ES_USE: An Efficient Rating Prediction Method

RONG YANG1 AND BING LI2, (Member, IEEE)
1College of Computer Science and Technology, Hubei University of Science and Technology, Xianning 437100, China
2School of Computer Science, Wuhan University, Wuhan 430072, China
Corresponding author: Rong Yang (yr80427@whu.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFB1400602, in part by the Natural Science Foundation of Hubei under Grant 2019CFC888, and in part by the Research Development Fund of Hubei University of Science and Technology under Grant 2019-21GP06.

ABSTRACT In an online shop scenario, learning high-quality product embedding that captures various aspects of the product is important to improve the accuracy of user rating prediction. There is a lot of research about product embedding learning, for example, the side information which is the fusion of user feedback and the appearance of a product. However, because of the diversity of a product’s aspects, taking into account only its appearance as side information is not sufficient to accurately learn its embedding. In this paper, we present a matrix co-factorization method that employs information hidden in the so-called “also-viewed” or “also-bought” products, i.e., a list of products that have also been viewed or have also been bought by a user who has viewed a target product. To improve the accuracy of the rating prediction, our first step is to find out similar users. However, suppose the dataset is very large, e.g., if we have to deal with tens of millions of users’ data, the similarity calculation among users will be very time-consuming. For dealing with this problem, we use a compact binary sketch (i.e. ES, Even Sketch for user similarity estimation) to estimate user similarity. Our experiments demonstrate the superiority of our method in comparison with a state-of-the-art baseline in generating high-accuracy rating prediction.

INDEX TERMS Online shopping, rating prediction, user similarity.

I. INTRODUCTION

Triggered by the Netflix Prize [1] in 2009, which focused on predicting user ratings on movies based on past user feedback, many studies have been addressed on accurately predicting user ratings [2]. However, computing user similarities is a key building block in modern recommender systems. To do this, many works have tried their best to learn a low dimensional embedding of users and items simultaneously [3]–[5]. Moreover, high-quality item embedding yields higher accuracy in user rating prediction [6]. So, various information related to items is considered to learn better item embedding such as product reviews and product images [7]–[9].

In this paper, aiming at learn high-quality product embedding, we talk about the influence of product aspects for users in different product domains. In a real-world online shopping scenario, e.g., Amazon and Taobao, a user may review or rate the product, which he bought. However, the reviews or ratings are almost different for different customers. What leads to these different results? Each user obviously has his or her preferences, which are mainly affected by products’ inherent aspects such as appearance, functionality or specifications [10]. Moreover, the influence extent of each aspect may be different, in different product domains [10]. More specifically, all these phenomena are embodied by “also-viewed” or “also-bought” historic records [10]–[12], which indicate those products that have also been viewed or have also been bought by users when viewing or buying a target product. Generally speaking, users and products can be viewed as sets and set elements. So, for rating prediction, set similarity estimation is one of the key challenges. In this paper, we present a method (detailed in Section 1.2) named ES (Even Sketch for user similarity estimation), which is a compact binary sketch for estimating the Jaccard similarity of users.

A. MOTIVATION

For the sake of clarity, we first detail our motivation. A user may have also viewed a list of other pens when he finally placed an order for one pen. Besides, he has probably bought...
a bottle of ink. Similarly, a user probably buys a set of batteries after having bought a toy car, or to simultaneously buy shoe polish when he has ordered a pair of shoes. Generally speaking, a user may view some similar products before he finally places an order for a certain product, and he may buy some related products at the same time. As a result, when we shop online, we focus the attention on different aspects of products in different product domains.

After having purchased a product, the user may review and rate it from his/her perspective. So, the question is: how to accurately do user rating prediction? To achieve this, most works have considered the appearance of products particularly in visually-aware product domain, but the effort is unevenly in other domains [10]. [10] presented a matrix co-factorization method named Visual Matrix Co-Factorization (VMCF) to improve the user rating prediction accuracy, where the “also-viewed” products were considered but other attributes were overlooked, such as “also-bought” and “bought-together”. In this paper, we address a rating prediction method named ES_USE (as shown in Fig. 1), which is a combination of three types of product attributes, i.e. “also-viewed”, “also-bought” and “bought-together”. Moreover, to effectively estimate the similarities among users in terms of different scale datasets, we present a compact binary sketch to compute the similarities among users (detailed in Fig.2), which is a variant of the Jaccard similarity. By means of this method, we can get a better accuracy but with much less computation and storage.

B. OUR APPROACH

Fig.1 indicates the high-level schema of our approach. Simply speaking, our method mainly consists of three steps. The first step in the process is to perform user similarity calculation. Then, based on the results of the previous step, we find similar users. In the end we leverage “also-viewed” and “also-bought” product records (specifically speaking, these products refer to that the similar users have also viewed or have also bought) to do rating prediction. As shown in Fig.1, we utilize Even Sketch (ES for short, detailed in Fig.2) to get user similarity, which is based on historical “also-viewed”, “bought-together” and “also-bought” records (as shown in Fig.2a). After a real-world online shopping runs over a period of time, there may be a large number of also-viewed, also-bought and bought-together records in the repository. So, user similarity calculation is really time-consuming. Even Sketch is clearly up to solve this problem, and its construction process is shown in Fig.2b. The experimental results in Section 5.3 proved its effectiveness and accuracy.

In this paper, we assume that each set $S$ is summarized in a data structure $D(S)$ of $n'$ bits. To facilitate the understanding, we detail Even Sketch method with an example as follows.

