Contents lists available at ScienceDirect

Decision Support Systems

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

Scalable learning of probabilistic latent models for collaborative filtering

Helge Langseth [a,*], Thomas D. Nielsen [b]

[a] Department of Computer and Information Science, The Norwegian University of Science and Technology, Trondheim, Norway
[b] Department of Computer Science, Aalborg University, Aalborg, Denmark

ARTICLE INFO

Article history:
Received 29 April 2014
Received in revised form 6 February 2015
Accepted 19 March 2015
Available online 31 March 2015

Keywords:
Collaborative filtering
Scalable learning
Probabilistic model
Latent variables
Variational Bayes

ABSTRACT

Collaborative filtering has emerged as a popular way of making user recommendations, but with the increasing sizes of the underlying databases scalability is becoming a crucial issue. In this paper we focus on a recently proposed probabilistic collaborative filtering model that explicitly represents all users and items simultaneously in the model. This model class has several desirable properties, including high recommendation accuracy and principled support for group recommendations. Unfortunately, it also suffers from poor scalability. We address this issue by proposing a scalable variational Bayes learning and inference algorithm for these types of models. Empirical results show that the proposed algorithm achieves significantly better accuracy results than other straw-men models evaluated on a collection of well-known data sets. We also demonstrate that the algorithm has a highly favorable behavior in relation to cold-start situations.

1. Introduction

Recommender systems have become a well-established technology to help users cope with vast amounts of information. This is achieved by only presenting to the users the information which is deemed most relevant. Over the last years the diversity of the domains in which recommender systems have been successfully applied has increased significantly and includes movies, books, news, and products in general.

Recommender systems are typically grouped into two categories: *content-based* systems make item recommendations by combining content descriptions of the items in questions with a preference model of the user (e.g. inferred using previously rated items). On the other hand, *collaborative filtering* systems provide recommendations based on the ratings of other users with similar preferences. The two types of systems exhibit different characteristics; collaborative filtering systems typically enjoy a greater flexibility in terms of the types of items that can be recommended whereas content-based systems are usually less susceptible to cold-start problems.

Collaborative filtering systems are often further sub-divided into *model-based* and *memory-based* [4] methods, although combinations of the two have also been proposed [29]. Memory-based systems rely on a distance measure to estimate user similarity, whereas model-based approaches learn a model of the user's preferences, which is subsequently used for making predictions. The earliest model-based approaches used a multinomial mixture model [10] for either grouping the users into user groups or items into item-categories. More recently, uniform models have been proposed that treat users and items equivalently and represent them jointly in the same model. An example of such a model is the probabilistic latent variable model proposed by Langseth and Nielsen [20].

The model described in [20] bears some resemblances with relational probabilistic models [13] in that it explicitly combines all users and items directly in the model [34,33,12]. More specifically, the model is a special type of conditional linear Gaussian model [21], where each item and user is represented by a collection of abstract latent variables encoding intrinsic properties about the user/item in question. The rating assigned to item i by user p is in turn modeled as a linear Gaussian distribution conditioned on the corresponding latent user/item representations. This joint representation of users and items allows the model to take advantage of all the user/item information available when making recommendations. Not only does this result in high-quality recommendations, as documented in [20], but it also supports a well-founded and principled way of making group recommendations [7,25].

In order to learn the probabilistic latent variable models, Langseth and Nielsen [20] proposed an Expectation–Maximization (EM) algorithm tailored to the specific model class. Unfortunately, the algorithm requires the calculation of the full covariance matrix for all the latent variables representing the users and items. Consequently, the algorithm does not scale to larger data sets.

In this paper we address the scalability problem by proposing approximate learning and inference algorithms based on a variational Bayes approach [1,3]. The algorithms employ a (generalized) mean-field approximation of the variational distribution, which ensures that the complexity of the learning algorithm grows *linearly* in the number of data points/ratings. Furthermore, we show that the model fits within the general class of statistical query models that in turn supports an

[*] Corresponding author at: Department of Computer and Information Science, The Norwegian University of Science and Technology, Sem Sælands vei 9, NO-7491 Trondheim, Norway.

