Knowledge-Based Systems 000 (2016) 1–11

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Aspect-based latent factor model by integrating ratings and reviews for recommender system

Lin Qiu, Sheng Gao*, Wenlong Cheng, Jun Guo

Beijing University of Posts and Telecommunications, Beijing, China

ARTICLE INFO

Article history:
Received 19 December 2015
Revised 19 July 2016
Accepted 24 July 2016
Available online xxx

Keywords:
Latent factor model
Review aspect
User interest
Item attributes

ABSTRACT

Recommender system has been recognized as a superior way for solving personal information overload problem. Rating, as an evaluation criteria revealing how much a customer likes a product, has been a foundation of recommender systems for a long period based on the popular latent factor models. However, review texts as the valuable user generated content have been neglected all the time. Recently, models integrating ratings and review texts as training sources have attracted a lot of attention, which may model review texts by topic model or its variants and then link latent factor vectors to topic distribution of review texts. For that, the integrated models need complicated optimization algorithms to fuse the heterogeneous sources, that may cause greater errors.

In this work, we aim to propose a novel model, called Aspect-based Latent Factor Model (ALFM) to integrate ratings and review texts via latent factor model, in which by integrating rating matrix, user-review matrix and item-attribute matrix, the user latent factors and item latent factors with word latent factors can be derived. Our proposed model aggregates all review texts of the same user on the respective items and builds a user-review matrix by word frequencies. Similarly, an item's review is considered as all review texts of the same item collected from respective users. According to different information abstracted from review texts, we introduce two different kinds of item-attribute matrix to integrate the item-word frequencies and polarity scores of corresponding words. Experimental results on real-world data sets from amazon.com illustrate that our model can not only perform better than traditional models and art-of-state models on rating prediction task, but also accomplish cross-domain task through transferring word embedding.

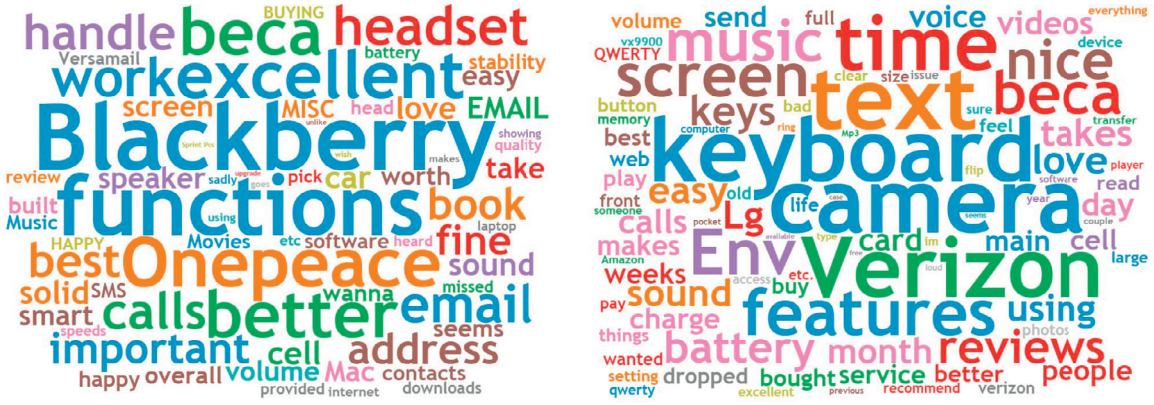
1. Introduction

With the rapid development of e-commerce, users always are trapped by endless choices. Recommender system, which can understand personal preference of users, provides much help in that condition. Most works on the traditional recommendation task make predictions by utilizing ratings, which is an evaluation criteria revealing how much a customer likes a product as a foundation of recommender systems. With the development of e-commerce web sites, ratings are nowadays always accompanied with review texts, which include more details about the purchased items. A plain-text review and a numeric score are supposed to be issued at the same time, which means ratings and review texts are the two different ways of showing how much a user likes an item. The nontrivial differences are that a review text expresses the user's specific feelings on specific aspects of the item but a rating only represents a general opinion on the item. For example, a user is quite likely to express his satisfaction on plot as well as director with a movie in his review text, in condition that a top-level score has been published. That means, review texts are likely to be used for explaining why the user liked or disliked an item, thus increasing ratings prediction accuracy.

Basically, review text is a kind of short text with strong purpose, which can be written for expressing users' feelings on items in the form of sentences or words about specific items. Since information from review texts are more abundant than from ratings, there are a lot of works studying on improving recommender system performance by using ratings and review texts as train data at the same time. Most previous works on unified model combining ratings and review texts concentrate on explaining latent dimensions in users latent factor vectors or items latent factor vectors as topics discovered from review text [2,13,14]. However, users' feeling is more complicated than simple topics. A straightforward approach to capturing aspect and the corresponding sentiment would

* Corresponding author.
E-mail addresses: linqiu@bupt.edu.cn (L. Qiu), gaosheng@bupt.edu.cn (S. Gao), chengwenlong@bupt.edu.cn (W. Cheng), guojun@bupt.edu.cn (J. Guo).

http://dx.doi.org/10.1016/j.knosys.2016.07.033
0950-7051/



(a) Words Cloud of a user's review text. (b) Words Cloud of an item's review text.