When getting a target product, a user may have also viewed a list of the same kind of products. Besides, he probably buys some related products. For example, a skirt with product ID 0000031852 happened to strike Betty’s fancy. However, before placing the order, Betty may have also viewed the other skirts with product ID list of “B00CEUWY8K, B004FOEEHC, 0000031895, B00BC4GY9Y, B003XRKA7A, B00K18LKX2, B00EM7KAG6, B00AMQI17JA, B00D9C32NI, B002C3Y6WG, B00JLL4L5Y, B003AVNY6I, B008UBQZKU, B00DOWDSD9A, B0613WDQT, B0053BSF5OK, B005C4Y4F6” (AV$_1$ for short), and she may have also...
bought those products with ID list of “B002R0FA24, B00D23MC6W, B00D2K0PA0, B00538F5OK” (AB1 for short). Similarly, for a target skirt, Sophia may have also viewed “0000031895, B00BC4GY9Y, B003XRKA7A, B00K18LKX2, B00EM7KAG6, B00B0AVOS54, B00E95LC8Q, B00GOR92SO, B007ZN5Y56, B00AL2569W, B00B0-8000, B008F05MUC, B00BFXLZ8M” (AV2 for short), and may have also bought “B00D23MC6W, B00D2K0PA0, B007R2RM8W” (AB2 for short). Then, for Betty and Sophia, suppose considering the intersection of the “also viewed” or “also bought” records, there exist the same portion, i.e. two subsets of “0000031895, B00BC4GY9Y, B003XRKA7A, B00K18LKX2, B00EM7KAG6” and “B00D23MC6W, B00D2K0PA0”, respectively. In fact, it is quite possible that Betty and Sophia both have bought a lot of products on a shopping website. In other words, there could be many same subsets on “also viewed” or “also bought” records between Betty and Sophia. Through the experiments in Section 5, we get that these common subsets contribute more on user similarity estimation, which we make full use of to improve the rating prediction accuracy.

However, suppose the dataset is very large, e.g., if we have to deal with tens of millions of users’ records, the calculation of similarities among users will be very time-consuming. To achieve this problem, we present a compact binary sketch, which hints the construction process in Fig.2b. Given two users, let U1 and U2 denote their sets of “also viewed” (or “also bought”) products, respectively. If two sets have common part (e.g. AB1 and AB2), they will be hashed into the same bits, i.e. the bits with the same location number (as showed in the shaded bits of Fig.2b). Moreover, instead of setting a bit to 1, we flip a bit according to the hash value of each element in U1 and U2. Specifically, we utilize the even sketch to hash value, which means that the ith location records the parity of the number of the elements, whose hash values hit location number i.

In summary, the contributions of our paper are three-fold.

1. To the best of our knowledge, few existing works considered information hidden in the so-called “also-viewed” or “also-bought” products in user rating prediction scenario. We recognize the substantial significance of the hidden information and specify the problem formally.

2. We employ a compact binary sketch (i.e. ES, Even Sketch for user similarity estimation) to estimate user similarity, which can deal with huge amounts of data and can greatly reduce computation time.

3. A wide range of experiments are performed on multiple public real-world datasets to validate the feasibility of our proposal. Experiment results indicate that ES_USE achieves high-accuracy rating prediction.

The remainder of this paper is organized as follows. In Section II, we discuss the related work relevant to our research. Then we formalize the used notions in Section III before we detail our approach in Section IV. We present the experiments in Section V and conclude this work in Section VI.

II. RELATED WORKS

Rating prediction is a hot spot in the research of recommender systems. There are many studies about how to utilize various side information, which is related to products. For example, Tong et al. [13] presented a rating prediction algorithm named NF-SVM based on the analysis of users’ natural noise and relationships. It firstly clustered users to sharpen the similarity attribute among users, then analyzed users’ rating history to obtain the attributes of users’ natural noise. For recommendation, in [14], Shi et al. addressed the correlation between expenditures and rating scores, such that an Expenditure aware Rating Prediction method (EARP) was presented, which was based on low-rank matrix factorization. Pradhan et al. [16] utilized multi-view clustering to cluster users or items leveraging information from multiple modalities, by which the accuracy of CF-based rating prediction systems was improved. A framework named AB (Attribute Boosting) was presented in [17], which took full advantage of the interactions with attributes from user aspect, item aspect, and attribute type aspect, respectively. Similarly, Pan et al. [24] presented a novel approach to cluster Mashups into groups, which integrates structural similarity and semantic similarity using fuzzy AHP (fuzzy analytic hierarchy process).

Meanwhile, some novel methods have been proposed to contribute to the rating prediction. In [15], Wu combined the rough sets with the back propagation neural network to do audience rating prediction, which can predict complicated audience ratings with dynamic and non-linear factors. By means of modeling ratings over each user’s latent interests and also each item’s latent topics, Harvey et al. [18] presented a Bayesian latent variable model for rating prediction. In [18], Harvey firstly described a Gibbs sampling procedure for estimating parameters that can compete with the gradient descent SVD methods commonly used in state-of-the-art systems, then extended the model so that rating estimation was improved significantly by means of user-dependent and item-dependant biases. For dealing with the cold-start users/items problem, [19] addressed a rating comparison strategy (RAPPARE) to learn the latent profiles of cold-start users/items, which focused on providing a fine-grained calibration on these latent profiles by exploring the differences between cold-start and existing users/items. Similarly, for the prediction accuracy, [20] addressed a sentiment-based rating prediction method (RPS), which not only calculated each user’s sentiment on items/products, but also took into account interpersonal sentimental influence and product reputation.

