Pattern Recognition Letters

Authorship Confirmation

Please save a copy of this file, complete and upload as the "Confirmation of Authorship" file.

As corresponding author I, Yonghong Yu, Ningning Zhou, hereby confirm on behalf of all authors that:

1. This manuscript, or a large part of it, has not been published, was not, and is not being submitted to any other journal.
2. If presented at or submitted to or published at a conference(s), the conference(s) is (are) identified and substantial justification for re-publication is presented below. A copy of conference paper(s) is(are) uploaded with the manuscript.
3. If the manuscript appears as a preprint anywhere on the web, e.g. arXiv, etc., it is identified below. The preprint should include a statement that the paper is under consideration at Pattern Recognition Letters.
4. All text and graphics, except for those marked with sources, are original works of the authors, and all necessary permissions for publication were secured prior to submission of the manuscript.
5. All authors each made a significant contribution to the research reported and have read and approved the submitted manuscript.

Signature Yonghong Yu, Ningning Zhou Date 2020-04-16

List any pre-prints:

Relevant Conference publication(s) (submitted, accepted, or published):
Lihong Jiao, Yonghong Yu, Ningning Zhou, Li Zhang, Hongzhi Yin, 2020. Neural pairwise ranking factorization machine for item recommendation, in the 25th International Conference on Database Systems for Advanced Applications. (Accepted)

Justification for re-publication:
Compared with the conference version, this journal version makes the following new contributions:

- Our proposed neural pairwise ranking factorization machine model that is accepted by DASFAA2020 considers each feature interaction equally and ignores that different feature interaction has different weight on the final predicted scores. Hence, in this extension, we also propose an attention boosted neural pairwise ranking factorization machine model, which integrates the attention mechanism into the neural pairwise ranking factorization machine model to further improve the recommendation performance of our proposed neural pairwise ranking factorization machine model. Specifically, we employ a neural attention network on the Bi-Interaction layers to learn the weight of each feature interaction.
- We conduct comprehensive experiments to evaluate the performance of the attention enhanced neural pairwise ranking factorization machine model as well as perform sensitive analysis for important model parameters.
- We have added the Related Work Section to make our paper more complete.
- To help readers to understand our work, we enrich the technical details of factorization machine model in Section Preliminaries as well as our proposed neural pairwise ranking factorization model in Section Neural pairwise ranking factorization machine, respectively.

Pattern Recognition Letters
journal homepage: www.elsevier.com

Enhanced Factorization Machine via Neural Pairwise Ranking and Attention Networks

Yonghong Yu[a,**], Lihong Jiao[b], Ningning Zhou[b,**], Li Zhang[c], Hongzhi Yin[d]

[a]*Tongda College, Nanjing University of Posts and Telecommunications, Nanjing, China.*
[b]*School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing, China.*
[c]*Department of Computer and Information Sciences, Northumbria University, Newcastle, UK.*
[d]*School of Information Technology and Electrical Engineering, The University of Queensland, Australia.*

ABSTRACT

The factorization machine models attract significant attention nowadays since they improve recommendation performance by incorporating context information into recommendation modeling. However, traditional factorization machine models often adopt the point-wise learning method for model parameter learning, as well as only model the linear interactions between features. They substantially fail to capture the complex interactions among features, which degrades the performance of factorization machine models. In this research, we propose a neural pairwise ranking factorization machine for item recommendation, namely NPRFM, which integrates the multi-layer perceptual neural networks into the pairwise ranking factorization machine model. Specifically, to capture the high-order and nonlinear interactions among features, we stack a multi-layer perceptual neural network over the bi-interaction layer, which encodes the second-order interactions between features. Moreover, instead of the prediction of the absolute scores, the pair-wise ranking model is adopted to learn the relative preferences of users. Since NPRFM does not take into account the importance of feature interactions, we propose a new variant of NPRFM, which learns the importance of feature interactions by introducing the attention mechanism. The empirical results on real-world datasets indicate that the proposed neural pairwise ranking factorization machine outperforms the traditional factorization machine models.

1. Introduction

With the development of information technology, a variety of network applications have accumulated a huge amount of data. Although the massive data provides users with rich information, it leads to the problem of "information overload". With the huge volume of data available, it is challenging for users to efficiently find the valuable information. On the other hand, for the content providers, it is vital to increase business revenue by recommending suitable products to potential users. The recommendation systems (Adomavicius and Tuzhilin, 2005) can greatly alleviate the problem of information overload. They infer users latent preferences by analyzing their past activities and provide them with personalized recommendation services.

In the field of recommendation systems, collaborative filtering (CF) (Breese et al., 1998; He et al., 2017; Linden et al., 2003) algorithms are the most popular methods, which utilize users' behavior information to make recommendations and are independent of the specific application domains. However, the traditional collaborative filtering methods ignore the contextual information related to users and items, resulting in a suboptimal recommendation performance. In reality, the contextual information (e.g. time, place and mood) greatly affects the decisions of users. For example, the user is more likely to watch different types of movies in different moods, and visit different popular spots at different cities. In order to make context-aware recommendations available to potential users, several context-based recommendation models are proposed (Adomavicius and Tuzhilin, 2011; Chen, 2005; Baltrunas et al., 2011; Zheng et al., 2015; Rendle, 2010, 2012). Adomavicius et al. (Adomavicius and Tuzhilin, 2011) provided an overview of the multifaceted notion of context, and discussed several approaches for incor-

**Corresponding author.
e-mail: yuyh@njupt.edu.cn (Yonghong Yu), zhounn@njupt.edu.cn (Ningning Zhou)

porating contextual information into the recommendation process. In addition, they illustrated the usage of context-aware recommendation methods in several application areas where different types of contexts were exploited. In particular, Rendle et al. (Rendle, 2010, 2012) proposed the popular factorization machine (FM) model. As a general predictor, the factorization machine takes the interactions between different context features into account for model building. In fact, FM has been the defacto standard for context-aware recommendation models, and various of extensions of FM have been proposed (He and Chua, 2017; Xiao et al., 2017; Xin et al., 2019; Hong et al., 2019; Yuan et al., 2016; Guo et al., 2016; Juan et al., 2016).

Recently, deep learning techniques have shown great potential in many fields, such as natural language processing, speech recognition and computer vision. In the field of context-aware recommendation systems, some researchers also have utilized deep learning techniques to improve the classic factorization machine models. Typical deep learning based factorization machine models include NFM (He and Chua, 2017), AFM (Xiao et al., 2017), CFM (Xin et al., 2019), and IFM (Hong et al., 2019). The Neural Factorization Machine (NFM) (He and Chua, 2017) seamlessly unifies the advantages of neural networks and the factorization machine. It not only captures the linear interactions between feature representations of variables, but also models nonlinear high-order interactions. However, both FM and NFM adopt a point-wise method to learn their model parameters. They fit the user's scores rather than learn the user's relative preferences for item pairs. In fact, common users usually care about the ranking of item pairs rather than the absolute rating on each item. The pairwise ranking factorization machine (PRFM) (Yuan et al., 2016; Guo et al., 2016) makes use of the Bayesian personalized ranking (BPR) (Rendle et al., 2009) and FM to learn the relative preferences of users over item pairs. Similar to FM, PRFM can only model the linear interactions among features. As a result, the above studies reveal that both the neural networks and the pair-wise learning method are beneficial for the factorization machine, which endow the factorization machine with capacities of modeling non-linear interactions and learning the ranking of item pairs, respectively. However, there are no effective schemes that unify the FM model, neural networks and the BPR criterion into an integrated framework, which is capable of tackling the intrinsic weaknesses of each independent model.