E-mail addresses: helgel@idi.ntnu.no (H. Langseth), tdn@cs.aau.dk (T.D. Nielsen).

http://dx.doi.org/10.1016/j.dss.2015.03.006
0167-9236/© 2015 Elsevier B.V. All rights reserved.

efficient use of the MapReduce framework [8,6]; hence the algorithms are easily parallelizable and can exploit distributed architectures. We empirically evaluate the proposed algorithms using several well-known data sets and demonstrate that the algorithm obtains results that are significantly better than what is obtained by a collection of straw-men methods. Finally, we analyze the performance of the method under cold-start conditions [19].

The remainder of the paper is structured as follows: In Section 2 we provide background information and describe the probabilistic latent variable model by Langseth and Nielsen [20]. Section 3 describes the variational Bayes based learning algorithm (with detailed derivations included in the Appendix A) and in Section 4 we present the empirical results. We conclude the paper in Section 5 and outline directions for future research.

2. A latent model for collaborative filtering

2.1. Bayesian network

A Bayesian network (BN) is a probabilistic graphical model that defines a compact representation of a joint probability distribution by exploiting and explicitly encoding conditional independence properties among the variables. The specification of a BN over a collection of variables $\{X_1, \ldots, X_n\}$ consists of two parts: a qualitative part and a quantitative part. The qualitative part corresponds to an acyclic directed graph $G = (\mathcal{V}, \mathcal{E})$, where the nodes $\mathcal{V}$ represent the variables $\{X_1, \ldots, X_n\}$ through a one-to-one mapping, and the edges $\mathcal{E}$ specify the direct dependencies between the variables. For ease of exposition, we shall refer to nodes and variables interchangeably.

We shall describe the relations between the variables in a Bayesian network using graph terminology. Thus, the nodes whose outgoing edges intersect a node/variable X_i are called the parents of X_i, denoted $\boldsymbol{\pi}_{X_i}$, and the nodes to which there exists an edge emanating from X_i are called the children of X_i. If there is a directed path from a node X_i to a node X_j, then X_j is said to be a descendant of X_i. Together the edges in the graph encode the conditional independence assumptions in the Bayesian network. Specifically, a node X_i is conditionally independent of its non-descendants given its parents.

The quantitative part is defined by a collection of conditional probability distributions or density functions s.t. each node is assigned exactly one probability distribution conditioned on its parents. In the remainder of this paper we shall assume that all variables are continuous. In particular, a variable X_i with parents $\boldsymbol{\pi}_{X_i}$ is assumed to follow a conditional linear Gaussian distribution

$$f\left(x_i|\boldsymbol{\pi}_{x_i}\right) = \mathcal{N}\left(\mu_i + \boldsymbol{w}_i^{\mathrm{T}}\boldsymbol{\pi}_{x_i}, \sigma_i\right),$$

i.e., the mean value is given as a weighted linear combination of the values of the parent variables whereas the variance is fixed. The underlying conditional independence assumptions encoded in the BN allow us to calculate the joint probability function using the chain rule:

$$f(x_1, \ldots, x_n) = \prod_{i=1}^{n} f\left(x_i|\boldsymbol{\pi}_{x_i}\right).$$

With linear Gaussian distributions assigned to all the variables it follows that the joint distribution is a multivariate Gaussian distribution. The inverse of the covariance matrix (also called the precision matrix) for this multivariate distribution directly reflects the independencies defined by the BN; the entry defined by a pair of variables is zero if and only if the two variables are conditionally independent given the other variables in the network.

2.2. A latent variable model

The collaborative filtering method proposed in [20] relies on a Bayesian network representation that provides a joint model of all items, users, and their ratings. Before presenting the details of the model, we shall first introduce some notation.