In [12], [21], McAuley et al. leveraged the “also-viewed” product information to recommend visually alternative products in clothing domain as a link prediction task. However, other product information was overlooked, and they only referred to the clothing domain. A similar work was presented in [10]. It addressed “also-viewed” product information
TABLE 1. Notations.

| Symbol  | Description |
|---------|-------------|
| $U, P$  | User set($U = \{u_1, u_2, \ldots, u_n\}$), Product set($P = \{p_1, p_2, \ldots, p_m\}$) |
| $R$     | Rating matrix ($n \times m$) |
| $M_{av}$| Also-viewed product-affinity matrix ($m \times m$) |
| $M_{ab}$| Also-bought product-affinity matrix ($m \times m$) |
| $K$     | Num. dimensions of product embedding |
| $V_k$   | \begin{align*} & \text{"Also-viewed" product embedding for product} \\ & k (K \times 1) \end{align*} |
| $B_k$   | \begin{align*} & \text{"Also-bought" product embedding for product} \\ & k (K \times 1) \end{align*} |
| $n'$    | The size of Even Sketch |
| $m'$    | The size of "also viewed"/"also bought"/"bought-together" products |
| $J$     | Jaccard similarity |
| $i_i$   | The parity of the number of set items that hash to the $i$th location |
| $h$     | $h: \Omega \rightarrow \{0, 1\}$ is a random hash function |
| $N$     | The number of ratings |

for rating prediction, where all the product domains were allowed. In this paper, we also leverage rating prediction, but we consider “also-viewed”, “bought-together” and “also-bought” product information simultaneously, which can improve user rating prediction accuracy in general domains.

### III. PROBLEM FORMULATION

#### A. NOTATIONS

The notations of this paper are defined in Table 1.

Assume that there are $n$ users and $m$ products in an online shopping scenario, $U = \{u_1, u_2, \ldots, u_n\}$ and $P = \{p_1, p_2, \ldots, p_m\}$ are the sets of users and products respectively. Rating matrix $R = \{r_{ij}\}_{n \times m}$ denotes the ratings assigned by users to products, e.g., the element $r_{ij}$ of $R$ means that user $i$ rates product $j$ with value $r_{ij}$. In different application scenario, $r_{ij}$ can be either a real number or a binary value [10]. In this paper, the datasets are extracted from Amazon.com by McAuley et al., where $r_{ij}$ in the range $[1, 5]$.

#### B. CONSTRUCTING PRODUCT-AFFINITY NETWORK

In this paper, we build two networks to embody “also viewed” and “also bought” product information respectively. Also-viewed product-affinity network is similar to Fig.3, whose nodes denote products and the directed edges show the “also viewed” relationships among products. In this product-affinity network, neighboring products share common product information such as appearance, functionality and specifications. When a customer wants to buy a pair of leisure shoes, he or she may view several pairs of other shoes, whose manufacturers, styles, even the price may be different. Moreover, after having placed an order for a pair of target shoes, he may probably buy a shoe brush, or a target jacket then a shirt. These purchase habits indicate the also-bought relationship among products, which can be described by means of Fig.4. In Fig.3 and Fig.4, the edges are directed from a target product to its “also viewed” or “also bought” products respectively, and obviously these two relationships are not symmetric. This paper uses two product-affinity matrices to represent the “also viewed” and “also bought” relationships, i.e. also-viewed product-affinity matrix $AV = [av_{ij}]_{m \times m}$ and also-bought product-affinity matrix $AB = [ab_{ij}]_{m \times m}$, where $[av_{ij}] = 1$ or $[ab_{ij}] = 1$ indicates that the product $j$ is the “also viewed” or “also bought” product of product $i$ respectively.

### IV. METHOD

#### A. MODELING RATING

Given an online shopping scenario consisting of $n$ users and $m$ products, the relationships between users and products are denoted by a user-product matrix named $R = \{r_{ij}\}_{n \times m}$. Every entry in this matrix $r_{ij}$ represents the rating of user $i$ for product $j$. $U \in \mathbb{R}^{K \times n}$ and $P \in \mathbb{R}^{K \times m}$ denote user and product embedding matrices, with column vectors $U_i$ and $P_j$ denoting user-specific and product-specific embedding vectors, respectively.

The conditional distribution over the observed ratings is as follows:

$$P(R|U, P, \sigma_R) = \prod_{i=1}^{n} \prod_{j=1}^{m} f(r_{ij}|g(U_i^T P_j), \sigma_R)^{I_{ij}}$$  \hspace{1cm} (1)$$

where $f(x|\mu, \sigma^2)$ denotes the probability density function of a Gaussian distribution with mean $\mu$ and variance $\sigma^2$, and $I_{ij}$ equals to 1 if user $i$ rated product $j$, and 0 otherwise. The range of $U_i^T P_j$ is restricted within $[0, 1]$ by means of the logistic function $g(\cdot)$. For the sake of simplicity, we also
convert rating $r_{ij}$ into $[0, 1]$. The hidden variables are placed zero-mean spherical Gaussian priors [22] as follows:

$$P(U|\sigma_U^2) = \prod_{i=1}^{n} f(U_i|0, \sigma_U^2 I)$$

(2)

$$P(P|\sigma_P^2) = \prod_{j=1}^{m} f(P_j|0, \sigma_P^2 I)$$

(3)