In this research, we propose the Neural Pairwise Ranking Factorization Machine (NPRFM) model, which integrates the multi-layer perceptual neural networks into the PRFM model to boost the recommendation performance. There are three fundamental components, i.e., multi-layer perceptual neural networks, factorization machine model and the BPR criterion. Specifically, to capture the high-order and nonlinear interactions among features, we stack a multi-layer perceptual neural network over the bi-interaction layer, which is a pooling layer that encodes the seconde-order interactions between features. Moreover, the BPR criterion is adopted to learn the relative preferences of users, which makes non-observed feedback contribute to the inference of model parameters. Hence, the proposed neural pairwise ranking factorization machine model unifies the strength of three fundamental components and effectively deals with their respective drawbacks. Owing to the fact that NPRFM does not consider the importance of feature interactions, we propose an attention boosted NPRFM to further improve the recommendation performance. Concretely, to learn the importance of feature interactions, we employ a neural attention network on the pooling operation in the Bi-Interaction layer. The empirical results on real world datasets indicate that our proposed neural pairwise ranking factorization machine model outperforms the traditional recommendation algorithms.

2. Related work

In this section, we review the key related studies, including traditional collaborative filtering methods, context-aware recommendation models and the attention mechanism, especially the factorization machine model and its extensions.

2.1. Traditional collaborative filtering

The traditional recommendation algorithms can be roughly divided into three categories: content-based, collaborative filtering and hybrid recommendation algorithms (Adomavicius and Tuzhilin, 2005). Collaborative filtering is one of the most popular recommendation techniques in the research of recommender systems. It mainly includes memory-based and model-based methods. Typical memory-based approaches include user-based CF (Breese et al., 1998) and item-based CF (Sarwar et al., 2001; Linden et al., 2003), while model-based filtering approaches include Bayesian networks (Breese et al., 1998), clustering model (Xue et al., 2005; Yu et al., 2013), latent semantic analysis (Hofmann, 2004, 2003), restricted Boltzmann machines (Salakhutdinov et al., 2007), and matrix factorization (Koren et al., 2009), etc. In fact, matrix factorization (Koren et al., 2009) has become the defacto standard in the research of recommendation systems, and various extensions of MF have been proposed (Mnih and Salakhutdinov, 2008; Koren, 2008; Lee and Seung, 1999; Yu et al., 2009). Matrix factorization maps both users and items into a low-dimensional latent factor space, using the inner product of the user's and item's low-dimensional feature vectors to predict the user's score on the item. Typical matrix factorization models include PMF (Mnih and Salakhutdinov, 2008), SVD++ (Koren, 2008), NMF (Lee and Seung, 1999), NPCA (Yu et al., 2009), etc.

2.2. Context-boosted collaborative filtering

The traditional recommendation methods overlook the contextual information involved in the recommendation systems, which greatly affects the decision-making of users. Adomavicius et al. (Adomavicius and Tuzhilin, 2011) provided an overview of the multifaceted notion of context, and discussed several approaches for incorporating contextual information into the recommendation process. In order to tackle the problem of context-aware recommendation, several context-boosted recommendation methods have been proposed. In (Chen, 2005),

Chen et al. proposed a context-aware collaborative filtering method that predicts a user's preferences in different context situations based on past experiences. In (Baltrunas et al., 2011), Baltrunas et al. presented a context-aware matrix factorization method, which models the interactions between the contextual factors and item ratings. Based on the assumption that recommendation lists should be similar if their contextual situations are similar, Zheng et al. (Zheng et al., 2015) proposed a similarity-learning model. Their proposed model integrates context similarity with the sparse linear recommendation model. In (Rendle, 2010, 2012), Rendel et al. proposed the factorization machine (FM), which is a general predictor that can be adopted for the prediction tasks working with any real valued feature vector. The FM method is able to model the interactions among different features. Especially, it is able to break the independence between interaction features by decomposing them, which means that the information related to one interaction is beneficial for learning the parameters of related interactions. Owing to its effectiveness and flexibility, various extensions of FM have been proposed. As an example, Qiang et al. (Qiang et al., 2013) proposed a ranking factorization machine (RankingFM) model, which applies FM model to microblog ranking on the basis of pairwise classification. The RankingFM model unifies the generality of learning to rank framework and the advantages of factorization model in estimating interaction parameters between features, leading to better retrieval performance. In (Guo et al., 2016), Guo et al. proposed the pairwise ranking factorization machine (PRFM), which alleviates the cold start problem and enhances the performance of personalized ranking by incorporating BPR (Rendle et al., 2009) with factorization machine. Juan et al. (Juan et al., 2016) presented the field-aware factorization machine (FFM), which is used to classify large sparse data. Inspired by LambdaRank (Yan et al., 2010), Yuan et al. (Yuan et al., 2016) proposed the Lambda factorization machine (LambdaFM), which is particularly intended for optimizing ranking performance for the problem of implicit feedback based context-aware recommendation. Recently, In (Chen et al., 2020), Chen et al. proposed an efficient non-sampling factorization machine framework, namely ENSFM, for context-aware top-k recommendation. ENSFM not only seamlessly connects the relationship between factorization machines and matrix factorization, but also resolves the challenging efficiency issue of non-sampling learning. Xu et al. (Xu and Wu, 2020) proposed a lightweight model named LorentzFM for recommendation and click through rate prediction tasks. Moreover, they proposed a new score function by characterizing if the triangle inequality for Lorentz distance is violated or not in the hyperboloid model.

