ECAI 2008
M. Ghallab et al. (Eds.)
IOS Press, 2008

doi:10.3233/978-1-58603-891-5-823

Probabilistic Reinforcement Rules for Item-Based Recommender Systems

Sylvain Castagnos and **Armelle Brun** and **Anne Boyer**[1]

Abstract. The Internet is constantly growing, proposing more and more services and sources of information. Modeling personal preferences enables recommender systems to identify relevant subsets of items. These systems often rely on filtering techniques based on symbolic or numerical approaches in a stochastic context. In this paper, we focus on item-based collaborative filtering (CF) techniques. We propose a new approach combining a classic CF algorithm with a reinforcement model to get a better accuracy. We deal with this issue by exploiting probabilistic skewnesses in triplets of items.

1 INTRODUCTION

This paper focuses on recommender systems based on collaborative filtering techniques (CF). CF algorithms provide personalization by exploiting the knowledge of a similar population and predicting future interests of a given user (called "active user") as regards his/her known preferences. In practical terms, this kind of algorithms is broken down into 3 parts. Firstly, the system needs to collect data about all users under the form of explicit and/or implicit ratings. Secondly, this data is used to infer predictions, that is to say to estimate the votes that the active user would have assigned on unrated items. Finally, the recommender system suggests to the active user items with the highest estimated values.

As the highest values of prediction are the only ones of interest, we propose a new model that focuses on prediction of high values, to improve accuracy. As the error on these values may be significant with a usual item-based CF algorithm, we propose to re-evaluate them by using reinforcement rules. The latter are automatically inferred by selecting triplets of items in the dataset according to their joint probabilities.

After a short state-of-the-art, we propose a model combining an Item-Based Algorithm (CIBA) with reinforcement rules. We call it "Reinforced Item-Based Algorithm" (RIBA).

2 RELATED WORK

2.1 Notations

To help the readers, we introduce the following notations:

- $U = \{u_1, u_2, \ldots, u_n\}$ is the set of the *n* users;
- $I = \{i_1, i_2, \ldots, i_m\}$ is the set of the *m* items;
- U_k refers to the set of users who have rated the item i_k;
- I_a is the list of items rated by the active user u_a;
- $v(j, k)$ is the vote of the user u_j on the item i_k;
- v_{min} and v_{max} are respectively the minimum and maximum values on the rating scale;
- v_l and v_d are the thresholds for liked and disliked items;
- $\bar{i_k}$ is the average of all users' ratings on i_k;
- $s(k, t)$ the similarity measure between i_k and i_t;
- $p(a, k)$ is the prediction of u_a for item i_k;
- $pr(a, k)$ is the prediction of u_a for i_k with reinforcement rules.

2.2 Classical Item-Based Algorithm

To supply the active user with information that is relevant to his/her concerns, the system first builds his/her profile under the form of a vector of item ratings. Profiles of all users are then aggregated in a user-item rating matrix, where each line corresponds to a user, and each column to an item.

Item-based CF is based on the observation that the consultation of a given item often leads to the consultation of another one [4]. To translate this idea, the system builds a model that computes the relationships between items. Most of time, the model is generated by transforming the user-item matrix in an item-item matrix. This conversion requires the computation of similarities between items (i.e. columns of the user-item rating matrix). The active user's predictions are then computed by taking into account his/her known ratings, and the similarities between the rated items and the unrated ones.

In this paper, we propose a model that can be plugged on an item-based collaborative filtering algorithm in order to refine some predictions. In this subsection, we present the Classical Item-Based Algorithm (CIBA) used as a base for our model.

