Symmetrization and overfitting in probabilistic latent semantic analysis *

Leksin V. A., vleksin@gmail.com

Moscow Institute of Physics and Technology, 141700, 9, Institutskii per., Dolgoprudny, Moscow Region, Russia

Аннотация

An algorithm is proposed for revealing latent user's interests from the observable protocol of users behavior, e.g., site visits. The algorithm combines the ideas of analysis of users' media and probabilistic latent semantic analysis. A quality criterion based on the classification of preliminarily labeled sites is introduced to optimize the algorithm parameters and compare algorithms. The experiments show that the quality has an optimum by the essential parameters of the algorithm, however the attempt of too precise optimization can lead to overfitting.

Key words: Probabilistic latent semantic analysis, collaborative filtering, customer environment analysis, symmetric models, latent profiles, overfitting

1 Introduction

Automatic revealing of users' needs and preferences by the data on their web behavior (purchases, visits, requests, etc.) is an urgent task in many spheres of client-oriented business. Specifically, such problems should be solved in order to personalize advertising in recommender systems, provide segmentation of the client base in marketing research, search for similarly minded people in social networks, etc.

The initial data are the sequence of "user u selected resource r" records. To successfully solve the aforesaid problems, it is required to adequately estimate the similarity of users and resources. Customer environment analysis (CEA) [17, 10, 11] is based on the following principle of consistent similarity measures: "resources (items) are similar if similar users use them and at the same time, users are similar if they use similar sets of resources."

The simplest methods for web usage mining (WUM) [9] and collaborative filtering (CF) [2], including the Pearson correlation method or the cosine similarity method, are based on either just the similar ity of users (user-based CF) or that of items (item- based CF). The need to store all initial data as well as the asymmetric character of the analysis relative to users and recourses limits the applicability of the methods. The latent semantic analysis (LSA) [7, 4, 15] have not these limitations and allows for revealing implicit characteristics of users and items and replace initial data with compressed descriptions, i.e., profiles of users and items. As a rule, different kinds of matrix decomposition are used

*This work has been supported by RFBR, project nos. 07-01-12076-ofi and 08-07-00422.

for this purpose. Probabilistic models (LSA or pLSA) [8] have more profound statistical substantiation and allow for interpreting the components of profiles as probabilities of preferences. To reveal implicit preferences from the initial data, the EM-algorithm is often used. The methods for collaborative filtering are discussed in more detail in Chapter 1.

This paper proposes an approach that combines the principle of consistent similarity from customer environment analysis and estimation of latent profile-based probabilistic latent analysis. It results in a symmetrized two-stage variant of the EM-algorithm with two enclosed iteration loops in contrast to the standard pLSA algorithm wherein latent variables are calculated in one iteration loop. In section 2 the method is considered in detail and compared to LSA and pLSA algorithms.

Restored profiles are easy to compare, which makes possible applying the simple k nearest neighbors (kNN) classifier and introducing objective quality criteria to compare different methods of collaborative filtering.

In section 3, the experimental results and comparative analysis of three methods of collaborative filtering (pLSA, symmetric pLSA, and the correlation method based on Fisher's exact test) will be presented using real search engine data and the data on articles purchased in a large furniture company. In the experiments the dependence of classification quality on the length of latent profiles and iteration number both in the inner and outer loop of the algorithm is studied. The quality proved to have an optimum by these parameters; however excessive optimization is redundant and can lead to overfitting.

This article is an extended variant of [11].

2 Objectives and Methods of Collaborative Filtering

Collaborative filtering as a research trend started at the end of the 1980s when numerous companies faced the problem of effective usage of a vast amount of "raw"data on users' behavior to solve a number of business problems such as service personalization and direct marketing [13, 12]. Collaborative filtering is based on the assumption that similar users have similar preferences when choosing items. In other words, by finding users that are similar to the active user and by examining their preferences, the recommender system can predict the active user's preferences for certain items and provide a ranked list of items which active user will most probably like. The methods of collaborative filtering do not analyze the content of items; therefore, they are applicable to a wide range of applied areas. Collaborative filtering can detect relationships between items that have no content similarities but are linked implicitly through the groups of users (collaborations) accessing them. These groups (communities) are formed around a specific user profile. The proper methods are considered in the given paper.

2.1 Collaborative Filtering Objective

Let U be the set of users, R be the set of items (resources), Y be the space of descriptions of transactions (either preference facts or users' estimates of items). The initial data are presented as the transaction protocol database of users' preferences, i.e., the sequence of l triples $D = (u_i, r_i, y_i)_{i=1}^{l} \subset U \times R \times Y$.

Commonly, the transaction protocol is aggregated in the cross-tabulation matrix $F = \|f_{ur}\|_{U \times R}$, where $f_{ur} = \text{aggr}\big\{(u_i, r_i, y_i) \in D \;\big|\; u_i = u,\ r_i = r\big\}$, aggr is the aggregation

function whose form depends on the subject area and a particular objective.

The main objectives of collaborative filtering are forecasting of unfilled cells f_{ur}, estimation of similarity functions $K(u,u')$, $K(r,r')$, and $K(u,r)$ between users and items, and revealing of interpretable latent characteristics (profiles) of users and resources.

Information about the transaction protocol can be accumulated in the form of implicit ratings or explicit ratings.

