 + \lambda_g \|g(\cdot)\|^2 \quad (11)
$$

where $g(\cdot)$ denotes decoding component of sDAE and $\lambda_g$ is the trade-off weight.

We further compare the experimental performance between SPE and its extended version SPE-sDAE. As stated in Section 5.1, we generate a series of corrupted versions of Amazon dataset by randomly setting entities of context information input to zero with different probabilities of corruption $\alpha$. In our experiments, 10%, 30%, 50%, 70% are chosen for parameter $\alpha$.

Comparing results with four metrics on corrupted version datasets between SPE and SPE-sDAE are plotted in Figure 3, which clearly shows, SPE-sDAE is superior to SPE on each version corrupted dataset with all the four evaluation metric. When we vary the corrupt-rate of context information input, the performance begins to decrease but SPE-sDAE is more robust than SPE, especially on the out-of-matrix items. SPE-sDAE’s performance decreases rapidly, which means the extended version SPE can actually learn a powerful representation and alleviate the cold-start problem on even noisy input datasets.

6 Conclusion

In this paper, we propose a semi-parametric embedding framework (named as SPE) for hybrid item-item recommendation. In particular, SPE explicitly represents item as combination of context-based vector and behavior-based vector. It is a tightly coupled hybrid framework which provides two-way interaction between behavior and context information. The parametric embedding can deal with cold-start items while SPE maintains performance for behavior-rich items with the help of non-parametric part. Our experimental results on real-world datasets show the effectiveness and robustness of SPE, especially on out-of-matrix items. We introduce sDAE to build a robust version of SPE and demonstrate its superiority. It is not difficult to find that, although the SPE framework is designed for the hybrid I2I problem in this paper, it is not difficult to extend it to the user-item recommendation problem, which can be interesting future work.

References