We will denote the matrix of ratings by $\mathbf{R}$, which is of size $\# U \times \# M$. Here $\# U$ is the number of users and $\# M$ is the number of items that are rated. $\mathbf{R}$ is a sparse matrix, meaning that it contains a considerable amount of missing values (more than 99% missing observations is quite common). The observed ratings are either realizations of ordinal variables (discrete variables with ordered states, e.g., "`Dislike`", "`Neutral`", "`Like`") or real numbers. In the following we will consider only continuous ratings encoded by real numbers, and assume that ratings given as ordinal variables have been translated into a numeric scale.

We use p as the index of an arbitrary person using the system, and i is the index of an item that can be rated. Consequently, $\mathbf{R}(p, i)$ is the rating that person p gives item i. Next, we will use $\delta(p, i)$ as an indicator function to show whether or not person p has rated item i. Specifically, $\delta(p, i) = 1$ if the rating exists and $\delta(p, i) = 0$ otherwise. Furthermore, $\mathcal{I}(p)$ is the set of items that person p has rated, i.e., $\mathcal{I}(p) = \cup_{i:\delta(p,i) \neq 0}\{i\}$, and similarly $\mathcal{P}(i) = \cup_{p:\delta(p,i)\neq 0}\{p\}$ is the set of persons that have rated item i. Lowercase letters are used to signify that a random variable is observed, so $\boldsymbol{r}(p, i)$ is the rating that p has given item i (that is, $\delta(p, i) = 1$ in this case). Finally, we let $\boldsymbol{r}$ denote all observed ratings (the part of $\mathbf{R}$ that is not missing).

When doing model-based collaborative filtering from a general perspective we look for a probabilistic model that for any item i and user p defines a probability distribution over $\mathbf{R}(p, i)$ given model parameters $\boldsymbol{\rho}$ and observed ratings $\boldsymbol{r}$. Given such a probability distribution, we can make recommendations based on the expected rating or the median rating for that distribution.

The probabilistic model that is proposed in [20] defines a joint distribution over all ratings by introducing abstract latent variable representations of both the items and the users. Specifically, each item i is represented by the random variables $\mathbf{M}_i$ and each user p is represented by the random variables $\mathbf{U}_p$. In a movie context one may for example interpret the different dimensions of $\boldsymbol{m}_i$ as representing different features of movie i such as to what extend the movie uses a well-known cast and the amount of explicit violence in the movie. Similarly, the dimensions of $\boldsymbol{u}_p$ can be interpreted as corresponding to different user characteristics. Hence, since the variables are continuous, the value $\boldsymbol{u}_{p,j}$ of the jth variable $\mathbf{U}_{p,j}$ can be interpreted as representing to what extent user p has the characteristics modeled by variable j. This also means that rather than assigning a user to a single "user group", the continuous variables $\mathbf{U}_{p,j}$ encode *to what extent* a user belongs to a certain group.[1] A priori we assume that $\mathbf{U}_p \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, for $1 \leq p \leq \# U$, and $\mathbf{M}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, for $1 \leq i \leq \# M$.

The rating assigned to item i by user p is modeled by assuming the existence of a linear mapping from the space describing users and items to the numerical rating scale:

$$\mathbf{R}(p,i)|\left\{\mathbf{M}_i = \boldsymbol{m}_i, \mathbf{U}_p = \boldsymbol{u}_p\right\} = \boldsymbol{v}_p^{\mathrm{T}}\boldsymbol{m}_i + \boldsymbol{w}_i^{\mathrm{T}}\boldsymbol{u}_p + \phi_p + \psi_i + \epsilon. \tag{1}$$

The rating in Eq. (1) is thus determined as an additive combination of user p's preferences $\boldsymbol{v}_p$ for (or attitude towards) the features describing item i and item i's disposition $\boldsymbol{w}_i$ towards the different user groups.[2] The constants ϕ_p and ψ_i in Eq. (1) can be interpreted as representing the average rating of user p and the average rating of item i (after compensating for the user average), respectively. Furthermore, ε represents "sensor noise", i.e., the variation in the ratings the model cannot explain.

[1] See [20] for an empirical investigation into the possible semantics of the abstract latent variable representations.

[2] Note that the relative importance of the movie features and the user group can be encoded in the weight vectors $\boldsymbol{v}_p$ and $\boldsymbol{w}_i$.