The implicit rating is obtained by monitoring user actions. For example, if U is the set of Internet users, R is the set of resources (sites, documents, news items, etc.), then the protocol of users' visits is aggregated in the cross-tabulation matrix $F = \|f_{ur}\|$ where f_{ur} is the number of visits of the resource r by the user u. The most typical objectives are as follows: 1) predict the active user's preferences for certain items and 2) provide a ranked list of items which active user u will most probably like.

In the case of explicit ratings, users rate the chosen items. Thus, if U are users of an online-store, R are goods (books, video, music, etc.), then a rating made by the user u of the good r can be the value of f_{ur}. Commonly, user ratings have discrete values in the rating scale. Personalization problems are set in a similar way in this case.

The collaborative filtering algorithm is usually divided into two large classes - memory-based and model-based.

2.2 Memory-Based Algorithms

Memory-based algorithms [13, 1, 6] are based on storage of the whole cross-tabulation matrix F and direct search for similar users (rows) and items (columns) in it. The value of the unknown rating $\hat{f}_{ur}$ for user u and resource r is estimated by the set of ratings given to the resource by other users whose preference is the most similar to those of the user u:

$$\hat{f}_{ur} = \operatorname*{aggr}_{u' \in U_\alpha(u)} f_{u'r},$$

where $U_\alpha(u) = \{u' \in U \mid K(u',u) > \alpha\}$ is the set of users similar to u; the function of user similarity $K(u,u')$ takes larger values the closer the u and u' preferences are; the α parameter specifies the threshold value for similarity; aggr is an aggregating function, e.g., the Nadaraj a-Watson formula of nonparametric (kernel) smoothening with the kernel functions $K(u,u')$:

$$\hat{f}_{ur} = \bar{f}_u + \frac{\sum\limits_{u' \in U_\alpha(u)} K(u,u')(f_{u'r} - \bar{f}_{u'})}{\sum\limits_{u' \in U_\alpha(u)} K(u,u')},$$

where $\bar{f}_u = \frac{1}{|R(u)|} \sum\limits_{r \in R(u)} f_{ur}$ is the average u user's rating, $R(u)$ is the set of items chosen by the user u. The closer the preferences of users u and u', the greater the contribution of rating $f_{u'r}$ into $\hat{f}_{ur}$ to be forecasted.

In collaborative recommender systems, different approaches to estimating the similarity between users $K(u,u')$ are used. Let $R(u,u')$ be the set of items chosen by both users u and u'. In the correlation approach [13, 6], the similarity of users u and u' is

estimated by the Pearson correlation coefficient:

$$K(u,u') = \frac{\sum\limits_{r \in R(u,u')} (f_{ur} - \bar{f}_u)(f_{u'r} - \bar{f}_{u'})}{\sqrt{\sum\limits_{r \in R(u,u')} (f_{ur} - \bar{f}_u)^2 \sum\limits_{r \in R(u,u')} (f_{u'r} - \bar{f}_{u'})^2}}.$$

With the approach using the linear similarity approach [6, 1], users u and u' are presented as vectors of m-dimensional space, $m = |R(u,u')|$, and the similarity is estimated as the cosine of the angle between these vectors:

$$K(u,u') = \frac{\sum\limits_{r \in R(u,u')} f_{ur} f_{u'r}}{\sqrt{\sum\limits_{r \in R(u,u')} f_{ur}^2 \sum\limits_{r \in R(u,u')} f_{u'r}^2}}.$$

One more similarity function applied to the cross-tabulation matrices is based on Fisher's exact test [17]. We consider the function relative to resources rather than users since in this proper form it will be used in the experimental part of our work. Generally, most similarity functions used in collaborative filtering can be defined for both users and resources. We will estimate the similarity of resources r and r' by testing the statistical hypothesis that users choose at least one of two resources r and r' independently. Let $U(r)$ and $U(r')$ be the sets of users who prefer either just the resource r or the resource r', respectively, and let $U(r,r')$ be the set of users choosing both resources. If the set of $U(r,r')$ is so large that the probability of joint choice of both resources

$$P(r,r') = C_{|U(r)|}^{|U(r,r')|} C_{|U|-|U(r)|}^{|U(r')|-|U(r,r')|} \;/\; C_{|U|}^{|U(r')|}$$

is less than the specified level of significance α, then one can assume that the data observed contradict the independence hypothesis. Consequently, there is a regular relationship between visits of this pair of resources. The less $P(r,r')$, the more similar the resources. The similarity function is defined as a monotonically decreasing function of probability, e.g., $K(r,r') = -\log P(r,r')$.

2.3 Model-Based Algorithms

In contrast to memory-based methods, model-based algorithms store in memory neither the initial protocol nor the cross-tabulation matrix F. Instead, a vector description (profile) is formed for each user and each item. The similarity functions of users and items are realized by direct comparison of these profiles. In some models the profile components have a content interpretation. In particular, they can correspond to the types and topics of items, users' interests, or user's social and demographic characteristics. In some applications profiles may not possess any interpretation.