Based on Eqs. 2 and 3, the log-posterior distribution over the hidden variables is computed by the following formula:

$$P(U, P|R, \sigma_U^2, \sigma_P^2) \propto \mathcal{P}(R|U, \sigma_U^2)\mathcal{P}(U|\sigma_U^2)\mathcal{P}(P|\sigma_P^2)$$

$$= \prod_{i=1}^{n} \prod_{j=1}^{m} [f(r_{ij}|g(U_i^T P_j)), \sigma_U^2]^{I_{jk}^R}$$

$$\times \prod_{i=1}^{n} f(U_i|0, \sigma_U^2 I) \times \prod_{j=1}^{m} f(P_j|0, \sigma_P^2 I)$$

(4)

### B. MODELING ALSO-VIEWED RELATIONSHIPS

For the also-viewed product-affinity matrix $M_{avp}$, its conditional distribution over the observed also-viewed product-affinity matrix is defined as:

$$P(M_{avp}|P, V, \sigma_{M_{avp}}^2, \sigma_V^2)$$

$$= \prod_{j=1}^{m} \prod_{k=1}^{K} [f(M_{avp}^{jk}|g(P_j^T V_k)), \sigma_{M_{avp}}^2]^{I_{jk}^{M_{avp}}}$$

(5)

where $V \in \mathbb{R}^{K \times m}$ is the “also-viewed” product embedding matrix, with $V_k$ denoting the $k$th “also-viewed” product-specific column embedding vector for product $k$. Given product $j$ and $k$, their “also-viewed” relationship is modeled by $P_j^T V_k$. $I_{jk}^{M_{avp}}$ equals to 1 if product $k$ belongs to one of the “also-viewed” products of product $j$, and 0 otherwise. Similarly, we place zero-mean spherical Gaussian priors on the hidden $P$ and $V$ as:

$$P(P|\sigma_P^2) = \prod_{j=1}^{m} f(P_j|0, \sigma_P^2 I),$$

$$P(V|\sigma_V^2) = \prod_{k=1}^{K} f(V_k|0, \sigma_V^2 I)$$

(6)

Thus, the log-posterior distribution over the hidden variables is computed by Eq.7:

$$P(P, V|M_{avp}, \sigma_P^2, \sigma_V^2, \sigma_{M_{avp}}^2)$$

$$\propto P(M_{avp}|P, V, \sigma_{M_{avp}}^2)\mathcal{P}(P|\sigma_P^2)\mathcal{P}(V|\sigma_V^2)$$

$$= \prod_{j=1}^{m} \prod_{k=1}^{K} [f(M_{avp}^{jk}|g(P_j^T V_k)), \sigma_{M_{avp}}^2]^{I_{jk}^{M_{avp}}}$$

$$\times \prod_{j=1}^{m} f(P_j|0, \sigma_P^2 I) \times \prod_{k=1}^{K} f(V_k|0, \sigma_V^2 I)$$

(7)

### C. MODELING ALSO-BOUGHT RELATIONSHIPS

Similar to the also-viewed relationships, the conditional distribution of the also-bought product-affinity matrix $M_{abp}$ is shown on Eq.8.

$$P(M_{abp}|P, B, \sigma_{M_{abp}}^2, \sigma_B^2)$$

$$= \prod_{j=1}^{m} \prod_{k=1}^{K} [f(M_{abp}^{jk}|g(P_j^T B_k)), \sigma_{M_{abp}}^2]^{I_{jk}^{M_{abp}}}$$

(8)

where $B \in \mathbb{R}^{K \times m}$ is the “also bought” product embedding matrix, with $B_k$ denoting the $k$th “also bought” product-specific column embedding vector for product $k$. $P_j^T B_k$ denotes the “also bought” relationship between product $j$ and $k$. $I_{jk}^{M_{abp}}$ equals to 1 if product $k$ belongs to one of the “also bought” products of product $j$, and 0 otherwise.

The zero-mean spherical Gaussian priors on the hidden $P$ and $B$ are defined by Eq.9.

$$P(P|\sigma_P^2) = \prod_{j=1}^{m} f(P_j|0, \sigma_P^2 I),$$

$$P(B|\sigma_B^2) = \prod_{k=1}^{m} f(B_k|0, \sigma_B^2 I)$$

(9)

At last, the log-posterior distribution over the hidden variables is computed by Eq.10.

$$P(P, B|M_{abp}, \sigma_P^2, \sigma_B^2, \sigma_{M_{abp}}^2)$$

$$\propto P(M_{abp}|P, B, \sigma_{M_{abp}}^2)\mathcal{P}(P|\sigma_P^2)\mathcal{P}(B|\sigma_B^2)$$

$$= \prod_{j=1}^{m} \prod_{k=1}^{K} [f(M_{abp}^{jk}|g(P_j^T B_k)), \sigma_{M_{abp}}^2]^{I_{jk}^{M_{abp}}}$$

$$\times \prod_{j=1}^{m} f(P_j|0, \sigma_P^2 I) \times \prod_{k=1}^{m} f(B_k|0, \sigma_B^2 I)$$

(10)

### D. EVEN SKETCH

As showed in Fig.2, we use a compact binary sketch (i.e. ES, Even Sketch for user similarity estimation) to estimate user similarity. The Even Sketch consists of an array $l$ of $n' > 2$ bits. Without the loss of generality, we assume the $i$th bit can be formally described as follows:

$$l_i = \oplus_{a \in S} 1_{h(a) = i}$$

where $h : \Omega \rightarrow [n']$ is a random hash function, and $0 \leq i < n'$. More specifically, $l_i$ is the parity of the number of set items that hash to the $i$th location.