With the development of deep learning techniques, some researchers have adopted deep learning algorithms to improve the performance of FM. Since FM can only model the linear interactions between feature representations of variables, He et al. (He and Chua, 2017) proposed a novel model for sparse data prediction, named Neural Factorization Machine (NFM), which seamlessly integrates neural networks into factorization machine mode. It not only captures the linear interactions among representations of features, but also models nonlinear high-order interactions among them. In (Xiao et al., 2017), Xiao et al. proposed the attentional factorization machine (AFM) model, which learns the importance of each feature interaction from data via a neural attention network. The AFM model is able to enhance the expressiveness as well as boost the interpretability of FM model. In (Guo et al., 2017), Guo et al. proposed a new neural network model DeepFM that integrates the architectures of FM and deep neural network (DNN). Specifically, the DeepFM models low-order feature interactions like FM and models high-order feature interactions like DNN. Moreover, based on the DeepFM model, Zhang et al. (Zhang et al., 2019) proposed a novel neural click through rate model named FAT-DeepFM that enhances the DeepFM model by introducing the compose-excitation network field attention to dynamically capture each feature's importance before explicit feature interaction procedure. In addition, Xin et al. (Xin et al., 2019) proposed a novel context-aware recommendation algorithm, called Convolutional Factorization Machine (CFM). CFM firstly models the second-order interactions with outer product, resulting in "images" which capture correlations between embedding dimensions. Then, all the generated "images" are stacked, and form an interaction cube. Finally, a 3D Convolutional Neural Networks (CNN) is subsequently applied to learn high-order interaction signals in an explicit manner.

2.3. Attention mechanism

Recently, attention mechanism has been adopted in many fields owing to its efficiency and robustness, such as natural language processing, speech recognition and computer vision. Some recent studies have also utilized the attention mechanism to improve the recommendation performance. As an example, Chen et al. (Chen et al., 2017b) proposed a novel convolutional neural network, called SCA-CNN, that incorporates spatial and channel-wise attentions into a CNN model. To effectively select "good" interactive features in context-aware recommendations, Cheng et al. (Cheng et al., 2014) proposed a novel gradient boosting factorization machine (GBFM) model, which incorporates feature selection algorithm with FM into a unified framework. In (Chen et al., 2017a), Chen et al. proposed an attention collaborative filtering (ACF) model to address the challenging item- and component-level implicit feedback in multimedia recommendation. ACF model consists of two attention modules: the component-level attention module, starting from any content feature extraction network, which learns to select informative components of multimedia items, and the item-level attention module, which learns to score item preferences.

Compared with the above methods, the main differences between our proposed methods and existing studies include the following aspects: (1) Unlike PRFM that only models the linear interactions among features, we stack a multi-layer perceptual neural network over the bi-interaction layer to capture the high-order and nonlinear interactions between features. (2) Differ from FM and NFM that adopt the point-wise method to learn their model parameters, our proposed models utilize the BPR, i.e. a pair-wise learning method, to learn model parameters. (3) Furthermore, we integrate the attention mechanism into the neural pairwise ranking factorization machine to learn

the weight of each feature interaction. In general, our proposed models unify the strength of multi-layer perceptual neural networks, factorization machine, BPR criterion and the attention mechanism.

3. Preliminaries

3.1. Factorization machine

Factorization Machine is able to model the interactions among different features by using a factorization model. Especially, FM model breaks the independence between interaction features by decomposing them to estimate interactions. In other words, the information related to one interaction is beneficial for learning the parameters of related interactions. Moreover, the FM model is endowed with strong expressiveness ability. For example, matrix factorization, Support Vector Machine (SVM) (Suykens and Vandewalle, 1999) and factorized personalized markov chains (FPMC) (Rendle et al., 2010) can be induced from FM by constructing an appropriate input data format.

Usually, the model equation of FM is defined as follows:

$$\hat{y}(\mathbf{x}) = w_0 + \sum_{i=1}^{n} w_i x_i + \sum_{i=1}^{n} \sum_{j=i+1}^{n} \langle \mathbf{v}_i, \mathbf{v}_j \rangle x_i x_j \tag{1}$$

where $\hat{y}(\mathbf{x})$ is the predicted value, and $\mathbf{x} \in R^n$ denotes the input vector of the model equation. x_i represents the i-th element of $\mathbf{x}$. $w_0 \in R$ is the global bias, $\mathbf{w} \in R^n$ indicates the weight vector of the input vector $\mathbf{x}$. $\mathbf{V} \in R^{n \times k}$ is the latent feature matrix, whose $\mathbf{v}_i$ represents the feature vector of x_i. $\langle \mathbf{v}_i, \mathbf{v}_j \rangle$ is the dot product of two feature vectors, which is used to model the interaction between x_i and x_j.

By mathematical derivation, $\hat{y}(\mathbf{x})$ can be further rewritten as:

$$\hat{y}(\mathbf{x}) = w_0 + \sum_{i=1}^{n} w_i x_i + \frac{1}{2} \sum_{f=1}^{k} ((\sum_{i=1}^{n} v_{i,f} x_i)^2 - \sum_{i=1}^{n} v_{i,f}^2 x_i^2) \tag{2}$$

According to Eq. (2), the time complexity of the model equation of FM is $O(k.n)$, which indicates that the computation cost is linear with respect to the dimension of latent feature and the number of features.

4. Neural pairwise ranking factorization machine

The factorization machine model is a strong competitor in the area of context-aware recommendation and has shown promising results. In fact, the factorization machine has been the defacto standard for context-aware recommendation task, and several variants of FM have been proposed, for instance, NFM (He and Chua, 2017), AFM (Xiao et al., 2017), PRFM (Guo et al., 2016; Qiang et al., 2013), CFM (Xin et al., 2019) and so on. However, FM only captures the second-order interactions among features, which is insufficient to model the complex interaction patterns between features. In order to tackle this issue, NFM integrates multi-layer perceptual neural networks into FM to learn the nonlinear high-order interactions. However, both FM and NFM focus on predicting the absolute ratings

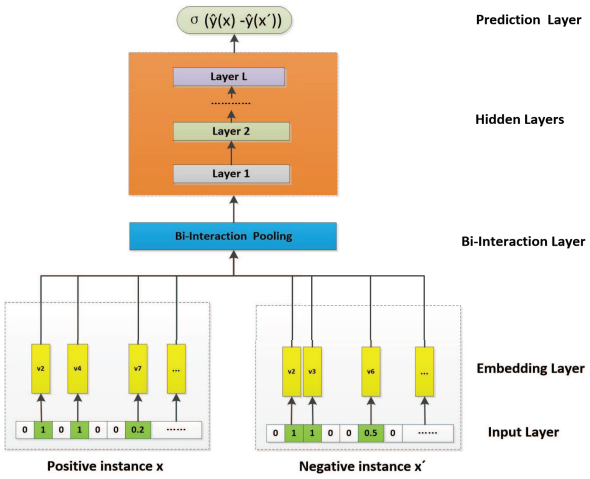


Fig. 1. The framework of the neural pairwise ranking factorization machine

for target items, which is different from the concern of common users that is to learn the relative ranking between item pairs. In addition, PRFM is designed for the ranking task, and learns the relative preferences of users for item pairs. Specifically, instead of using the point-wise learning method, the PRFM mechanism adopts the pair-wise learning method to learn the model parameters. To some extent, the scheme of pair-wise learning method is able to alleviate the issue of data sparsity because both observed and unobserved feedback contributes to the learning of model parameters. But, similar to FM, the PRFM can not model the complex interaction patterns between different features. To model the high-order interaction behaviors among features as well as learn the relatively preferences of user over item pairs, we propose the neural pairwise ranking factorization machine (NPRFM) model, whose underlying components are NFM and PRFM. In NPRFM, we stack a multi-layer perceptual neural network (MLP) over the bi-interaction layer to capture the high-order and nonlinear interactions among features. Fig. 1 presents the framework of the proposed neural pairwise ranking factorization machine, which consists of four layers, i.e. embedding layer, Bi-interaction layer, hidden layer and prediction layer. The input of NPRFM includes positive and negative instances. Both positive and negative instances contain user, item and context information. By using one-hot encoding, the positive and negative instances are converted into sparse feature vectors $\mathbf{x} \in R^n$ or $\mathbf{x}' \in R^n$, respectively. A toy example of one-hot encoded positive or negative instance is illustrated as follows,