Latent semantic analysis (LSA) is based on matrix factorizations [7, 15, 4]. Let us assume that each user is interested in a set of topics from a set of all possible topics T. Commonly, the number of topics $|T|$ is considerably smaller than the number of users $|U|$, the number of items $|R|$ and protocol length l. Let us denote the importance degree of a topic t for a user $u \in U$ through p_{tu}. Then the vector $(p_{tu})_{t \in T}$ is a latent profile of user u, and $P = (p_{tu})_{|T| \times |U|}$ is the matrix of all profiles of all users. Similarly, we denote

through q_{tr} the ability of the item $r \in R$ to satisfy the interest in topic t. Then the vector $(q_{tr})_{t\in T}$ is a latent profile of the item r, and $Q = (q_{tr})_{|T|\times|R|}$ is the matrix of profiles of all items. We also introduce λ_t, characterizing the degree of importance of the topic $t \in T$ independent of items and users. Then it is quite natural to describe values in the cross-tabulation matrix by the model $\hat{f}_{ur} = \sum\limits_{t\in T} \lambda_t p_{tu} q_{tr}$, or, in the matrix form, $\hat{F} = P\Lambda Q^{\intercal}$. To find matrices P and Q, the least squares method is applied:

$$\sum_{u\in U}\sum_{r\in R}\left(\hat{f}_{ur} - f_{ur}\right)^2 = \left\|F - P\Lambda Q^{\intercal}\right\|^2 \to \min_{P,\Lambda,Q}. \tag{1}$$

The given problem is solved by means of singular value decomposition, or, which is the same, the principal components analysis. In practice, explicit solution of the problem of eigenvalue is time consuming. There are fast iteration methods estimating $P\Lambda$ and Q without direct calculation of the matrix spectrum and at the same time determining the number of topics T [16].

In most typical applications, matrix F is not filled completely and, moreover, is strongly sparse. Then, the summing up in (1) is only made by pairs $(u, r) \in U \times R$, for which the values of f_{ur} exist. In these cases the sparse methods of principal components analysis are used [3].

The singular factorization has a drawback that some components of profiles p_{tu} and q_{tr} turn out to be negative, which impedes their interpretation. Nonnegative matrix factorizations (NNMF) [5] guarantee that $p_{tu} \geqslant 0$ and $q_{tr} \geqslant 0$. In this case the elements of profiles p_{tu} could be interpreted as the conditional probabilities $p(u|t)$, if normalization $\sum\limits_{u\in U} p(u|t) = 1$ was fulfilled.

Probabilistic latent semantic analysis is based on explicit probabilistic interpretation of the profiles [14]. Owing to effective numerical methods, these models are widely used [6, 2, 1].

The latent profile of a user $u \in U$ is the vector of (unknown) conditional probabilities $p_{tu} = p(t|u)$ that the given user u is interested in the topic $t \in T$. The profile should meet the normalization condition $\sum\limits_{t\in T} p_{tu} = 1$.

The latent profile of an item $r \in R$ is the vector of (unknown) conditional probabilities $q_{tr} = q(t|r)$ that the given item r corresponds to the topic $t \in T$. Similarly, $\sum\limits_{t\in T} q_{tr} = 1$.

Here and below all probabilities concerning users are denoted by p, and the probabilities concerning items are denoted by q.

We consider the probabilistic model of item r preference by a user u:

$$p(u, r) = \sum_{t\in T} p(t)p(u|t)q(r|t), \tag{2}$$

where $p(t)$ is a priori probability characterizing the "popularity"of the topic $t \in T$, $p(u|t)$ is a posteriori distribution of users by each topic t, $q(r|t)$ is the a posterior distribution of items by each topic t.

Then the principle of the maximum likelihood is used to find unknown parameters $p(t)$, $p(u|t)$, and $q(r|t)$ in the given model by the cross-tabulation matrix $F = \|f_{ur}\|_{U\times R}$ observed where f_{ur} is the number of times that the user u has chosen the item r:

$$L = \ln \prod_{u\in U}\prod_{r\in R} p(u, r)^{f_{ur}} = \sum_{u\in U}\sum_{r\in R} f_{ur} \ln p(u, r) \to \max_{p(t),\, p(u|t),\, q(r|t)}. \tag{3}$$

To estimate the maximum likelihood the Expectation-Maximization (EM) algorithm is used. The idea is as follows. To estimate unknown parameters $p(t)$, $p(u|t)$, and $q(r|t)$, auxiliary latent variables $p(t|u,r)$ are introduced that can be interpreted as the probability that the user u, choosing the item r, was interested in the topic t. Latent variables can be calculated easily if the parameters $p(t)$, $p(u|t)$, and $q(r|t)$ are known. On the other hand, the solution of the problem of maximum likelihood is strongly simplified if the latent parameters are known.

The EM algorithm consists of two steps repeated iteratively.

At the E-step (Expectation), the expected values of latent variables $p(t|u,r)$ are calculated by the Bayes formula based on the current values of unknown parameters:

$$p(t|u,r) = \frac{p(t)p(u|t)q(r|t)}{\sum\limits_{t' \in T} p(t')p(u|t')q(r|t')}, \quad u \in U,\ r \in R,\ t \in T.$$

At the M-step (Maximization), the problem of maximum likelihood is to be solved and the following approximation of unknown parameters $p(t)$, $p(u|t)$, and $q(r|t)$ is found. The problem can be solved analytically using latent variables found at the E-step. The solution is written in the following way:

$$p(t) = \frac{\sum\limits_{u \in U} \sum\limits_{r \in R} f_{ur} p(t|u,r)}{\sum\limits_{u \in U} \sum\limits_{r \in R} f_{ur}}, \quad t \in T;$$

$$q(r|t) = \frac{\sum\limits_{u \in U} f_{ur} p(t|u,r)}{\sum\limits_{u \in U} \sum\limits_{r' \in R} f_{ur'} p(t|u,r')}, \quad r \in R,\ t \in T;$$

$$p(u|t) = \frac{\sum\limits_{r \in R} f_{ur} p(t|u,r)}{\sum\limits_{u' \in U} \sum\limits_{r \in R} f_{u'r} p(t|u',r)}, \quad u \in U,\ t \in T.$$

Further, one needs to return to the E-step at new values of parameters $p(t)$, $p(u|t)$, and $q(r|t)$. Iterations continue until stabilization of the values of parameters and/or likelihood. Initial approximations for $p(t)$, $p(u|t)$, and $q(r|t)$ are initialized by random or uniform distributions.

