Knowledge-Based Systems 85 (2015) 307–315

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Multi-linear interactive matrix factorization

Lu Yu, Chuang Liu, Zi-Ke Zhang *

Alibaba Research Center for Complexity Sciences, Hangzhou Normal University, Hangzhou 311121, PR China

ARTICLE INFO

Article history:
Received 7 April 2014
Received in revised form 2 April 2015
Accepted 16 May 2015
Available online 23 May 2015

Keywords:
Recommender systems
Collaborative filtering
Matrix factorization
Latent factor model
Time-aware recommendation

ABSTRACT

Recommender systems, which can significantly help users find their interested items from the information era, has attracted an increasing attention from both the scientific and application society. One of the widest applied recommendation methods is the Matrix Factorization (MF). However, most of MF based approaches focus on the user-item rating matrix, but ignoring the ingredients which may have significant influence on users' preferences on items. In this paper, we propose a multi-linear interactive MF algorithm (MLIMF) to model the interactions between the users and each event associated with their final decisions. Our model considers not only the user-item rating information but also the pairwise interactions based on some empirically supported factors. In addition, we compared the proposed model with three typical other methods: user-based collaborative filtering (UCF), item-based collaborative filtering (ICF) and regularized MF (RMF). Experimental results on two real-world datasets, *MovieLens* 1M and *MovieLens* 100k, show that our method performs much better than other three methods in the accuracy of recommendation. This work may shed some light on the in-depth understanding of modeling user online behaviors and the consequent decisions.

1. Introduction

In recent years, the unprecedented proliferation of information has extremely changed our lifestyles. People all around the world are connected closely because of the daily basis millions of micro-blog posts, tweets and status updates of the social network. The popular online consumption is becoming an essential part of people's daily life, with the result that millions of e-commercial orders are generated per day. However, people are suffering from a serious and widely known problem: how to acquire quality recommendations from the numerous web service providers? Since the early work [1] was published in 1990s, personalized *recommender systems* (RS) [2,3] has been a thriving subfield of data mining to tackle this concern.

In general, RS, serving as a special category of knowledge-based systems, attempts to automatically measure the relevance of user-user or item-item pairs, then delivers items to fit user's tastes via two basic strategies: *Content Based* (CB) [4] and *Collaborative Filtering* (CF) [5]. CB profiles items and users by extracting characteristic units from their content (e.g. demographic data, product information/description), and then identifies the matching-degree by comparing the corresponding profiles. However, due to the high cost to collect the necessary information about items and the lack of motivated users to share their personal data, CB fails to be the most popular recommendation approach. In contrast with CB, CF generates recommendations according to the structure of *virtual community* [6]. The virtual community is based on the underlying assumption that a group of people sharing similar characteristics in the past would also agree on their tastes in future. In addition, CF requires no domain knowledge and offers an alternative approach to reveal the latent patterns that are difficult to be captured by CB methods.

According to pioneering research, CF mainly contains two families: the *Neighborhood Based Models* (NBMs) [7,8] and the *Latent Factor Models* (LFMs) [9,10]. NBMs namely outline the act of working together with neighbors. Here the term "neighbor" does not only point to users, but also items, who share many characteristics in essence. Noteworthiness, *user-* and *item-based* CF [7,8] are two typical strategies to implement NBMs by measuring the likelihood of neighborhood between users or items with pre-defined similarity function. NBMs make predictions based on the known ratings involved with the active users'/items' neighbors. Comparatively, LFMs identify a couple of entities with the same dimensional feature vector inferred from the existing ratings, and straightly express the preference power with the dot-product of the corresponding feature vector pairs. On the basis of previous works, LFMs offer another idea to express various aspects or patterns of data, usually along with high accuracy and scalability.

As the most representative technique of LFM, Matrix Factorization (MF) results in numerous variants validated against the real data sets because of its high accuracy, scalability and

* Corresponding author.
E-mail address: zhangzike@gmail.com (Z.-K. Zhang).

http://dx.doi.org/10.1016/j.knosys.2015.05.016
0950-7051/

expressive ability to capture various context factors (e.g. emotion, location, time). The earliest work of employing MF to implement CF was proposed by Sarwar et al., who conducted a case study on the application of dimension reduction in CF with Singular Value Decomposition (SVD) method [11]. Recently, Hofmann [9] reported on applying Latent Semantic Model to implementing LFM. At the beginning of Netflix Prize Competition [12] in 2006, Brandyn Webb detailed how the Regularized Matrix Factorization (RMF) [13] helped his team rank in the third place under the pseudonym Simon Funk. Subsequently, several works [10,14–17] showed that RMF has played a significant role in the solution that won the Netflix Prize (NP). The attractive characteristics (e.g. methodological simplicity, easy incorporation of additional information, high accuracy) of RMF inspire many researchers to mine its potential from different aspects, such as [10,14,15,18–22] and so on.

As the aforementioned principles of LFMs, the standard RMF can be easily used to discover the latent relationship hidden in the interactions between two entities. In real life, people could take a number of factors into account before making a decision. However, it is difficult for RMF to integrate the interactions between users and the factors beyond items themselves. Though this challenge can be addressed by the Tensor Factorization (TF) [23], the model complexity will grow exponentially with the number of contextual factors. Recently, Koren [14] claimed a methodology to incorporate the RMF with neighborhood information. In addition, Koren [15] proposed a novel work on addressing temporal changes in user behaviors with matrix factorization models. Baltrunas et al. [24] presented the context-aware matrix factorization, which models the interaction of the contextual factors with items. Ma et al. [25] extended the RMF by integrating the social regularization terms under the assumption that two users tend to have similar feature vectors if they are closely connected in social networks.

