Collaborative Filtering for Music Recommender System

Elena Shakirova
National Research University of Electronic Technology
Zelenograd, Moscow, Russia
elena_shakirova_1@mail.ru

***Abstract*— Nowadays recommender systems is a software that is used as an important tool of e-commerce, which helps to analyze users' tastes and provide them with lists of products that they would like to prefer. This paper is an investigation of using collaborative filtering techniques for a music recommender system. Collaborative filtering is the technology that focuses on the relationships between users and between items to make a prediction. The goal of the recommender system is to compute a scoring function that aggregates the result of computing similarities between users and between items. We focus on the reviewing two strategies of collaborative filtering: user-based and item-based recommendations. For experimental purpose we explore different metrics to measure the similarity of users and items such as Euclidean distance, cosine metric, Pearson correlation and others. Finally, we compare different evaluations metrics that represent the effectiveness of the recommender system.**

Keywords — collaborative filtering, recommender systems.

I. Introduction

Development of recommender systems is a result of work of scientists from different fields: Data Mining, Machine Learning, Marketing, Information Technology, Statistics, Human Computer Interaction, Adaptive User Interfaces, Decision Support Systems, Consumer Behaviour. Recommender system is a software which uses history of users' preferences to provide them with a list of items (movies, audio records, articles, books and others) that they would like to prefer. Such systems include all steps of process: from extraction information about the users till presenting the recommendations to users. Two basic strategies of recommender systems are content filtering and collaborative filtering.

II. The Goals of Recommender Systems

Recommender systems change the human interaction with the Internet: it increases the degree of interactivity, enhances the users' experience. The main goal of using recommender systems is to increase sales of products and profit. That is why the technical goals are following [1]:

1. Relevance: Users are more likely to consume the products they find interesting.
2. Novelty: Recommender system should recommend to the target user things that he has not seen before. Repeated recommendation of popular items can also lead to reduction in sales diversity.
3. Serendipity: This goal differs from novelty because the recommendations should be unexpected, surprising to user. For example, if we know that the user likes one genre of music we can't recommend songs of that genre all the time, because these songs will be very similar and soon the user may lose interest. Recommender systems should help the user to discover new genres and expand range of interests.
4. Diversity: If the recommendation contains very similar elements, it increases the risk that the target user might like no one of them. If the recommendation contains elements of different types there is a greater chance that the user might like at least one of these items.

III. Collaborative Filtering

Collaborative filtering is predicting the users' preferences based on their past preferences and preferences of similar users. Collaborative filtering focuses on the relationship between objects [2-3].

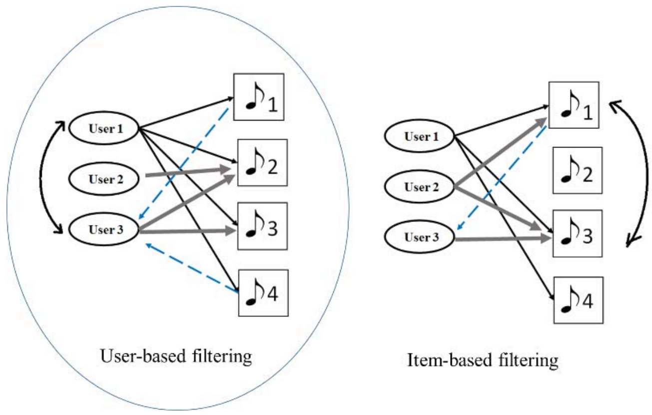

Fig. 1. Two basic approaches of collaborative filtering

User-based approach of collaborative filtering based on the fact that the ratings provided by similar users to the target user A are used to make recommendations for A.

Item-based approach is based on the fact that some items are often got together. The target user A likes one item from the set of items. So we can recommend him to consume other items from this set.