The found parameters $p(u|t)$ and $q(r|t)$ are conditional distributions of users and items relative to each topic t. However, the desired profiles of users and items should have the form of conditional distributions of topics $p(t|u)$ and $q(t|r)$. To calculate them the Bayes formula is used:

$$p_{tu} = p(t|u) = \frac{p(t)p(u|t)}{\sum\limits_{t' \in T} p(t')p(u|t')}, \quad u \in U,\ t \in T;$$

$$q_{tr} = q(t|r) = \frac{p(t)q(r|t)}{\sum\limits_{t' \in T} p(t')q(r|t')}, \quad r \in R,\ t \in T.$$

3 Symmetric Probabilistic Latent Semantic Model

We propose a symmetric model of probabilistic latent semantic analysis wherein the profiles of users and items are specified alternately. The experiments show that in this

case both the accuracy of profiles and the rate of convergence increase.

Let $F = \|f_{ur}\|_{U\times R}$ be a cross-tabulation matrix where f_{ur} is the number of times when the user $u \in U$ has chosen the item $r \in R$.

Instead of probabilistic model (2), we will write another expression formally equivalent to it for the probability of the choice of the item r by the user u:

$$p(u,r) = \sum_{t\in T} p(u)p(t|u)q(r|t,u), \tag{4}$$

where $p(u) = p_u$ and $q(r) = q_r$ are prior probabilities of the occurrence of a user u and an item r respectively, $p(t|u) = p_{tu}$ is the probability that the user u is interested in the topic t, and $q(r|t,u) = q(r|t)$ is the posteriori probability that the item r will be chosen under the condition that the choice is due to interest in topic t. The hypothesis that the posteriori probability $q(r|t,u)$ does not depend on the user u is a necessary simplifying assumption in the given model. A priori probabilities and profiles should meet the conditions of normalization $\sum\limits_{u\in U} p_u = 1$ и $\sum\limits_{t\in T} p_{tu} = 1$ for all $u \in U$.

We express the posteriori probability $q(r|t)$ through a priori probabilities $q(r) = q_r$ and item profiles $q(t|r) = q_{tr}$ by using the Bayes formula:

$$q(r|t) = \frac{q(t|r)q(r)}{\sum\limits_{r'\in R} q(t|r')q(r')} = \frac{q_{tr}q_r}{\sum\limits_{r'\in R} q_{tr'}q_{r'}}.$$

We substitute the expression into the formula for $p(u,r)$:

$$p(u,r) = \sum_{t\in T} p_u p_{tu} \frac{q_r q_{tr}}{\sum\limits_{r'\in R} q_{tr'}q_{r'}}.$$

A priori probabilities p_u and q_r are easy to estimate empirically as the share of transactions wherein, respectively, the choice was made by the user u or the item r was chosen:

$$p_u = \frac{1}{l}\sum_{r\in R} f_{ur}; \qquad q_r = \frac{1}{l}\sum_{u\in U} f_{ur}; \qquad l = \sum_{u\in U}\sum_{r\in R} f_{ur}.$$

Therefore, the probability $p(u,r)$ is expressed through known priori probabilities p_u, q_r, and unknown profiles of users p_{tu} and items q_{tr}. Note that $p(u,r)$ linearly depends on the profiles of users p_{tu} and in a rather complex way on the profiles of items q_{tr}. Therefore, we will assume that the profiles of items q_{tr} have been known and fixed already. To find the profiles of users $P = (p_{tu})_{|T|\times|U|}$, we shall maximize the likelihood at $|U|$ equality restrictions

$$L(P) = \sum_{u\in U}\sum_{r\in R} f_{ur} \ln p(u,r) \to \max_{\{p_{tu}\}};$$
$$\sum_{t\in T} p_{tu} = 1, \quad u \in U.$$

There are also inequality restrictions $p_{tu} \geqslant 0$. However, they will not be considered and later we will prove that they are fulfilled with a guarantee.