$$\underbrace{\underbrace{[0,0,0,1...,0]}_{userID=3}\ \underbrace{[0,0,1,0...,0]}_{itemID=2}\ \underbrace{[0,1,1,0,0,1,0,0]}_{city=china,mood=happy,weather=sunny}}_{A\ one-hot\ encoded\ instance} \tag{3}$$

where the first component represents the user information and the second element indicates the item information. And context information, such as country, mood, weather etc., are located in the third factor. In the one-hot encoded sparse feature vector, the feature value $x_i = 0$ means the i-th feature does not exist in the instance.

4.1. Embedding layer

The goal of embedding layer is to map each feature into a low-dimensional space, where each feature is represented as a compact and dense real-value vector, instead of a sparse and high-dimensional vector. After one-hot encoding, we use the embedding table lookup operation to obtain the embedded representations of features included in the input instance. Formally, the embedded representation of $\mathbf{x}$ is,

$$\mathbf{V}_x = \mathbf{V}.onehot(\mathbf{x}) \tag{4}$$

where $\mathbf{V}_x$ is a set of embedding vectors, i.e., $\mathbf{V}_x = \{x_1\mathbf{v}_1, ..., x_n\mathbf{v}_n\}$, and $\mathbf{v}_i \in R^k$ is the embedded representation of the i-th feature. Owing to the sparsity of $\mathbf{x}$, only the embedded representations of non-zero features (i.e.,$x_i \neq 0$) are included in $\mathbf{V}_x$.

4.2. Bi-Interaction layer

The Bi-Interaction layer is a pooling operation, which converts the set of embedding vectors $\mathbf{V}_x$ into one vector $f_{BI}(\mathbf{V}_x)$:

$$f_{BI}(\mathbf{V}_x) = \sum_{i=1}^{n}\sum_{j=i+1}^{n} x_i\mathbf{v}_i \odot x_j\mathbf{v}_j \tag{5}$$

where $\odot$ represents the element-wise product of two vectors. As shown in Eq.(5), the Bi-Interaction layer captures the pair-wise interactions among the low dimensional representations of features. In other words, the Bi-Interaction pooling only encodes the second-order interactions among features.

4.3. Hidden layers and prediction layer

Since the Bi-interaction layer only captures the second-order interactions among features, and can not model the complexity interactive patterns among features, we utilize the multi-layer perceptron (MLP) to learn the interaction relationships among features, which endows the proposed model with the ability of capturing the high-order interactions. In fact, as reported in (Hornik et al., 1989), the multi-layer perceptron is able to approximate any measurable function. Moreover, some researchers also utilized the MLP to improve the performance of recommendation models (He et al., 2017; Yu et al., 2019). Specifically, in the hidden layers, we stack multiple fully connected hidden layers over the Bi-Interaction layer, where the output of a hidden layer is used as the input of the subsequent hidden layer that makes use of the weighted matrix and non-linear activation function, such as sigmoid, tanh and ReLU, to nonlinearly transform this output. Formally, the MLP model is defined as,

$$\begin{aligned}
&\mathbf{z}_1 = \sigma_1\left(\mathbf{W}_1 f_{BI}\left(\mathbf{V}_x\right) + \mathbf{b}_1\right),\\
&\mathbf{z}_2 = \sigma_2\left(\mathbf{W}_2\mathbf{z}_1 + \mathbf{b}_2\right),\\
&...\\
&\mathbf{z}_L = \sigma_L\left(\mathbf{W}_L\mathbf{z}_{L-1} + \mathbf{b}_L\right)
\end{aligned} \tag{6}$$

where L denotes the number of hidden layers. $\mathbf{W}_l \in R^{k_l \times k_{l-1}}$ and $\mathbf{b}_l \in R^{k_l}$ represent the weight matrix and bias vector for the l-th layer, respectively. And k_l denotes the transform size of the l-th hidden layer.

The prediction layer is connected to the last hidden layer, and is used to predict the score $\hat{y}(\mathbf{x})$ for the instance $\mathbf{x}$, where $\mathbf{x}$ can be positive or negative instances. Formally,

$$\hat{y}(\mathbf{x}) = \mathbf{h}^T\mathbf{z}_L \tag{7}$$

where $\mathbf{h}$ is the weight vector of the prediction layer.

Combining the Eq.(6) and (7), the model equation of NPFFM is reformulated as:

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{n} w_i x_i + \mathbf{h}^T\sigma_L(\mathbf{W}_L(...\sigma_1(\mathbf{W}_1 f_{BI}(\mathbf{V}_x) + \mathbf{b}_1)...) + \mathbf{b}_L) \tag{8}$$

4.4. Model learning

Our proposed NPRFM approach focuses on collaborative filtering with implicit feedback, which learns the relative preferences for item pairs rather than predicts the absolute ratings. Hence, we adopt a ranking criterion, i.e, the BPR criterion, to optimize the model parameters. Formally, the objective function of NPRFM is defined as:

$$\mathcal{L}^{NPRFM} = \sum_{(\mathbf{x},\mathbf{x}')\in\chi} -ln\sigma(\hat{y}(\mathbf{x}) - \hat{y}(\mathbf{x}')) + \frac{\lambda}{2}(\|\Theta\|_F^2) \tag{9}$$

where $\sigma(.)$ is the logistic sigmoid function. And $\Theta = \{w_i, \mathbf{W}_l, \mathbf{b}_l, \mathbf{v}_i, \mathbf{h}\}$, $i \in (1...n)$,$l \in (1...L)$ denotes the model parameters. χ is the set of positive and negative instances.

In the process of model training, we adopt the uniform sampling scheme to sample one negative instance for each positive instance. After shuffling the sampled instances, we feed a batch of instances into our proposed neural personalized ranking factorization machine model. In addition, we adopt the Adagrad (Lu et al., 2017) optimizer to update model parameters since the Adagrad optimizer utilizes the information of the sparse gradient and gains an adaptive learning rate, which is suitable for the scenarios of data sparsity.