When implementing an item-based CF algorithm, the designer has to choose a pairwise similarity metric, and a prediction formula. We decide to use the Pearson correlation coefficient, as litterature shows this similarity metric works better [4]. Consequently, we fill the item-item similarity matrix by applying the equation 1 for each pair of items.

$$s(k,t) = \frac{\sum_{u_j \in U_k \cap U_t} (v(u_j, i_k) - \bar{i_k})(v(u_j, i_t) - \bar{i_t})}{\sqrt{\sum_{u_j} (v(u_j, i_k) - \bar{i_k})^2} \sqrt{\sum_{u_j} (v(u_j, i_t) - \bar{i_t})^2}} \quad (1)$$

We also compared different prediction formulas [2, 3]. We chose to adapt the weighted sum of the deviation from the mean, usually used in user-based framework, to an item-based context (cf. formula 2). This formula leads to the highest accuracy.

$$p(a,k) = max\Big(v_{min}, min\Big(\frac{\sum_{i_t \in I_a} s(k,t) \times (v(a,t) - \bar{i_t})}{\sum_{i_t \in I_a} |s(k,t)|} + \bar{i_k}, v_{max}\Big)\Big) \quad (2)$$

[1] LORIA - University Nancy 2, email: {sylvain.castagnos, armelle.brun, anne.boyer}@loria.fr

3 REINFORCED ITEM-BASED ALGORITHM

Our model, called "Reinforced Item-Based Algorithm" (RIBA), is a combination of a Classic Item-Based Algorithm (CIBA) and probabilistic association rules that come to reinforce some predictions. This section is dedicated to the way to combine these two approaches.

3.1 Probabilistic Reinforcement Rules

In standard item-based CF algorithms, similarities are computed between each neighbor item and the target item. We argue that, in some cases, pair-wise similarities may be insufficient to explain the interest of a user for an item. We propose here to evaluate similarities of triplets, rather than pairs of items, before the prediction phase. A triplet is an association rule where the premisse is made up of two terms. The conclusion is the reinforced item.

To illustrate this statement, we can consider three items i_k="Cinderella", i_t="Scary Movie", and i_w="Shrek". A user may have liked i_k which is a fairytale without appreciating i_w. At the same time, a user who enjoys the horror film parody i_t should probably rate lowly i_w. However, a film goer who likes both fairy tales and parodies will take fun when watching Shrek.

Let introduce the following additional notations:

- I_k denotes the fact to like i_k, i.e. when $v_{j,k} \geq v_l$;
- $\overline{I_k}$ is the fact to dislike i_k, i.e. when $v_{j,k} \leq v_d$;
- $\ddot{I_k}$ when i_k has not been rated (by convention, the vote is equal to 0 in this case);
- $\breve{I_k}$ when i_k has been rated (the vote is between v_{min} and v_{max});
- $P(I_k,\ I_t\ , I_w)$ the probability to like the three items i_k, i_t, and i_w;
- $P(I_k,\ I_t\ \mid\ \ddot{I_w})$ the probability to like i_k and i_t for users who have not rated i_w;
- $N(I_k,\ I_t\ ,\ \ddot{I_w})$ the number of users who have liked i_k and i_t, and not rated i_w.

Then a rule $< I_k, I_t > \Rightarrow I_w$ means that I_k alone does not explain I_w, I_t alone does not explain I_w, but $< I_k, I_t >$ together explain I_w. Let notice that 3 items could lead up to 8 reinforcement rules, such as $< \overline{I_k}, I_t > \Rightarrow I_w$, or $< \overline{I_k}, \overline{I_t} > \Rightarrow \overline{I_w}$.

3.2 Determination of the reinforcement rules

A triplet $< i_k, i_t, i_w >$ is candidate to be a reinforcement rule $< I_k, I_t > \Rightarrow I_w$ if the similarities between each pair of its items are around the mean similarity. In that case, the resulting reinforcement rule could impact accurately I_w.

Thus a triplet is a candidate if the following constraints are satisfied:

$$0 < t_{min} \leq |s(k,t)| \leq t_{max} < 1 \tag{3}$$

$$0 < t_{min} \leq |s(k,w)| \leq t_{max} < 1 \tag{4}$$

$$0 < t_{min} \leq |s(t,w)| \leq t_{max} < 1 \tag{5}$$

where t_{min} and t_{max} respectively refer to the minimum and maximum similarity threshold that will be set experimentally.