Let us write the Lagrangian of the optimization problem:

$$L(P,\lambda) = \sum_{u\in U}\sum_{r\in R} f_{ur}\ln\Big(p_u\sum_{t\in T} p_{tu}q(r|t)\Big) - \sum_{u\in U}\lambda_u\Big(\sum_{t\in T} p_{tu} - 1\Big),$$

where $\lambda = (\lambda_u)_{u\in U}$ is the vector of dual variables. We differentiate the Lagrangian with respect to p_{tu} and set the derivative equal to zero:

$$\frac{\partial L}{\partial p_{tu}} = \sum_{r\in R} f_{ur}\frac{1}{p_{tu}}\frac{p_{tu}q(r|t)}{\sum\limits_{t'\in T} p_{t'u}q(r|t')} - \lambda_u = 0, \quad t\in T,\ u\in U. \tag{5}$$

We introduce auxiliary *latent variables* $H_{tr}(u)$:

$$H_{tr}(u) = \frac{p_{tu}q(r|t)}{\sum\limits_{t'\in T} p_{t'u}q(r|t')}, \quad t\in T,\ r\in R,\ u\in U.$$

Note that according to the Bayes formula, $H_{tr}(u) = H(t|r,u)$ is a posteriori probability of the topic t for the given pair (u,r). In other words, it is the probability that interest in topic t caused the choice of the item r by the user u. Evidently, for any pair $(u,r)\in U\times R$, the normalization condition $\sum\limits_{t\in T} H_{tr}(u) = 1$ is fulfilled.

Let us assume that the latent variables $H_{tr}(u)$ are known. We multiply both parts of equation (5) by p_{tu} and the sum by t:

$$\sum_{t\in T}\sum_{r\in R} f_{ur}H_{tr}(u) = \lambda_u\sum_{t\in T} p_{tu}, \quad u\in U.$$

Permuting the summing signs and taking into account the normalization conditions $\sum\limits_{t\in T} p_{tu} = 1$ and $\sum\limits_{t\in T} H_{tr}(u) = 1$, we get $\lambda_u = \sum\limits_{r\in R} f_{ur}$. Substituting λ_u again into (5), we have:

$$p_{tu} = \frac{\sum\limits_{r\in R} f_{ur}H_{tr}(u)}{\sum\limits_{r\in R} f_{ur}}.$$

Since the latent variables $H_{tr}(u)$) are not actually fixed and depend on p_{tu}, to maximize likelihood we need to apply an iteration that is a variant of the EM-algorithm. Each iteration consists of two steps. At the E-step, profiles p_{tu} are fixed and according to the Bayes formula latent variables $H_{tr}(u)$ are calculated. At the M-step, latent variables $H_{tr}(u)$ are fixed and profiles p_{tu} are calculated. Uniform distribution $p_{tu} = |T|^{-1}$ can be used as the initial approximation. Note that at any nonnegative initial approximation the nonnegative value of latent variables and profiles p_{tu} is guaranteed at all subsequent iterations.

The item profiles q_{tr} have thus far been fixed. The problem of optimization of the item profiles q_{tr} at fixed user profiles p_{tu} is set and solved in the "symmetric way"if we replace $u\leftrightarrow r$ and $q\leftrightarrow p$ in all formulas. Omitting further considerations, we write only the initial probabilistic model that is formally equivalent to models (2) and (4):

$$p(u,r) = \sum_{t\in T} q(r)q(t|r)p(u|t,r), \tag{6}$$

Algorithm 3.1. Symmetric Algorithm.

Require:

cross-tabulation matrix $F = \|f_{ur}\|_{U \times R}$;
number of topics $|T|$;
number of iterations at the outer loop I_{pq};
number of iterations at the inner loop I_{EM};

Ensure:

p_{tu} — user profiles, q_{tr} — item profiles;

1: estimates of a priori probabilities:
$p_u := \frac{1}{l}\sum_{r \in R} f_{ur}$; $q_r := \frac{1}{l}\sum_{u \in U} f_{ur}$ for all $u \in U,\ r \in R$;
2: initial approximation of profiles:
$p_{tu} := |T|^{-1}$; $q_{tr} := |T|^{-1}$ for all $u \in U,\ r \in R,\ t \in T$;
3: **for** the outer iteration loop I_{pq} times: **do**
4: $q(r|t) := \dfrac{q_{tr} q_r}{\sum_{r' \in R} q_{tr'} q_{r'}}$ for all $r \in R,\ t \in T$;
5: **for** the inner iteration loop I_{EM} times: **do**
6: E-step: $H_{tr}(u) := \dfrac{p_{tu} q(r|t)}{\sum_{t' \in T} p_{t'u} q(r|t')}$ for all $t \in T,\ u \in U,\ r \in R$;
7: M-step: $p_{tu} := \dfrac{\sum_{r \in R} f_{ur} H_{tr}(u)}{\sum_{r \in R} f_{ur}}$ for all $u \in U,\ t \in T$;
8: $p(u|t) := \dfrac{p_{tu} p_u}{\sum_{u' \in U} p_{tu'} p_{u'}}$ for all $u \in U,\ t \in T$;
9: **for** the inner iteration loop I_{EM} times: **do**
10: E-step: $H_{tu}(r) := \dfrac{q_{tr} p(u|t)}{\sum_{t' \in T} q_{t'r} p(u|t')}$ for all $t \in T,\ u \in U,\ r \in R$;
11: M-step: $q_{tr} := \dfrac{\sum_{u \in U} f_{ur} H_{tu}(r)}{\sum_{u \in U} f_{ur}}$ for all $r \in R,\ t \in T$;

where $p(u|t,r) = p(u|t)$ is a posteriori probability that the choice will be made by the user u under the condition that the choice is due to interest in topic t.

The main idea of the symmetric EM-algorithm is in the arrangement of two enclosed iteration loops. At the outer loop, two problems are solved alternately, i.e., first user profiles p_{tu} are optimized at fixed item profiles q_{tr} and then vice versa, profiles q_{tr} are optimized at fixed p_{tu}. Each optimization of profiles is implemented by the inner loop of the EM algorithm wherein latent variables $H_{tr}(u)$ and the profile approximation are calculated alternately. The implementation of the procedure is shown in more detail in Algorithm 3.1.