 978-1-5090-4865-6/17/$31.00 ©2017 IEEE

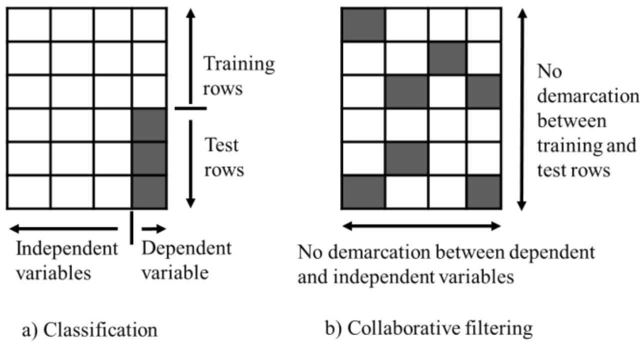

Fig. 2. Comparing the classification problem with collaborative filtering

Collaborative filtering can be considered as a generalization of the classification problem.

The missing class variable needs to be predicted from the feature variables values in the classification problem. The values of a dependent variable are known on training rows and they are missed on test rows.

In collaborative filtering, any of the matrix entries may be missing and need to be predicted in a data-driven way from the observed entries in the remaining matrix. There is no demarcation between training and test rows and no demarcation between independent and dependent variables. Shaded entries on the Fig.2. are missing and need to be predicted.

A. Similarity measure

In this paper, we assume U – a set of n users, I – a set of m items (songs). On the basis of users' history of listening we prepare the matrix of preferences $R = \{r_{ui}\} \in \mathbb{R}^{n\times m}$, which contains information how many times the user u listened to song i.

Modern recommender systems use different similarity functions to compute similarity between users and between items: Euclidean distance, cosine metric, Pearson correlation, Manhattan distance and others. Cosine metric can be used for both approaches of collaborative filtering: user-based and item-based.[4-5]

Let $I(u)$ denote as a set of items rated by user u. Then the cosine similarity between users u and v is computed by

$$w_{uv} = \frac{|I(u) \cap I(v)|}{|I(u)|^{\frac{1}{2}} |I(v)|^{\frac{1}{2}}} \tag{1}$$

The cosine similarity between items i and j is computed by

$$w_{ij} = \frac{|U(i) \cap U(j)|}{|U(i)|^{\frac{1}{2}} |U(j)|^{\frac{1}{2}}} \tag{2}$$

where $U(i)$ is a set of users which listened to a song i.

The alternative way to compute similarity is the conditional probability measure:

$$w_{uv} = P(u|v) = \frac{|I(u) \cap I(v)|}{|I(v)|} \tag{3.1}$$

$$w_{uj} = P(i|j) = \frac{|U(i) \cap U(j)|}{|U(j)|} \tag{3.2}$$

The parametric generalization of conditional probability measure:

$$w_{uv} = P(v|u)^{\alpha} P(u|v)^{1-\alpha} \tag{4.1}$$

$$w_{uj} = P(j|i)^{\alpha} P(i|j)^{1-\alpha} \tag{4.2}$$

where $\alpha \in [0,1]$ is parameter to tune.

$$w_{uv} = P(v|u)^{\alpha} =$$

$$= \frac{P(v \cap u)^{\alpha}}{P(u)^{\alpha}} \cdot \left(\frac{P(u \cap v)}{P(v)} \cdot \frac{P(v)^{\alpha}}{P(u \cap v)^{\alpha}} \right) =$$

$$= \frac{P(u \cap v)}{P(u)^{\alpha} \cdot P(v)^{1-\alpha}} = \frac{|I(u) \cap I(v)|}{|I(u)|^{\alpha} |I(v)|^{1-\alpha}}$$

Thus, the similarity function is:

$$w_{uv} = \frac{|I(u) \cap I(v)|}{|I(u)|^{\alpha} |I(v)|^{1-\alpha}} \tag{5.1}$$

$$w_{ij} = \frac{|U(i) \cap U(j)|}{|U(i)|^{\alpha} |U(j)|^{1-\alpha}} \tag{5.2}$$

It is obvious that the cosine similarity measure is a special case of a conditional probability measure.

B. Scoring function

We use simple weighted sum strategy for aggregating the information provided by similar users/items.

In the user-based type of recommendation the scoring function is computed by

$$h_{ui}^{U} = \sum\nolimits_{v \in U} f(w_{uv}) r_{ui} = \sum\nolimits_{v \in U(i)} f(w_{uv}) \tag{6.1}$$

The score of the item i is proportional to the similarities between the target user u and other users v who have the item i in their history of listening. This score is higher for items which are often rated by similar users.