Based on the Even Sketch, we can greatly reduce the storage space, especially when we deal with large-scale datasets, since we just record the parity of the number of elements, which hash to a location. Moreover, we further handle the Even Sketch values of two sets by means of an exclusive-or operation, thus the public portion is omitted. For notational convenience we will think of $E(S_1)$ and $E(S_2)$ as the sets of bit positions containing 1, which means that their exclusive-or corresponds exactly to the symmetric difference.
Let \( m' \) and \( n' \) be the size of set \( S \) (e.g. set of “also viewed” products) and the size of \( ES(S) \) in bits, respectively. In this work, hash functions are fully random. So, the process of constructing \( ES(S) \) can be referred as a voting problem. Assume that there are \( m' \) voters and \( n' \) candidates, we calculate the parity of the number of votes for each candidate.

We consider the parity of number of votes for any candidate as a simple two-state Markov chain model. The first/second state corresponds to the odd/even parity. As the probability of changing states is \( 1/n' \), let \( p_i \) be the probability that any specific candidate has an even number of votes after \( i \) votes have cast their ballots, we can get \( p_i \) as follows

\[
p_i = \frac{1 + (1 - 2/n')^i}{2}
\]

(11)

Generally speaking, let \( X_i \) be a 0-1 random variable corresponding to the parity of the number of voters who voted for the \( i \)th candidate, then

\[
X = \sum_i X_i
\]

\[
E[X] = n' \frac{1 + (1 - 2/n')m'}{2}
\]

(12)

Assume that \( Z \approx E[X] \), in which case

\[
z \approx n' \frac{1 + (1 - 2/n')m'}{2}
\]

(13)

Finally we can get an estimate \( m' \) by

\[
\hat{m}' = \frac{\ln(z) - 1}{\ln(1 - 2/n')}
\]

(14)

Given \( S_1 \) and \( S_2 \), through the method detailed in Fig.2, we can construct Even Sketches \( ES(S_1) \) and \( ES(S_2) \). To improve the accuracy, we should increase the number of independent permutations (i.e. variable \( k \)). We describe the symmetric set difference of \( S_1 \) and \( S_2 \) as \( S_1 \Delta S_2 \), then

\[
E[|S_1 \Delta S_2|] = 2k(1 - J)
\]

(15)

where \( J \) is the Jaccard similarity of \( S_1 \) and \( S_2 \). For simplicity, we think of \( ES(S_1) \) and \( ES(S_2) \) as the sets of bit positions containing 1. So, we can get the symmetric difference of \( S_1 \) and \( S_2 \) by means of exclusive-or of \( ES(S_1) \) and \( ES(S_2) \). Let \( |S_1 \Delta S_2| \) be the estimate of \(|S_1 \Delta S_2|\), then

\[
|S_1 \Delta S_2| \approx n' \frac{1}{2} \ln(1 - \frac{2|ES(S_1) \Delta ES(S_2)|}{n'})
\]

(16)

Here \( |ES(S_1) \Delta ES(S_2)| \) denotes the number of 1s in the structure. Let \( |S_1 \Delta S_2| \) as a proxy for \( E[|S_1 \Delta S_2|] \), the Jaccard similarity can be estimated as follows:

\[
\hat{J}ES = 1 - \frac{|S_1 \Delta S_2|}{2k} = 1 + \frac{n' \ln(1 - \frac{2|ES(S_1) \Delta ES(S_2)|}{n'})}{4k}
\]

(17)

E. OBJECTIVE FUNCTION

In the previous sections, we described how to independently model the ratings, “also-viewed” relationships and “also-bought” relationships among products. Moreover, we detailed how to estimate user similarity. Maximizing the log-posterior over the hidden variables is equivalent to minimizing the following objective function [10], where fixed hyper-parameters (i.e., the observation noise variances and prior variances) are used.

\[
L(R, M_{avg}, M_{abp}, U, P, V, B) = \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{m} I_{ij}^D (r_{ij} - g(U_i^T P_j))^2 + \frac{\lambda_{M_{avg}}}{2} \sum_{i=1}^{m} \sum_{k=1}^{m} M_{avg}^i (M_{avg}^i - g(P_j^T V_k))^2 + \frac{\lambda_{M_{abp}}}{2} \sum_{j=1}^{m} \sum_{k=1}^{m} M_{abp}^j (M_{abp}^j - g(P_j^T B_k))^2 + \frac{\lambda_U}{2} \|U\|_F^2 + \frac{\lambda_P}{2} \|P\|_F^2 + \frac{\lambda_V}{2} \|V\|_F^2 + \frac{\lambda_B}{2} \|B\|_F^2
\]

(18)

where \( \lambda_{M_{avg}} = \frac{\sigma^2}{\sigma_{M_{avg}}^2}, \lambda_{M_{abp}} = \frac{\sigma^2}{\sigma_{M_{abp}}^2}, \lambda_U = \frac{\sigma^2}{\sigma_U^2}, \lambda_P = \frac{\sigma^2}{\sigma_P^2}, \lambda_V = \frac{\sigma^2}{\sigma_V^2}, \lambda_B = \frac{\sigma^2}{\sigma_B^2} \) and \( \|\cdot\|_F \) denotes the Frobenius norm. The importance of “also-viewed” and “also-bought” products in the unified model can be regulated by \( \lambda_{M_{avg}} \) and \( \lambda_{M_{abp}} \) respectively. For the non-convex objective function showed in Eq.19, by means of computing the gradient of each embedding variable, i.e., \( U, P, V, B \), we can learn and obtain a local minimum solution.