Note that the equivalence of all three models (2), (4), and (6) follows from the conditional probability determined in the following way:

$$p(u)p(t|u) = p(u,t) = p(t)p(u|t), \quad u \in U,\ t \in T;$$
$$q(r)q(t|r) = q(r,t) = p(t)q(r|t), \quad r \in R,\ t \in T.$$

However, in the standard pLSA algorithm and the symmetric one, the iteration is arranged in a different way, has different characteristics, and, generally speaking, yields different results.

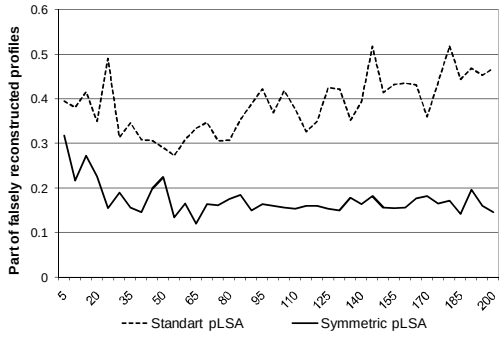

Рис. 1. Optimization of the topic number.

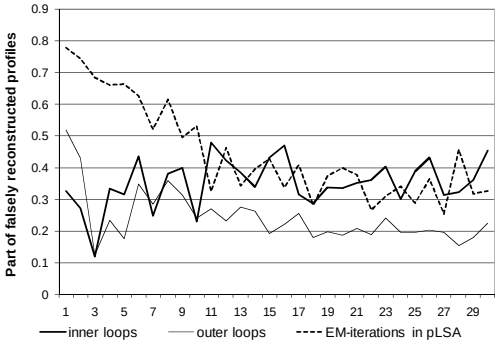

Рис. 2. Optimization of the iteration number.

4 Experiments

The suggested algorithm was tested and compared with standard algorithms using real data of Yandex search machine, goods purchase in a large furniture company, and model data.

4.1 The Search Engine Data

The search engine data were a protocol of clicks on documents returned by the search machine. The one week log file of the "Yandex"search machine, 3.7 Gb in size, contained the data on 129 000 resources, 14 606 users, and 207 696 user clicks. After the data preprocessing stage we retained 1024 most visited web sites as items and 1902 most active users (having made not less than 30 visits).

In the experiments different methods for constructing the similarity functions were compared including those based on the construction of site profiles.

In order to estimate the quality of the similarity functions, 400 sites were labeled by 12 classes. The quality criterion was found as the share of errors in classification of the labeled sites by the simple k nearest neighbor classifier. The site labeling was used only to estimate the quality quality but not used to calculate of profiles or similarity functions. The similarity of profiles was estimated through mean squared deviation with preliminary zeroing of noninformative components.

In order to construct profiles, two algorithms were used and compared: the standard pSLA and the symmetric algorithm. For the symmetric pSLA, coordinatewise optimization of parameters $|T|$, I_{pq}, and I_{EM} by the aforesaid criterion was used. After the profile construction, all components were zeroed in it except 3 maximum components and renormalization was undertaken.

Figure 1 shows the results of the optimization of the number of topics $|T|$ in both algorithms. The best results were achieved at $|T| = 65$ for the symmetric algorithm and $|T| = 55$ in standard pLSA.

Figure 2 shows the results of optimization of the number of inner and outer iterations for the two-stage (symmetric) algorithm and the number of EM-iterations for standard pLSA. The best quality was achieved for three iterations at the outer loop in the case of the two-stage algorithm and at 27 EM-iterations in standard pLSA.

Further increase in the number of topics $|T|$ or iterations can deteriorate the quality

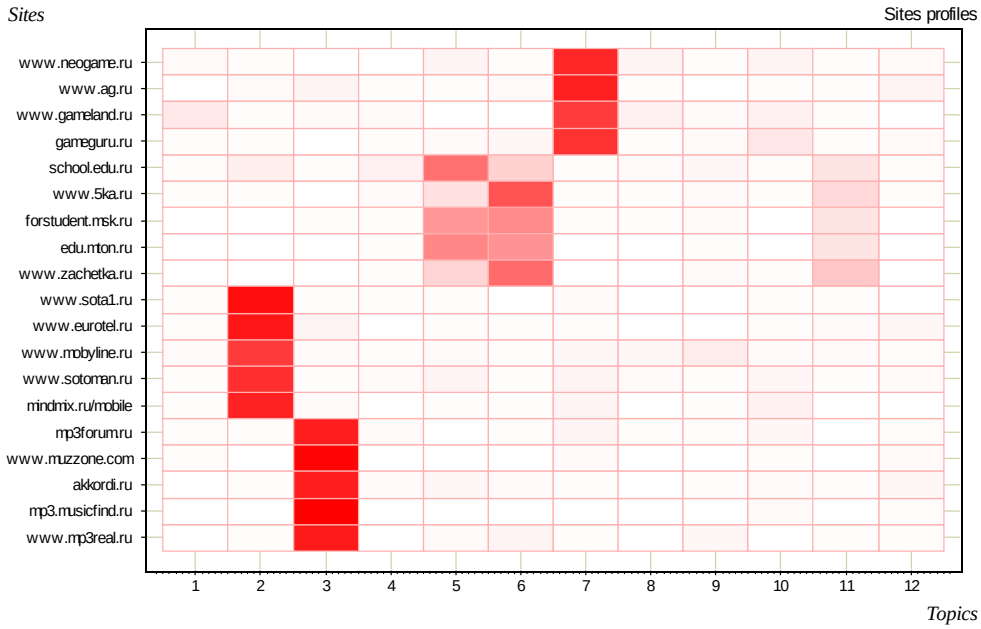

Рис. 3. Site profiles example.

of the profile. It can be interpreted as overfitting with an attempt to excessively fit profiles with respect to a particular data.