Fig. 1. Words clouds. Words clouds graphs are generated by using an online website: http://www.wordclouds.com/. A word cloud describes an article with a graph. In a words cloud, the more frequently the word appears in the text, the bigger the word is displayed in the graphic. Word clouds for a user and an item are obviously different. The user focus on 'blackberry' and 'function', but the item is good at 'keyboard' and 'camera'.

be to apply statistic method or topic model on review texts, by combining latent factor model on ratings, respectively [5,16,27]. However, these approaches just jointly learn the aspects from ratings and reviews texts, failing to exploit meaningful dependencies between latent factors for predicting ratings and latent variables for modeling opinion implied in users' comments on different aspects on items. Here the *aspects* have two meanings: one is for the user-specific aspects on an item based on her individual review text, indicating her specific concerns on some attributes of the item and explaining why she gives a numeric rating score for this item, the other is for the item-specific aspects on itself based on all the respective review texts from corresponding users, indicating the highlighted perspectives on its attributes commented by the users.

For example, we randomly choose a user and an item from one of Amazon data sets as examples. For a user, we count the words frequency in all his reviews, then graphed them, just as shown in Fig. 1(a). The bigger a word are, the more frequently it occurs in the user's reviews. This result suggests that not all aspects or topics share the same weight when they are generated. Users tend to express opinions on 'important' topics firstly. For an item, we do the same things. Fig. 1(b) shows the graph generated by an item's review text. Different from reviews of a user, a 'bigger' word in Fig. 1(b) means that there are more people talking about the topic. That is, it is an advantage or disadvantage of the item.

In this work, we propose an Aspect-based Latent Factor Model (ALFM) to capture the two kinds of aspect information by modeling the dependencies between overall ratings and review texts via latent factors. Unlike traditional latent factor model on rating, ALFM factories three special matrices instead of the single rating matrix. The first one is rating matrix, where each element represents preference of an item by a user, where each element is product of a user latent factor and an item latent factor. The second one is user-review matrix, where each element is word frequency in review texts of users. Here the each element is product of a user latent factor and a word-representation latent factor. The third one is item-attribute matrix which may have two types: item-property matrix and item-quality matrix. For convenience, we use ALFM-P representing the model employing item-property matrix and ALFM-Q representing the model employing item-quality matrix. Although item-property matrix is made of word frequency and item-quality matrix is made of word polarity score, they both are factorized by item latent factors and word latent factors. For the two models, user latent factor vectors capture importance of aspects. The values of latent dimensions in a vector implies whether an aspect is a critical aspect. If an aspect is a critical aspect, its properties can determine whether a user equally likes the item even if whether a user wants to purchase the item. User latent factors are used for predicting ratings and calculating word frequency. With different item-attribute matrix employed, item latent factor vectors have different meanings. With item-property matrix, it captures items' properties. And with item-quality matrix, it captures users' satisfaction on aspects. For the former, a high value means a impressed feature of an item. For the latter, higher the value of latent dimension in a vector is, more likely a user gives a high score on the corresponding aspect. Latent ratings on aspects are dependent on items' quality instead of users' mood. Since it is demonstrated that the polarity of ratings and polarity of review texts are coherent [8], we can readily use it to improve rating prediction.

The contributions of our work are as follows:

1. We propose an aspect-based model on ratings and review texts for recommender system. We demonstrate that the joint models based on users' interested aspect information and items' property or items' quality information on aspects is more expressive than collaborative filtering models integrating with topic model or statistical model. That is, our models obtain lower mean-squared error than other unified models on each dataset. In addition, we give a simple comparison of different results with varied weights of auxiliary matrix. We also compare results of ALFM-P with JMARS on 'cold-start' users and items.
2. We transform features of users, items, words into the same vector space. Through the model, we obtain not only embedding of users and items, but also latent representations of words. Moreover, two kinds of aspect information can be learned based on our proposed model to explain clearly why users show their preferences on the items.
3. Experiment results on real data sets illustrates our models can not only perform better than traditional models and art-of-state models on rating prediction task, but also accomplish cross-domain task through transferring word embedding. The improvement on 'cold-start' users and items are significant, too.

This paper is organized as follows. Section 2 gives briefly reviews of some related works. Section 3 first formulates recommendation problem with review texts, then introduces ALFM in detail. Section 4 introduces the parameter estimation via gra-

Please cite this article as: L. Qiu et al., Aspect-based latent factor model by integrating ratings and reviews for recommender system, Knowledge-Based Systems (2016), http://dx.doi.org/10.1016/j.knosys.2016.07.033