For each reinforcement rule candidate, we compute the probability of the corresponding triplet. Thus for each triplet $< i_k, i_t, i_w >$, we compute the joint probabilities $P(I_k,\ I_t\ ,\ I_w)$, $P(I_k,\ I_w\ \mid\ \ddot{I_t})$, and $P(I_t,\ I_w\ \mid\ \ddot{I_k})$:

$$P(I_k,\ I_t\ ,\ I_w) = \frac{N(I_k,\ I_t\ ,\ I_w)}{N(\breve{I_k},\ \breve{I_t}\ ,\ \breve{I_w})} \tag{6}$$

$$P(I_k,\ I_w\ \mid\ \ddot{I_t}) = \frac{N(I_k,\ \ddot{I_t}\ ,\ I_w)}{N(\breve{I_k},\ \ddot{I_t}\ ,\ \breve{I_w})} \tag{7}$$

If this probability is significantly higher than the probability of each pair of its items, than this triplet is selected as a reinforcement rule. The reinforcement rule $< I_k, I_t > \Rightarrow I_w$ is then generated when the following conditions are fulfilled:

$$P(I_k,\ I_t\ ,\ I_w) \gg P(I_k,\ I_w\ \mid\ \ddot{I_t}) \tag{8}$$

$$P(I_k,\ I_t\ ,\ I_w) \gg P(I_t,\ I_w\ \mid\ \ddot{I_k}) \tag{9}$$

3.3 Rating Refining Process

The generated reinforcement rules allow to refine some predictions. For each prediction $p(a,k)$, a rule is applicable if i_k corresponds to the item in the conclusion and if the premises are valid. Each applicable rule associated to $p(a,k)$ is set to a weight $w(r,a,k)$. This weight is equal to 1 when the conclusion of the rule is I_k, and it $w(r,a,k) = -1$ if the conclusion of the rule is $\overline{I_k}$.

We call $AR_{a,k}$ the set of rules that can be applied for the prediction computation of $p(a,k)$.

We refine the vote with the following equation:

$$pr(a,k) = p(a,k) + \frac{coef * \sum_{r \in AR_{a,k}} w(r,a,k)}{\sum_{r \in AR_{a,k}} |w(r,a,k)|} \tag{10}$$

"coef" is the coefficient of refinement. The greater this coefficient is, the more important the refinement will be.

4 CONCLUSION

In order to increase the quality of suggestions in recommender systems, we proposed a new approach combining an item-based collaborative filtering model with reinforcement rules. These rules are generated automatically by analyzing joint probabilities in triplets, and allow us to refine predictions of items where pair-wise similarities are not sufficient. The experiments show that this approach improves significantly the accuracy of high predictions. We validate our model by using the MovieLens dataset (http://www.movielens.org/) and get an improvement from 6 to 8% as regards the High MAE measure [1].

REFERENCES

[1] Linas Baltrunas and Francesco Ricci, 'Dynamic item weighting and selection for collaborative filtering', in *Workshop PriCKL07, in conjunction with the 18th European Conference on Machine Learning (ECML) and the 11th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD)*, Warsaw, Poland, (September 2007).

[2] Sylvain Castagnos and Anne Boyer, 'A client/server user-based collaborative filtering algorithm: Model and implementation', in *4th Prestigious Applications of Intelligent Systems special section (PAIS 2006), in conjunction with the European Conference on Artificial Intelligence (ECAI 2006)*, Riva del Garda, Italy, (August 2006).

[3] Bradley N. Miller, Joseph A. Konstan, and John Riedl, 'Pocketlens: Toward a personal recommender system', in *ACM Transactions on Information Systems*, volume 22, pp. 437–476, (July 2004).

[4] Badrul M. Sarwar, George Karypis, Joseph A. Konstan, and John Reidl, 'Item-based collaborative filtering recommendation algorithms', in *World Wide Web*, pp. 285–295, (2001).