The semantic interpretation of the components was not specified a priori; nevertheless, the same components occurred to be dominants in profiles of sites having similar content. By way of example, the figure 3 shows the profiles of length $|T| = 12$ for some sites. Here we managed to ascribe topical interpretations to all 12 components. Therefore, interpretation of the profile components is possible already after the problem was solved on the basis of analysis of a small part of the resources whose topic is known.

In the symmetric algorithm inside each of two inner loops, latent variables $H_{tr}(u)$ and $H_{tu}(r)$ are calculated, which estimate conditional probabilities $H(t|u,r)$ in two different ways. The dependence of the mean modulus of deviation of these two estimates on the number of outer iterations is shown in Fig. 4. It is clearly seen that mean deviations become small after the first iteration and converge rapidly in the next iterations, which indirectly confirms the correctness of the algorithm.

There are numerous ways to introduce the distance functions (metrics) on users $\rho(u,u')$ and items $\rho(r,r')$. The most evident one is the mean squared deviation between profiles. The most evident one is the mean squared deviation between profiles. To give a visual check-up of metrics quality we used the multidimensional scaling (MDS) representing a finite set of points with a given pairwise distances as a two-dimensional scatter plot also called a similarity map (Fig. 5). Sites having similar topics turn out to form visually separable clusters on the map. Moreover, site profiles in each cluster have, as a rule, the same maximal components (see the example in the figure 3). The best clustering was achieved at preliminary zeroing of all components except for the three maximal ones in each profile.

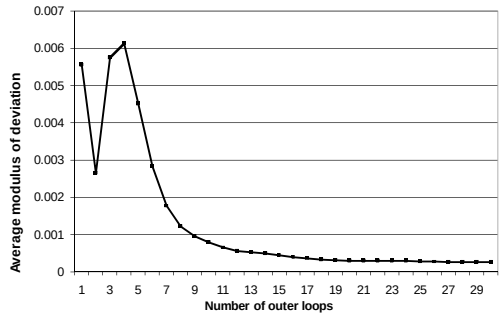

Рис. 4. Mean modulus of deviation of latent probabilities $H_{tr}(u)$ и $H_{tu}(r)$.

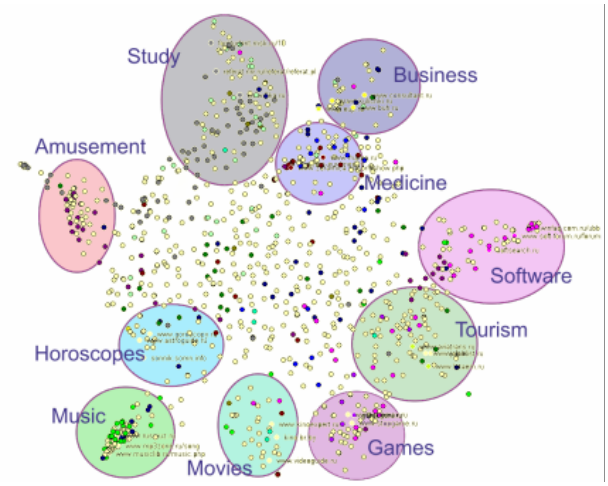

Рис. 5. Resource similarity map.

4.2 The Data on Goods Purchased

The data on goods purchased in the furniture company is the history of sales for three years of operation of the company. For analysis 1920 goods that were sold more than 100 times and 1328 users that purchased goods more than 30 times were chosen. By the users and goods chosen, the sampling of 112 256 facts of goods purchased was analyzed.

In order to estimate the quality by goods, 403 goods were divided into 12 categories. In a similar way, the data of the search engine were processed using the simple k nearest neighbor classifier (at $k = 5$) to estimate the fraction of correctly classified goods. Optimal functional (3% of classification errors) was achieved at the topic number $|T|$=30, 4 inner, and 4 outer iterations for the symmetric algorithm. In addition the profiles and the metric were constructed on the set of users.

4.3 Comparison of Distance functions

To compare the quality of different algorithms of collaborative filtering via the classification quality, the simple k nearest neighbor classifier was used. In Figs. 6 and 7, three distance functions are used to compare items, i.e., the distance between profiles in the symmetric algorithm, the distance between profiles in the standard pLSA, and the distance based on the Fisher's exact test for the data of the search engine and the furniture company.

Therefore, the symmetric model converges faster and provides a higher quality of the similarity functions compared to standard pLSA and the correlation measure of similarity based on Fisher's exact test.

4.4 Model Data

In the experiment on the model data at $|R| = 500$, $|U| = 1000$, true profiles were specified by random choice of two topics in each profile. The usage sampling was generated according to probabilistic model (4). The quality of profiles was estimated by the absolute deviation from true profiles, two maxima being revealed in the restored profiles and the remaining components being zeroed.