4.5. Attention Boosted NPRFM

FM enhances linear regression models by combining second-order feature interactions. Despite effectiveness, FM can be disturbed by modeling all feature interactions with the same weights, since not all feature interactions are equally useful and predictive. For example, the interactions with useless features may even generate noise and greatly degrade the performance. In this section, we introduce the attention mechanism into NPRFM to further improve the recommended performance. Fig. 2 presents the framework of the proposed attention boosted NPRFM. The experimental results are shown in Section 5.3. The attention mechanism is employed to the pooling operation in the Bi-Interaction layer:

$$f_{ABI}(\mathbf{V}_x) = a_f \odot f_{BI}(\mathbf{V}_x) \quad (f = 1, \cdots, k) \tag{10}$$

where a_f is the attention score for $f_{BI}(\mathbf{V}_x) \in R^k$, which can be interpreted as the importance of $f_{BI}(\mathbf{V}_x)$ in predicting the target. We parameterize the attention score with a multi-layer perceptron (MLP), which is called the attention network. The input

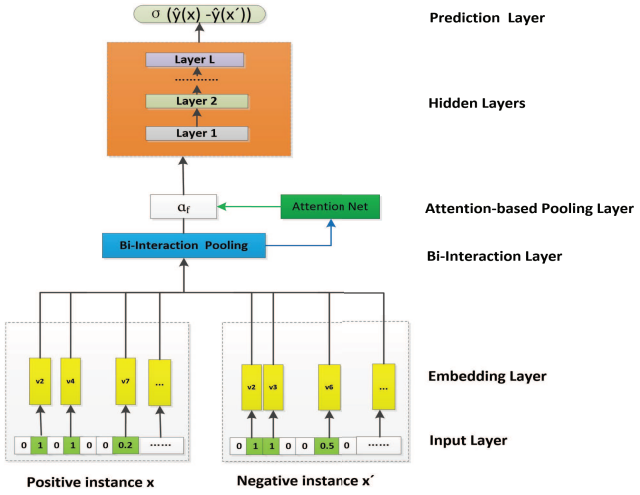


Fig. 2. The framework of attention boosted NPRFM

of the attention network is a k-dimensional vector obtained after the pooling operation. Formally, the attention network is defined as:

$$a'_f = \mathbf{P}^T ReLU\left(\mathbf{W}' f_{BI}(\mathbf{V}_x) + \mathbf{b}'\right) \tag{11}$$

$$a_f = \frac{exp(a'_f)}{\sum_{f=1}^{k} exp(a'_f)} \tag{12}$$

where $\mathbf{W}' \in R^{t \times k}$, $\mathbf{b}' \in R^t$ and $\mathbf{P} \in R^t$ represent the weight matrix, bias vector and prediction weight for the attention network, respectively. And t denotes the hidden layer size of the attention network, which we call the attention factor. The attention scores are normalized by the softmax function, while the rectifier(ReLU) is used as the activation function.

Formally, the model equation of NPRFM with the attention mechanism is reformulated as:

$$\hat{y}_{Att}(\mathbf{x}) = \sum_{i=1}^{n} w_i x_i + \mathbf{h}^T \sigma_L(\mathbf{W}_L(...\sigma_1(\mathbf{W}_1(a_f \odot f_{BI}(\mathbf{V}_x)) + \mathbf{b}_1)...) + \mathbf{b}_L) \tag{13}$$

5. Experiments

In order to evaluate the performance of the proposed neural pairwise ranking factorization machine, we compare our proposed models with other baselines on real-world datasets.

5.1. DataSets and evaluation metrics

In our experiments, we choose two real-world implicit feedback datasets: Frappe[1] and Last.fm[2], to evaluate the effectiveness of the proposed model.

Frappe: Frappe is a context-aware application discovery tool. This dataset was collected by Baltrunas et al. (Baltrunas et al., 2015). It contains 96,203 application usage logs with different contexts. Besides the user ID and application ID, each log contains eight contexts, such as weather, city and country and so on. We use one-hot encoding to convert each log into one feature vector, resulting in 5382 features.

Last.fm: The last.fm dataset is used for music recommendation. This dataset was collected by Xin et al. (Xin et al., 2019). The contexts of user consist of the user ID and the last music ID listened by the specific user within 90 minutes. The contexts of item include the music and artist IDs. This dataset contains 214,574 music listening logs. After transforming each log by using one-hot encoding, we retrieve 37,358 features.

We adopt the leave-one-out validation to evaluate the performance of all compared methods, which has been widely used in the literature (He et al., 2017; Xiao et al., 2017; Hong et al., 2019). For each user, we take his/her latest interaction log as the test set and the remaining interactions as the training set. Since both original datasets contain only positive instances, we extract two negative instances pertaining to each positive instance. For example, for each log of Frappe, we randomly extract two applications that the user did not adopt in the context, which is given in this log.

Table 1 summarizes the statistics of the datasets.

Table 1. Dataset statistics

Dataset	# User	# Item	# Transactions	# Features	# Context
Frappe	957	4082	96203	5382	10
Lastfm	1000	20301	214574	37358	4

We utilize two widely used ranking based metrics, i.e., the Hit Ratio (*HR*) and Normalized Discounted Cumulative Gain (*NDCG*), to evaluate the performance of all comparisons. *HR@n* measures whether the generated recommendation list contains the test item. *NDCG@n* is the normalization of *DCG* (Discounted Cumulative Gain), which assigns higher scores to the test item with top ranks.

5.2. Experimental settings

In order to evaluate the effectiveness of the proposed algorithms, we employ FM, NFM, PRFM as the baseline methods.

- **FM** : FM (Rendle, 2010, 2012) is a strong competitor in the field of context-aware recommendation, and captures the interactions between different features by using a factorization model. In addition, FM focuses on the task of predicting the absolute ratings of items.

- **NFM** : NFM (He et al., 2017) seamlessly integrates neural networks into factorization machine model. Based on the neural networks, NFM can model nonlinear and high-order interactions between latent representations of features. Similar to FM, NFM also focuses on predicting the absolute ratings of items.

- **PRFM**: PRFM (Guo et al., 2016) applies the BPR standard to optimize its model parameters. Different from FM and NFM, PRFM focuses on the ranking task that learns the relative preferences of users for item pairs rather than predicts the absolute ratings.

[1] http://baltrunas.info/research-menu/frappe
[2] http://www.dtic.upf.edu/ocelma/MusicRecommendationDataset

In order to make a fair comparison, we set the parameters of each method according to respective references or based on our experiments. Under these parameter settings, each method achieves its best performance. For all compared methods, we set the dimension of the hidden feature vector $k = 64$. In addition, for FM, we set the regularization term $\lambda = 0.01$ and the learning rate $\eta = 0.001$. For NFM, we set the number of hidden layers 1, the regularization term $\lambda = 0.01$ and the learning rate $\eta = 0.001$. For PRFM, we set the regularization term $\lambda = 0.001$ and the learning rate $\eta = 0.1$. For both the NPRFM and the attention boosted NPRFM, we set the regularization term $\lambda = 0.001$, the learning rate $\eta = 0.1$, and the number of hidden layers $L = 1$. In addition, we initialize the latent feature matrix V of NPRFM with the embedded representations learned by PRFM.