F. COMPLEXITY ANALYSIS

ItemCF [23] is a basic item-based collaborative filtering algorithm. For generating predictions for a user \( u \) on item \( i \), it first retrieves the pre-computed \( K \) most similar items corresponding to the target item \( i \). Then it finds how many of those \( K \) items were purchased by the user \( u \). PMF [22] utilizes regularization parameters to provide a more flexible method to regularization, its main drawback is computationally expensive, since instead of training a single model it has to train a multitude of models. As for VMCF [10], it obtains a total complexity of \( O(\rho(K + D) + \mu K) \) (\( \rho \) denotes the average number of observed ratings in rating matrix, and \( \mu \) denotes the average number of observed elements in product-affinity matrix), where \( K \) and \( D \) are Num. dimensions of product embedding and Num. dimensions of product visual embedding, respectively. Our method mainly consists of three steps. The first step is to perform user similarity calculation by means of ES (Even Sketch for user similarity estimation), and this step has complexity \( O(n \times m_1 + n \times m_2 + n \times m_3) \) (\( m_1, m_2 \) and \( m_3 \) denote the average numbers of “also-viewed”, “bought-together” and “also-bought” products, respectively). Then, for a user \( u \), we find its similar users, and this incurs complexity \( O(n) \). Finally, based on those similar users (actually, our method can effectively reduce the
searching space), we do rating prediction. The last step takes complexity $O(§ \times K + J \times K + E \times K)$ ($§$ denotes the average number of observed ratings which the similar users rate, $J$ denotes the average number of observed elements in also-viewed product-affinity matrix, and $E$ denotes the average number of observed elements in also-bought product-affinity matrix). The above analysis indicates that the complexity of ES_USE is a linear value. Moreover, $§ \ll \rho$, $m_1 \ll m$, $m_2 \ll m$ and $m_3 \ll m$, and they contribute to the high-efficiency of ES_USE.

V. EXPERIMENTS

A. DATASETS

In this section, we conduct experiments to evaluate our approach. We compare our ES_USE with several state-of-the-art methods on multiple public real-world datasets, which are extracted from Amazon.com by McAuley et al. [12]. The datasets contain product reviews (i.e., ratings, text, helpfulness votes) and product metadata. Specifically, the metadata includes price, title, a list of also viewed products, a list of also bought products, and so on. We use the lists of also viewed products and also bought products to construct also-viewed product-affinity matrix and also-bought product-affinity matrix, respectively. We preprocess all datasets so that each user rated at least four products. And by the way, every product in the product-affinity network must exist in the user ratings data. Table 2 details the statistics of our datasets, which include four datasets, i.e. books, baby, office products and pet supplies. In Figure 5, the number of rated products in each dataset is counted, respectively.

We compare our ES_USE with ItemCF, PMF and VMCF, introduced in [23], [22], and [10], respectively. ItemCF is a traditional recommendation method based on the similarity of products, and it uses Pearson’s correlation coefficient as similarity metric. However, PMF considers only user ratings information, which is a matrix factorization-based recommendation method. As for VMCF, it leverages “also-viewed” products to do rating prediction. In order to present a clear understanding of baseline methods, we detail a summary of their properties in Table 3.

B. EXPERIMENTAL SETTINGS

For comparison, we employ MSE (Mean Squared Error) [10], MAE (Mean Absolute Error) [26] and F1 score [25] to evaluate the performance of user rating prediction.

$$MSE = \frac{\sum_{i,j} (r_{ij} - \hat{r}_{ij})^2}{N}$$

(19)

$$MAE = \frac{\sum_{i,j} |r_{ij} - \hat{r}_{ij}|}{N}$$

(20)

where $r_{ij}$ denotes the rating that user $i$ rated product $j$, $\hat{r}_{ij}$ denotes the corresponding predicted value, and $N$ denotes the number of ratings. Considering our method detailed by the equations above, we can get a better performance with a lower MSE or MAE.

We divide the user ratings datasets into two parts, one (i.e. 70% of the user ratings datasets) for training and the other for testing (i.e. the remaining 30%). For ItemCF, we set the number of neighbors to 20. For all other methods, we carry out grid search with $\lambda_U$, $\lambda_P \in \{0.01, 0.1, 0.2, 0.5, 1.0\}$, $\lambda_{Mavp}$, $\lambda_{Mabp} \in \{0.01, 0.1, 0.2, 0.5, 1.0\}$ and $\lambda_V, \lambda_B \in \{0.01, 0.1\}$ while $K$ is fixed to 5.

C. EXPERIMENTAL RESULTS

Figures 6-9 present the evaluation results on each dataset in terms of MSE. We observe that in all product domains (i.e. Books, Baby, Office Products and Pet Supplies), ES_USE obtains smaller MSE values, which means better prediction accuracy. This indicates that the prediction accuracy of ES_USE is improved by modeling the “also-viewed” and “also-bought” relationships. Moreover, the Even Sketch (i.e. a compact binary sketch for estimating the Jaccard similarity of users) is incorporated into ES_USE, which in turn yields further improvements in user rating prediction accuracy.

Figures 10-12 detail the influence of balancing parameters $\lambda_{Mavp}$ and $\lambda_{Mabp}$, which are hyper-parameters that regulate the importance of “also-viewed” and “also-bought” products in...
the unified model, respectively. In our experiments, we set 
\[ \lambda = \alpha \times \lambda_{M_{avp}} + (1-\alpha) \times \lambda_{M_{abp}}, \]
where \( \alpha \) is the weight of \( \lambda_{M_{avp}} \). When \( \lambda_{M_{avp}} = 0 \), we ignore “also-viewed” product information but exploit “also-bought” product information, and vice versa. Observing from Figures 10-12, we draw the conclusion that incorporating “also-viewed” and “also-bought” product information indeed improves user rating prediction accuracy, and the optimal value of \( \lambda \) is different in each product domain, which is mostly a value in the range \([0.0, 1.0]\).