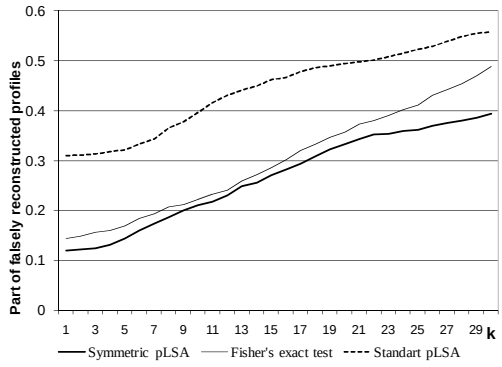

Рис. 6. Comparison of different metrics by kNN by the search engine data.

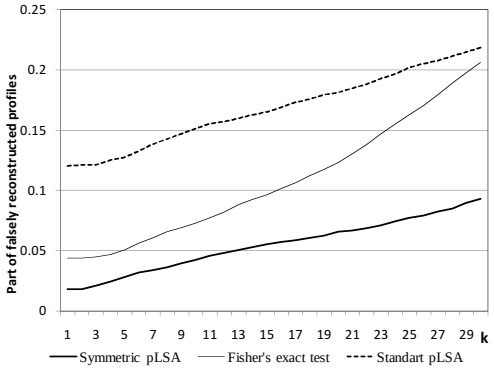

Рис. 7. Comparison of metrics by the furniture company data.

Parameter optimization has shown that the best quality of profile restoration is achieved at 6 iterations at the outer loop and 2 EM-iterations at the inner loop.

The algorithm divergence was also studied using model data. We should have found out at which minimal number of topics and minimal initial protocol length the algorithm converges. The number of topics was taken equal in the initial and restored profiles. Under the conditions of the given experiment, the algorithm proved to diverge at the number of topics less than 10 or a sampling length less than 700.

Список литературы

[1] *Adomavicius G., Tuzhilin A.* Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions // *IEEE Transactions on Knowledge and Data Engineering.* — 2005. — Vol. 17, no. 6.

[2] *Billsus D., Pazzani M. J.* Learning collaborative information filters // Proc. 15th International Conf. on Machine Learning. — Morgan Kaufmann, San Francisco, CA, 1998. — Pp. 46–54.

[3] *Brand M.* Fast online svd revisions for lightweight recommender systems // SIAM International Conference on Data Mining. — 2003.

[4] *Deerwester S., Dumais S. T., Furnas G. W., Landauer T. K., Harshman R.* Indexing by latent semantic analysis // *Journal of the American Society for Information Science.* — 1990. — Vol. 41. — Pp. 391–407.

[5] *Gaussier E., Goutte C.* Relation between plsa and nmf and implications // SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval. — New York, NY, USA: ACM, 2005. — Pp. 601–602.

[6] *Grcar M.* User profiling: Collaborative filtering // SIKDD 2004 at multiconference IS 12-15 Oct 2004, Ljubljana, Slovenia. — 2004.

[7] *Hofmann T.* Latent semantic models for collaborative filtering // *ACM Transactions on Information Systems.* — 2004. — Vol. 22, no. 1. — Pp. 89–115.

[8] *Hofmann T., Puzicha J.* Latent class models for collaborative filtering // International Joint Conference in Artificial Intelligence. — 1999.

[9] *Jin X., Zhou Y., Mobasher B.* Web usage mining based on probabilistic latent semantic analysis. — 2004.

[10] *Leksin V. A., Vorontsov K. V.* The client environment analysis: the reconstruction of latent profiles and similarity estimation of users and items // Mathematical Methods of Pattern Recognition–13. — MAKS Press, Moscow, 2007. — C. 488–491.

[11] *Leksin V. A., Vorontsov K. V.* The overfitting in probabilistic latent semantic models // Pattern Recognition and Image Analysis: new information technologies (PRIA-9). — Vol. 1. — Nizhni Novgorod, Russian Federation, 2008. — Pp. 393–396.

[12] *Marlin B.* Collaborative filtering: A machine learning perspective: Ph.D. thesis / Master's thesis, University of Toronto. — 2004.

[13] *Resnick P., Iacovou N., Suchak M., Bergstorm P., Riedl J.* GroupLens: An open architecture for collaborative filtering of netnews // Proceedings of ACM 1994 Conference on Computer Supported Cooperative Work. — Chapel Hill, North Carolina: ACM, 1994. — Pp. 175–186.

[14] *Schein A. I., Popescul A., Ungar L. H., Pennock D. M.* Generative models for cold-start recommendations // the SIGIR'01 Workshop on Recommender Systems. — 2001.

[15] *Srebro N., Rennie J. D. M., Jaakkola T. S.* Maximum-margin matrix factorization // Advances in Neural Information Processing Systems 17. — MIT Press, 2005. — Pp. 1329–1336.

[16] *Vorontsov K. V.* Preliminary data processing for a special class of recognition problems // *Comp. Maths Math. Phys.* — 1995. — Vol. 35, no. 10. — Pp. 1259–1267.

[17] *Vorontsov K. V., Rudakov K. V., Leksin V. A., Efimov A. N.* Web usage mining based on web users and web sites similarity measures // *Artificial Intelligence.* — 2006. — C. 285–288.