5.3. Performance comparison

We set the length of recommendation list $n = 3, 5, 7$ to evaluate the performance of all compared methods. When the dimension of feature representation k is equal to 64, the experimental results on the two datasets are illustrated in Tables 2 and 3. In addition, the experimental results with $k = 32$ are listed in Tables 4 and 5.

Table 2. Performance comparison on the Frappe dataset (k=64)

Recommendation	n=3		n=5		n=7	
Algorithm	HR	NDCG	HR	NDCG	HR	NDCG
FM	0.2445	0.1795	0.3050	0.2107	0.3422	0.2216
NFM	0.2510	0.1797	0.3702	0.2199	0.4686	0.2504
PRFM	0.4650	0.3868	0.5654	0.4280	0.6383	0.4533
NPRFM	0.4786	0.3962	0.5751	0.4358	0.6469	0.4607
NPRFM+attention	0.4824	0.4036	0.5813	0.4442	0.6578	0.4706

Table 3. Performance comparison on the Last.fm dataset (k=64)

Recommendation	n=3		n=5		n=7	
Algorithm	HR	NDCG	HR	NDCG	HR	NDCG
FM	0.0770	0.0584	0.1064	0.0706	0.1344	0.0803
NFM	0.0972	0.0723	0.1372	0.0886	0.1702	0.1000
PRFM	0.1828	0.1374	0.2545	0.1667	0.3094	0.1857
NPRFM	0.1855	0.1402	0.2624	0.1715	0.3219	0.1921
NPRFM+attention	0.1984	0.1504	0.2761	0.1822	0.3341	0.2023

Table 4. Performance comparison on the Frappe dataset (k=32)

Recommendation	n=3		n=5		n=7	
Algorithm	HR	NDCG	HR	NDCG	HR	NDCG
FM	0.2435	0.1843	0.3017	0.2075	0.3413	0.2096
NFM	0.2478	0.1846	0.3076	0.2092	0.3540	0.2252
PRFM	0.4305	0.3521	0.5309	0.3933	0.6011	0.4177
NPRFM	0.4535	0.3697	0.5515	0.4090	0.6144	0.4310
NPRFM+attention	0.4659	0.3845	0.5583	0.4226	0.6242	0.4453

As illustrated in Tables 2-5, we have the following observations: (1) On both datasets, FM performs the worst among all the compared methods. The reason is that FM learns its model parameters by adopting a point-wise learning scheme, which usually suffers from data sparsity. (2) NFM is superior to FM with regards to all evaluation metrics. This observation demonstrates that integrating neural networks is beneficial for FM to improve its recommendation performance. One reason is that the non-linear and high-order interactions among representations of features are captured by utilizing the neural networks, resulting in the improvement of recommendation performance. (3) On both datasets, PRFM achieves better performance than those of FM and NFM. This is because PRFM learns its model parameters by applying the BPR criterion, in which the pair-wise learning method is used to infer the latent representations of users and items. To some extent, the pair-wise learning scheme is able to alleviate the problem of data sparsity by making non-observed feedback contribute to the learning of model parameters. (4) Our proposed NPRFM model consistently outperforms other compared methods, which demonstrates the effectiveness of the proposed strategies. Specifically, when $n = 3$, NPRFM improves the *HR* of PRFM by 2.9% and 1.5% on Frappe and Last.fm, respectively. In terms of *NDCG*, the improvements of NPRFM over PRFM are 2.4% and 2.0% on Frappe and Last.fm, respectively. This observation confirms our assumption that it is beneficial to unify the strengths of NFM model in capturing non-linear and high-order interaction relationships and the PRFM model in learning users preferences ranking between items. (5) On both datasets, the attention boosted NPRFM obtains better performance than NPRFM. This is because attention boosted NPRFM enhances NPRFM by learning the importance of feature interactions with an attention network, which not only improves the representation ability but also the interpretability of the NPRFM model. (6) All the compared methods with $k = 64$ are more competent than those with $k = 32$. The sensitivity analysis of the dimension of the latent representation of feature is presented in the following section.

5.4. Sensitivity analysis

5.4.1. Impact of the depth of neural networks

In the proposed model, we use the neural networks, i.e, MLP, to learn the nonlinear interactions between embedded representations of different features. The depth of neural networks is an important factor that affects the expressiveness of neural networks. In this section, we conduct a group of experiments to investigate the impact of the depth of neural networks on the recommendation quality. We set $n = 5$ and $k = 64$, and vary the depth of neural networks from 1 to 3.

In Table 6, NPRFM-i denotes the NPRFM model with i hidden layers. Particularly, NPRFM-0 is equal to PRFM. NPRFM-attention-i denotes the attention enhanced NPRFM with i hidden layers. Principally, NPRFM-attention-0 is equal to atten-

Table 5. Performance comparison on the Last.fm dataset (k=32)

Recommendation	n=3		n=5		n=7	
Algorithm	HR	NDCG	HR	NDCG	HR	NDCG
FM	0.0661	0.0501	0.0916	0.0604	0.1163	0.0690
NFM	0.0898	0.0662	0.1314	0.0835	0.1666	0.0959
PRFM	0.1345	0.1024	0.1927	0.1259	0.2369	0.1412
NPRFM	0.1453	0.1107	0.2027	0.1342	0.2498	0.1505
NPRFM+attention	0.1532	0.1175	0.2192	0.1445	0.2665	0.1609

tion boosted PRFM, which introduces the attention mechanism into PRFM. We only present the experimental results on $HR@5$ in Table 6 and the experimental results on $NDCG@5$ illustrate a similar trend.

Table 6. Impact of L

Methods	Frappe	Lastfm
NPRFM-0	0.5654	0.2545
NPRFM-1	0.5751	0.2624
NPRFM-2	0.5592	0.2572
NPRFM-3	0.5654	0.2077
NPRFM-attention-0	0.5751	0.2669
NPRFM-attention-1	0.5813	0.2761
NPRFM-attention-2	0.5692	0.2649
NPRFM-attention-3	0.5719	0.2294

Table 7. Impact of k

k	Frappe	Lastfm
NPRFM-16	0.4650	0.1641
NPRFM-32	0.5515	0.2027
NPRFM-64	0.5751	0.2624
NPRFM-128	0.5692	0.2514
NPRFM-attention-16	0.4694	0.1686
NPRFM-attention-32	0.5583	0.2192
NPRFM-attention-64	0.5813	0.2761
NPRFM-attention-128	0.5784	0.2647

As indicated in Table 6, we observe that NPRFM depicts the best performance when the number of the hidden layer is equal to one, and the performance of NPRFM degrades when the number of the hidden layer increases. This is owing to the fact that the training data available are not sufficient enough for NPRFM to accurately learn its model parameters when the number of hidden layers is relatively large. In fact, although the multi-layer perceptron theoretically is able to approximate any measure functions, its premise condition is that there is a sufficient amount of data for the learning of neural network parameters. By contrast, if the number of layers is small, NPRFM has limited ability of modeling the complex interactions among embedded representations of features, resulting in the sub-optimal recommendation performance. We also observe that NPRFM-attention-1 gains the best performance among variants of the attention boosted NPRFM model, and the performance of attention boosted NPRFM degrades as the number of the hidden layer increases or decreases, which is similar to NPRFM. Moreover, the performance of NPRFM-attention-i is better than that of NPRFM-i. The reason is that attention boosted NPRFM can learn the importance of feature interactions to lighten the interference of useless features on interaction.