In the third set of experiments, we try to evaluate the precision ratio for our ES_USE sketch on four datasets, respectively. Figures 13-16 show the experimental results on books, baby, office products and pet supplies, respectively. On the whole, ES_USE has a high precision ratio. Under the different sizes of ES_USE (i.e. the size of Even Sketch in bits (n’)), it almost has a precision above 85% with different size scales. As shown in Figures 13-16, when the size of ES_USE varies from 100 bits to 600 bits, the precision ratio of the four datasets all grows. Actually, if bigger the size of Even Sketch is, more accurate Even Sketch will be. Generally speaking, “also-viewed” and “also-bought” records can be hashed into a bigger bits space, and this can reduce conflict probability. ES_USE obtains higher relative precision ratio on Books and
PET supplies datasets. ES_USE achieves up to 92% and 90% on Books and PET supplies, respectively.

In addition, the purpose of the fourth set of experiments is to verify the effects of user similarity assessment on ES_USE. As we can see from Figures 17-18, ES_USE-Y and ES_USE-N both have a high precision ratio, but ES_USE-Y is better. Especially when \( n' = 1024 \), ES_USE-Y is 9.57% superior to ES_USE-N, though the precision ratio of ES_USE-N is above 0.82 on the four datasets. It is clear that ES_USE-Y performs more accurately.

However, in the fifth set of experiments, we verify the precision performance of ES_USE for different Jaccard.
similarities on the Books dataset, where the size of Even Sketch (i.e. n’) is fixed at 512 or 1024 bits. As shown in Figures 19-20, when the Jaccard similarity of users varies from 0.7 to 0.9, the precision ratio has a slow growth in both of the above scenarios. It is worth noting that the performance can be improved if we double the size of Even Sketch, i.e. n’ from 512 to 1024 bits. The influence of user similarity assessment on ES_USE is further confirmed by this set of experiments. Though the size of Even Sketch influences the accuracy of user similarity, it is not always that a higher value is better. Through our experiments, it is concluded that about 800 bits is the optimal value.

To further evaluate the performance of our approach, we make use of MAE metric (the smaller the better) in the sixth set of experiments. The size of ES_USE (i.e. n’) is fixed at 1024 bits. The experiment results are shown in Fig.21. Concretely, on the four datasets, our proposed ES_USE approach outperforms the rest three ones in terms of prediction accuracy, which is due to the inherent nature of our adopted Even Sketch technique (i.e., a compact binary sketch for user similarity estimation). Besides, this set of experiment results is consistent with those of Figures 6-9.

Fig.22 shows that the F1 value of ES_USE-Y (user similarity assessment involved) approach is larger than other approaches. The F1 value is defined as the mean of the accuracy rate and recall rate, and it is a more objective system performance evaluation index. So, it is considered that the accuracy of ES_USE-Y has improved compared with other approaches. The prediction method which is combining the “also-viewed”, “also-bought” and the “bought-together” product attributes. Moreover, user similarity assessment can make more comprehensive predictions for users.

Finally, we verify the F1 performance of ES_USE for different sketch size on the Books dataset. As shown in Fig.23, when the size of Even Sketch (i.e. n’) varies from 100 to 1000 bits, the F1 score has a slow growth. It has the maximum value when n’ holds 800 bits. From this set of results we summarize as follows. Firstly, the Even Sketch can contribute a great deal to the rating prediction accuracy, which can estimate the Jaccard similarity among users. Secondly, a bigger Even Sketch size means a better prediction accuracy, but it is not always the case (about 800 bits is the optimal value).

D. DISCUSSION AND ANALYSIS

By these experiments conducted on a large scale dataset, we firstly find that ES_USE is the most accurate method among ItemCF, PMF and VMCF. Specifically, by means of Even Sketch (ES for short), we can estimate the Jaccard
try our best to improve the accuracy and scalability of Even Sketch, since it influences heavily the performance of the whole system.

**ACKNOWLEDGMENT**

The authors would like to thank the anonymous reviewers for their careful work and thoughtful suggestions that will help them improve this paper substantially.