In the item-based type of recommendation the scoring function is computed by

$$h_{ui}^{S} = \sum\nolimits_{j \in I} f(w_{ij}) r_{uj} = \sum\nolimits_{j \in I(i)} f(w_{ij}) \tag{6.2}$$

The score is proportional to the similarities between item i and other items already listened to by the user u.

The function $f(w)$ can be assumed monotonic not decreasing. Its role is to emphasize/deemphasize similarity contributions.

We use the exponential family of functions $f(w) = w^q$, $q \in \mathbb{N}$. The effect is following: when q is high, smaller weights drop to zero while higher ones are emphasized.

C. Ranking aggregation

There are many sources about music. Using them we can recommend music to customers in different ways. It is impossible to determine the only correct recommendation strategy. To achieve greater efficiency we can use ensemble of strategies. Each strategy has a special focus on the user. One strategy can recommend only a small number of elements that are likely users might like, but it does not contain elements that are likely users might like, which another strategy recommends. Different strategies give different recommendations. To achieve the technical goals of the recommender system such as relevance, novelty, serendipity and diversity it is better to use aggregating of the results of multiple strategies.

D. Evaluation metrics

Recommender systems can be evaluated using either online methods or offline methods. In an online system, the user reactions are measured with respect to the presented recommendations.

The most common techniques for testing recommendation algorithms are offline methods. When working with offline methods, accuracy measures can often provide an incomplete picture of the true conversion rate of a recommender system. It is important to design the evaluation system carefully so that the measured metrics truly reflect the effectiveness of the system from the user perspective.

The most popular measure of effectiveness is accuracy metrics. They are used to evaluate either the prediction accuracy of estimating the ratings of specific user-item combinations or the accuracy of the top-k ranking predicted by a recommender system [6].

One of the most popular methods of computing accuracy metric is the root mean squared error, or RMSE, which is borrowed from the literature on regression modelling:

$$RMSE = \sqrt{\frac{1}{|k|}\sum\nolimits_{(u,i)\in k}(p_{ui} - r_{ui})^2} \quad (7)$$

where p_{ui} is predicted rate, k — quantity of testing rates. The smaller value RMSE is better.

Another commonly used metric is mean average precision (mAP).

Precision at K is defined as the proportion of correct recommendations within the top-k of the predicted ranking. Let y denote a ranking over items $Y \leftrightarrow I: y(p) = i$, it means that item i is ranked at position p.

$$p_k(u, y) = \frac{1}{k}\sum\nolimits_{p=1}^{k} r_{uy(p)} \quad (8)$$

This formula is simple, is has a disadvantage – it does not take into account the order of the items in the "top".

Average precision at K is average precision at each recall point:

$$AP(u, y) = \frac{1}{\tau_u}\sum\nolimits_{p=1}^{\tau} p_k(u, y) r_{uy(p)} \quad (9)$$

The idea of mAP is compute AP for each user:

$$mAP(u, y) = \frac{1}{N}\sum\nolimits_{p=1}^{N} AP(u, y_u) \quad (10)$$

IV. Conclusions

In this paper we reviewed collaborative filtering methods and evaluation metrics to estimate effectiveness of recommender systems. We have prepared a theoretical basis for the implementation of collaborative filtering techniques for a music recommender system. We plan to vary different parameters such as similarity measure, scoring function to improve the effectiveness of recommender systems.

The first results obtained fixing the parameter q and α. A correct setting of the parameter q is critically important. The best result was obtained for non-trivial α=0.15 and q=3 in the item-based case, α=0.3 and q=5 in the user-based case.

References

[1] Charu C. Aggarwal «Recommender Systems. The Textbook», 2016, 518p.

[2] A. Gomzin, A. Korshunov, “Recomender Systems: an overview of current approaches”. Press-print. Moscow: The Institute for System Programming of Russian Academy of Sciences, 2012, 20 p.

[3] T. Segaran “Programming Collective Intelligence”, O’Reilly Media Inc, 2007, 368p.

[4] Francesco Ricci, Lior Rokach, Bracha Shapira, Paul B. Kantor «Recommender Systems Handbook», 2011, 845p.

[5] Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich, “Recommender Systems. An Introduction”, 2011

[6] Company Blog E-Contenta. "Quality Metrics ranking".Access:https://habrahabr.ru/company/econtenta/blog/303458/