In this paper, we present a novel approach, namely the Multi-Linear Interactive Matrix Factorization (MLIMF), to model the interactions between users and the factors (e.g. emotions, locations, the time when the rating is given, movie genres, movie directors), which may have significant influence on the user's decision process. Generally, web systems could log multiple information correlated with customer's rating over a specific item. In our model, besides the interaction between the user-item pair, we represent the relationship between a specific user-factor pair in a same latent space. Then, through extending the standard RMF, we linearly integrate the total pairwise interactions together as the components of the customer's final rating decision to construct MLIMF. To clear the principles and application scenarios of MLIMF, we conduct two examples in two real datasets of Movielens with different size. Experimental results prove that, comparing with the standard RMF and other baseline algorithms, MLIMF could obtain better accuracy with linear complexity. The main contributions of this work include:

– In addition to the rating matrix, online users' rating decision could be probably influenced by some other factors. We propose that user could have a special interaction with each factor, and such pairwise relationship could be represented in a same latent space.
– MLIMF, maintaining the principles and expressive scalability of MF, presents an alternative approach to take into account extra information based on the RMF. In fact, the key idea of MLIMF can be incorporated into other invariants of RMF.
– Two application scenarios of MLIMF are given. First, we show that the extracted different feasible features from the training sample serving as the accessorial information which could have significant influence on the user's rating action. Then, we describe how to model the user's temporal dynamic preferences by integrating the time factor into MLIMF.

The remainder of this paper is organized as follows. Section 2 describes the preliminaries. In Section 3 we detail our proposed recommendation model. Section 4 gives two application scenarios. Experimental results are given in Section 5. Finally, Section 6 summarizes this work and outlooks future work.

2. Preliminaries

The CF problem can be simply defined as generating personalized recommendations for a given user by seeking for a group of people or items with similar features from a finite data sample. In the area of CF, the user preferences over involved items are quantized into the user-item rating matrix $R_{|U|\times|I|}$, where $|U|$ and $|I|$ respectively denote the size of the given user set U and item set I. Each entry r at position $(u,\ i)$ of $R \in \mathbb{R}^{|U|\times|I|}$, denoted by r_{ui}, presents the user u's preference on item i, usually with high value expressing the strong relationship between the user-item pair. Typically, in terms of system's received specific feedback, r_{ui} can be binary ($r_{ui} \in \{0,1\}$), integers from a given range (e.g. $r_{ui} \in \{1,2,3,4,5\}$), or a continuous numerical interval (e.g. $r_{ui} \in [-5,5]$). In practice, matrix R is usually very sparse and we can only observe a limited set, $X = \{(u_1,i_1,r_1),(u_2,i_2,r_2),\ldots,(u_t,i_t,r_t)\}$, normally $|X| \ll |U|\times|I|$. Thereby, CF based recommendation tasks can be regarded as missing data estimation through the known user-item rating pairs.

2.1. Regularized matrix factorization

Among the huge amount of solutions to CF problem, RMF has been demonstrated to be superior to classic NBMs on the grand NP competition. Furthermore, numerous RMF variants are proposed to discuss the probable applications of MF and show their high efficiency and accuracy on several real rating datasets as well. Different from traditional NBMs, the goal of RMF is to approximate R by constructing two low-rank matrices. The basic principle of RMF is to map a pair of entities into the same low-dimension feature space. Thus each entity could be represented as a low-dimension feature vector. Taking the rating prediction problem as an example, let f denote the dimension of the feature space. $P \in \mathbb{R}^{|U|\times f}$ denotes the user feature matrix where each row p_u corresponds to a particular user u and $Q \in \mathbb{R}^{|I|\times f}$ represents the item feature matrix where each row q_i corresponds to a particular item i (usually $f \ll \min(|U|,|I|)$). Then the rating approximation of user u on item i could be transformed as calculation of the dot-product of corresponding user-item feature vector pair,

$$\widehat{r}_{ui} = p_u q_i^T, \tag{1}$$

where $\widehat{r}_{ui}$ is the estimate of r_{ui}. Usually, the values of parameters in P and Q can be learned from the training samples by applying the stochastic gradient decent (SGD) to optimize the objective function $J(P,\ Q)$:

$$\min_{U,I} J(P,\ Q) = \frac{1}{2}\sum_{u\in U}\sum_{i\in I} 1(u,i)\left(r_{ui} - p_u q_i^T\right)^2 + \frac{\lambda}{2}(\|p_u\|_F^2 + \|q_i\|_F^2), \tag{2}$$

where $||\cdot||_F$ represents the Frobenius norm. $1(u,i)$ is an indicator function and $1(u,i) = 1$ if user u rates item i, otherwise $1(u,i) = 0$. The second term of Eq. (2) serves as the regularizing bulk for avoiding overfitting, meaning that the trained model has bad generalization for the new coming case. According to [19], λ is the weight parameter for the regularized term. As Eq. (2) shows, J is a quadratic function with local minimum. Under the principles of SGD solver, the involved parameters of feature matrices, P and Q, can be updated by moving in the opposite direction of the gradient for each training example. The optimized result could always be found after looping through all