**REFERENCES**

[1] J. Bennett and S. Lanning, “The Netflix prize,” in *Proc. KDD Cup Workshop*, 2007, p. 35.
[2] Y. Koren, “Factorization meets the neighborhood: A multifaceted collaborative filtering model,” in *Proc. KDD*, Las Vegas, NV, USA, 2008, pp. 426–434.
[3] U. Paquet and N. Koenigstein, “One-class collaborative filtering with random graphs,” in *Proc. WWW*, Rio de Janeiro, Brazil, 2013, pp. 999–1008.
[4] Y. Koren, R. Bell, and C. Volinsky, “Matrix factorization techniques for recommender systems,” *Computer*, vol. 1, no. 8, pp. 30–37, Aug. 2009.
[5] R. Salakhutdinov and A. Mnih, “Bayesian probabilistic matrix factorization using Markov chain Monte Carlo,” in *Proc. ICML*, Helsinki, Finland, 2008, pp. 880–887.
[6] O. Barkan and N. Koenigstein, “ITEM2VEC: Neural item embedding for collaborative filtering,” in *Proc. MSLP*, Vietri sul Mare, Italy, 2016, pp. 1–6.
[7] D. Kim, C. Park, J. Oh, S. Lee, and H. Yu, “Convolutional matrix factorization for document context-aware recommendation,” in *Proc. RecSys*, New York, NY, USA, 2016, pp. 233–240.
[8] C. Wang and D. M. Blei, “Collaborative topic modeling for recommending scientific articles,” in *Proc. KDD*, San Diego, CA, USA, 2011, pp. 448–456.
[9] H. Wang, N. Wang, and D.-Y. Yeung, “Collaborative deep learning for recommender systems,” in *Proc. KDD*, Sydney, NSW, Australia, 2015, pp. 1235–1244.
[10] C. Park, D. Kim, J. Oh, and H. Yu, “Do ‘also-viewed’ products help user rating prediction?” in *Proc. WWW*, Perth, WA, Australia, 2017, pp. 1113–1122.
[11] R. He and J. McAuley, “Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering,” in *Proc. WWW*, Montreal, QC, Canada, 2016, pp. 507–517.
[12] J. McAuley, C. Targett, G. Shi, and A. van den Hengel, “Image-based recommendations on styles and substitutes,” in *Proc. SIGIR*, Santiago, Chile, 2015, pp. 43–52.
[13] C. Tong, Y. Lian, J. Niu, and X. Long, “A novel rating prediction method based on user relationship and natural noise,” *Multimedia Tools Appl.*, vol. 77, no. 2, pp. 1–16, 2017.
[14] C. Shi, B. He, M. Zhang, F. Zhuang, P. S. Yu, and N. Guo, “Expenditure aware rating prediction for recommendation,” in *Proc. Big Data*, Washington, DC, USA, 2016, pp. 1018–1025.
[15] M. M. Wu, Y. Wang, and X. L. Liu, “Application of rough sets in audience rating prediction,” *Comput. Model. New Technol.*, vol. 18, no. 12, pp. 521–525, 2014.
[16] L. Pradhan, C. Zhang, and P. Chitrakar, “Multi-view clustering in collaborative filtering based rating prediction,” in *Proc. ICSIC*, Laguna Hills, CA, USA, 2016, pp. 250–253.
[17] L. Zheng, F.-X. Zhu, and X. Yao, “Recommendation rating prediction based on attribute boosting with partial sampling,” *Chin. J. Comput.*, vol. 39, no. 8, pp. 1501–1514, Aug. 2016.
[18] M. Harvey, M. J. Carman, I. Ruthven, and F. Crestani, “Bayesian latent variable models for collaborative item rating prediction,” in *Proc. CIKM*, Glasgow, U.K., 2011, pp. 699–708.
[19] J. Xu, Y. Yao, H. Tong, X. Tao, and J. Lu, “RaPare: A generic strategy for cold-start rating prediction problem,” *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 6, pp. 1296–1309, Jun. 2017.
[20] X. Lei, X. Qian, and G. Zhao, “Rating prediction based on social sentiment from textual reviews,” *IEEE Trans. Multimedia*, vol. 18, no. 9, pp. 1910–1921, Sep. 2016.
[21] J. McAuley, R. Pandey, and J. Leskovec, “Inferring networks of substitutable and complementary products,” in *Proc. KDD*, Sydney, NSW, Australia: Hilton Sydney, 2015, pp. 785–794.
[22] R. Salakhutdinov and A. Mnih, “Probabilistic matrix factorization,” in Proc. ICONIP, Kitakyushu, Japan, 2007, pp. 1–8.
[23] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, “Item-based collaborative filtering recommendation algorithms,” in Proc. WWW, Hong Kong, 2001, pp. 285–295.
[24] W. Pan, X. Xu, H. Ming, and C. K. Chang, “Clustering mashups by integrating structural and semantic similarities using fuzzy AHP,” Int. J. Web Services Res., vol. 18, no. 1, pp. 34–57, Jan. 2021, doi: 10.4018/IJWSR.2021010103.
[25] G. Forman, “An extensive empirical study of feature selection metrics for text classification,” J. Mach. Learn. Res., vol. 3, pp. 1289–1305, Mar. 2003.
[26] Z. Zheng, H. Ma, M. R. Lyu, and I. King, “WSRec: A collaborative filtering based web service recommender system,” in Proc. IEEE Int. Conf. Web Services, Jul. 2009, pp. 437–444, doi: 10.1109/ICWS.2009.30.

RONG YANG received the Ph.D. degree from the School of Computer Science, Wuhan University, in 2015. Since 2015, he has been with the School of Computer Science and Technology, Huabei University of Science and Technology (HUST). His research interests include services computing and software engineering.

BING LI (Member, IEEE) received the B.E. degree from the Department of Computer Science, the M.S. degree, and the Ph.D. degree from the Computer Science School, Huazhong University of Science and Technology (HUST), in 1990, 1997, and 2003, respectively. He has worked as a Postdoctoral Researcher with the State Key Laboratory of Software Engineering (SKLSE) & School of Computer, from 2003 to 2005. He is currently a Professor with the School of Computer Science, Wuhan University. He is a Senior Member of the China Computer Federation (CCF). He serves as a member in three CCF Technical Committees such as open systems, software engineering, and service computing. In recognition of his contributions to scientific and technological progress, he received Second Class Award of the Chinese National Scientific and Technological Progress Award (2009), First Class Award of Hubei Province Scientific and Technological Progress Award (2008–2011), and First Class Award of the Natural Sciences Award of the Chinese Ministry of Education (2007).