5.4.2. Impact of k

In this section, we conduct another set of experimental studies to investigate the impact of the dimension of embedded representations of features k on the recommendation quality. We fix the number of the hidden layer to one, and other parameters remain unchanged. We change the value of k within [16,32,64,128]. The experimental results of $HR@5$ on the two datasets are provided in Table 7.

As indicated in Table 7, the proposed NPRFM model is sensitive to the dimensions of the embedded representations of features. We find that the performance of NPFFM is optimal when the dimension of embedded representation of feature is equal to 64. A possible explanation is that the proposed model already has enough expressiveness to describe the latent preferences of user and characteristics of items when $k = 64$. In addition, a large dimension of the embedded representation may introduce noises into NPRFM, degrading the performance of NPRFM. Similar to NPRFM, the performance of attention boosted NPRFM is the best when the dimension of the embedded representation of feature is equal to 64. Meanwhile, we further observe that NPRFM-attention-k performs better than NPRFM-k, owing to the fact that the attention boosted NPRFM is able to learn the importance of useful and predictive feature interactions, thereby further improving the performance.

5.4.3. Impact of parameter λ

The regularization parameter λ may also affect the performance of NPRFM. Hence, we perform another group of experiments to evaluate the sensitivities of λ. We fix the number of the hidden layer to one, $k = 32$, and other parameters remain unchanged. We vary the value of λ to observe the effect of λ on NPRFM. The experimental results of $HR@5$ are presented in Tables 8.

As shown in Table 8, NPRFM achieves its best performance when λ is around 0.001. While, NPRFM performs the worst when λ is around 0.1.

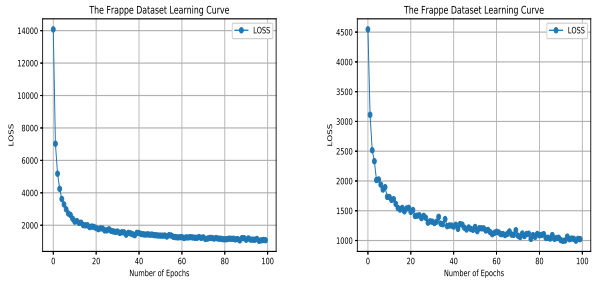
Table 8. Impact of parameter λ

The value of λ	Frappe	Lastfm
$\lambda = 0.1$	0.4742	0.1924
$\lambda = 0.01$	0.5249	0.2006
$\lambda = 0.001$	0.5598	0.2026
$\lambda = 0.0001$	0.5465	0.1972

5.5. Pre-training

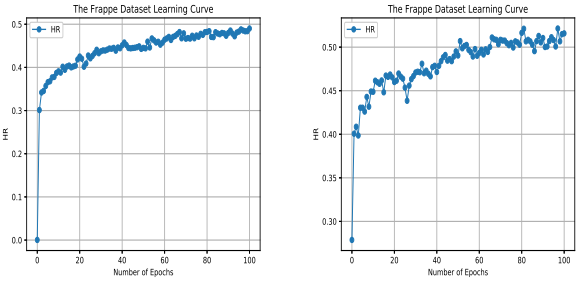
In this section, we investigate the effect of the pre-training on the performance of the NPRFM. We fix the dimension of the embedded representation of feature to 32, and other parameters remain unchanged. Meanwhile, we use the result of PRFM to initialize the corresponding latent feature matrix V defined in NPRFM. The experimental results are presented in Fig. 3 and Fig. 4.

From Fig.3, we observe that the loss function score of NPRFM without pre-training converges from 14000 to 2000 when the number of iterations reaches 20. By contrast, during the first 20 iterations, the loss function of NPRFM with pre-training converges from 4500 to 1500. It indicates that the pre-training is able to accelerate the convergence of NPRFM. Moreover, as shown in Fig. 4, the HR of NPRFM without pre-training is around 0.5 when it iterates over 100 times. Under some conditions, with the pre-training, the HR of NPRFM on



(a) Loss of NPRFM without pre-training (b) Loss of NPRFM with pre-training

Fig. 3. The effect of pre-training on the loss



(a) HR of NPRFM without pre-training (b) HR of NPRFM with pre-training

Fig. 4. The effect of pre-training on HR

Frappe is around 0.54. This observation indicates that the pre-training with PRFM is also helpful to boost the recommendation performance of NPRFM.

6. Conclusion

In this research, we propose the neural pairwsie ranking factorization machine model, which integrates the multi-layer perceptual neural networks into the PRFM model to boost the recommendation performance of factorization model. Specifically, we stack a multi-layer perceptual neural networks over the bi-interaction layer to capture the non-linear and high-order interactions among the embedded representations of features. Meanwhile, the BPR framework is adopted to learn the relative preferences of users, and make non-observed feedback contribute to the inference of model parameters. Hence, our proposed neural pairwise ranking factorization machine model unifies the strength of its three fundamental components, i.e., neural networks, factorization machine and BPR, and effectively tackles their respective drawbacks. In addition, we introduce an attention mechanism into NPRFM to learn the importance of feature interactions. Experimental results on real world datasets indicate that the proposed neural pairwise ranking factorization machine model outperforms the traditional recommendation algorithms. Recently, the generative adversarial network (GAN) (Goodfellow et al., 2014) has shown promising potential in the fields of natural language processing and computer vision, and integrating the GAN into factorization machine would be an interesting future direction.

Declaration of Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this article.

Acknowledgments

This work is supported in part by the Natural Science Foundation of the Higher Education Institutions of Jiangsu Province (Grant No. 17KJB520028), NUPTSF (Grant No. NY217114), Tongda College of Nanjing University of Posts and Telecommunications (Grant No. XK203XZ18002) and Qing Lan Project of Jiangsu Province.

References

Adomavicius, G., Tuzhilin, A., 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. TKDE 17, 734–749.

Adomavicius, G., Tuzhilin, A., 2011. Context-aware recommender systems, in: Recommender systems handbook, pp. 217–253.

Baltrunas, L., Church, K., Karatzoglou, A., Oliver, N., 2015. Frappe: Understanding the usage and perception of mobile app recommendations in-the-wild. arXiv preprint arXiv:1505.03014 .

Baltrunas, L., Ludwig, B., Ricci, F., 2011. Matrix factorization techniques for context aware recommendation, in: RecSys, ACM. pp. 301–304.

Breese, J.S., Heckerman, D., Kadie, C., 1998. Empirical analysis of predictive algorithms for collaborative filtering, in: UAI, pp. 43–52.

Chen, A., 2005. Context-aware collaborative filtering system: Predicting the users preference in the ubiquitous computing environment, in: International Symposium on Location-and Context-Awareness, pp. 244–253.

Chen, C., Zhang, M., Ma, W., Liu, Y., Ma, S., 2020. Efficient non-sampling factorization machines for optimal context-aware recommendation, in: WWW, pp. 2400–2410.

Chen, J., Zhang, H., He, X., Nie, L., Liu, W., Chua, T.S., 2017a. Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention, in: SIGIR, pp. 335–344.

Chen, L., Zhang, H., Xiao, J., Nie, L., Shao, J., Liu, W., Chua, T.S., 2017b. Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning, in: CVPR, pp. 5659–5667.

Cheng, C., Xia, F., Zhang, T., King, I., Lyu, M.R., 2014. Gradient boosting factorization machines, in: RecSys, pp. 265–272.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014. Generative adversarial nets, in: NIPS, pp. 2672–2680.

Guo, H., Tang, R., Ye, Y., Li, Z., He, X., 2017. Deepfm: a factorization-machine based neural network for ctr prediction, in: IJCAI, pp. 1725–1731.

Guo, W., Wu, S., Wang, L., Tan, T., 2016. Personalized ranking with pairwise factorization machines. Neurocomputing 214, 191–200.

He, X., Chua, T.S., 2017. Neural factorization machines for sparse predictive analytics, in: SIGIR, ACM. pp. 355–364.

He, X., Liao, L., Zhang, H., Nie, L., Hu, X., Chua, T.S., 2017. Neural collaborative filtering, in: WWW, pp. 173–182.

Hofmann, T., 2003. Collaborative filtering via gaussian probabilistic latent semantic analysis, in: SIGIR, ACM. pp. 259–266.

Hofmann, T., 2004. Latent semantic models for collaborative filtering. TOIS 22, 89–115.

Hong, F., Huang, D., Chen, G., 2019. Interaction-aware factorization machines for recommender systems, in: AAAI, pp. 3804–3811.

Hornik, K., Stinchcombe, M., White, H., 1989. Multilayer feedforward networks are universal approximators. Neural Networks 2, 359–366.

Juan, Y., Zhuang, Y., Chin, W.S., Lin, C.J., 2016. Field-aware factorization machines for ctr prediction, in: RecSys, ACM. pp. 43–50.

Koren, Y., 2008. Factorization meets the neighborhood: a multifaceted collaborative filtering model, in: SIGKDD, ACM. pp. 426–434.

Koren, Y., Bell, R., Volinsky, C., 2009. Matrix factorization techniques for recommender systems. Computer 42, 30–37.

Lee, D.D., Seung, H.S., 1999. Learning the parts of objects by non-negative matrix factorization. Nature 401, 788–791.

Linden, G., Smith, B., York, J., 2003. Amazon.com recommendations: Item-to-item collaborative filtering. IEEE Internet Computing 7, 76–80.

Lu, Y., Lund, J., Boyd-Graber, J., 2017. Why adagrad fails for online topic modeling, in: EMNLP, pp. 446–451.

Mnih, A., Salakhutdinov, R.R., 2008. Probabilistic matrix factorization, in: NIPS, pp. 1257–1264.

Qiang, R., Liang, F., Yang, J., 2013. Exploiting ranking factorization machines for microblog retrieval, in: CIKM, ACM. pp. 1783–1788.

Rendle, S., 2010. Factorization machines, in: ICDM, IEEE. pp. 995–1000.

Rendle, S., 2012. Factorization machines with libfm. TIST 3, 57.

Rendle, S., Freudenthaler, C., Gantner, Z., Schmidt-Thieme, L., 2009. Bpr: Bayesian personalized ranking from implicit feedback, in: UAI, AUAI Press. pp. 452–461.

Rendle, S., Freudenthaler, C., Schmidt-Thieme, L., 2010. Factorizing personalized markov chains for next-basket recommendation, in: WWW, ACM. pp. 811–820.

Salakhutdinov, R., Mnih, A., Hinton, G., 2007. Restricted boltzmann machines for collaborative filtering, in: ICML, ACM. pp. 791–798.

Sarwar, B.M., Karypis, G., Konstan, J.A., Riedl, J., et al., 2001. Item-based collaborative filtering recommendation algorithms, in: WWW, pp. 285–295.

Suykens, J.A., Vandewalle, J., 1999. Least squares support vector machine classifiers. Neural Processing Letters 9, 293–300.

Xiao, J., Ye, H., He, X., Zhang, H., Wu, F., Chua, T.S., 2017. Attentional factorization machines: Learning the weight of feature interactions via attention networks, in: IJCAI, pp. 3119–3125.

Xin, X., Chen, B., He, X., Wang, D., Ding, Y., Jose, J., 2019. Cfm: convolutional factorization machines for context-aware recommendation, in: IJCAI, AAAI Press. pp. 3926–3932.

Xu, C., Wu, M., 2020. Learning feature interactions with lorentzian factorization machine, in: AAAI, pp. 6470–6477.

Xue, G.R., Lin, C., Yang, Q., Xi, W., Zeng, H.J., Yu, Y., Chen, Z., 2005. Scalable collaborative filtering using cluster-based smoothing, in: SIGIR, ACM. pp. 114–121.

Yan, J., Xu, N.Y., Cai, X.F., Gao, R., Wang, Y., Luo, R., Hsu, F.H., 2010. Lambdarank acceleration for relevance ranking in web search engines, in: Proceedings of the 18th annual ACM/SIGDA international symposium on Field programmable gate arrays, ACM. pp. 285–285.

Yu, K., Zhu, S., Lafferty, J., Gong, Y., 2009. Fast nonparametric matrix factorization for large-scale collaborative filtering, in: SIGIR, ACM. pp. 211–218.

Yu, Y., Wang, C., Gao, Y., Cao, L., Chen, X., 2013. A coupled clustering approach for items recommendation, in: PAKDD, Springer. pp. 365–376.

Yu, Y., Zhang, L., Wang, C., Gao, R., Zhao, W., Jiang, J., 2019. Neural personalized ranking via poisson factor model for item recommendation. Complexity 2019.

Yuan, F., Guo, G., Jose, J.M., Chen, L., Yu, H., Zhang, W., 2016. Lambdafm: learning optimal ranking with factorization machines using lambda surrogates, in: CIKM, ACM. pp. 227–236.

Zhang, J., Huang, T., Zhang, Z., 2019. Fat-deepffm: Field attentive deep field-aware factorization machine, in: ICDM, pp. 43–57.

Zheng, Y., Mobasher, B., Burke, R., 2015. Integrating context similarity with sparse linear recommendation model, in: International Conference on User Modeling, Adaptation, and Personalization, pp. 370–376.