An Effective Model-Based Trust Collaborative Filtering for Explainable Recommendations

1st Hafed Zarzour  
Dept. of Computer Science  
University of Souk Ahras  
Souk Ahras, Algeria  
hafed.zarzour@gmail.com

2nd Yaser Jararweh  
Dept. of Computer Science  
Jordan University of Science and Technology  
Irbid, Jordan  
yijararweh@just.edu.jo

3rd Ziad A. Al-Sharif  
Dept. of Software Engineering  
University of Souk Ahras  
Souk Ahras, Algeria  
zasharif@just.edu.jo

Abstract—Nowadays, many companies through the world wide web like YouTube, Netflix, AliExpress and Amazon, provide personalized services as recommendations. Recommender systems use the related information about products or services to suggest the most relevant of them to particular users. The recommendation is usually made based on the prediction of the users’ preferences, improving overall satisfaction and loyalty of users. Current recommender systems typically are classified into two main categories: content-based filtering and collaborative filtering [2], [3]. Content-based filtering methods rely on the item’s description and past preferences of the user to attempt to recommend items based on feature representation of the services or products [4]. The problem with these methods is how to find the user’s preference using the item contents. In contrast, collaborative filtering methods focus on the preference pattern identification within a community of users in which the recommendation is generated based only on gathered data about users’ and items’ ratings profiles [5]. These methods can be divided into two classes: memory-based and model-based. For the memory-based filtering, similarity measures such as person correlation coefficient [6] and cosine similarity [7], are used to determine the relationships between users or items. Thus, a user-item rating matrix is used to make prediction. On the other hand, the model-based filtering requires a model to be created and trained from training data before making recommendation. For example, in [8], the authors developed a new effective recommender system for TED lectures to enhance the recommendation quality. Authors followed three steps: (1) employing the person correlation coefficient similarity with TED lectures to generate the TED user-user matrix; (2) using the k-means clustering technique to group users that are having the same preferences in clusters and create a predictive model; finally, (3) using the created model to make relevant recommendations.

Moreover, the authors in [9] proposed to use the deep neural network technology to build a recommender system that is able to predict the rating scores based on forward propagation algorithm. They combined both the users embeddings and items embeddings with the deep neural network for enhancing the recommendation according to two steps. The first one involves the creation of dense numeric representations for all users and items, while the second one involves the use of the deep neural network model to predict the ratings’ scores using the algorithm of forward propagation.

Despite that most of the existing recommender systems give accurate recommendation results, they do not provide explainable recommendation support. Thus, integrating such a support in a recommender system can play an important role, it would be interesting to develop new methods for explainable recommendations to serve as bridge between users and new generation of recommender systems. The main goal of the explainable recommendation is to get a clear response on why a particular item is recommended to the particular user? This can help in enhancing the trustworthiness, persuasiveness, effectiveness, relevance, comprehensibility, and transparency of recommender systems. Therefore, in this paper we propose a new effective model-based trust collaborative filtering for explainable recommendations that aims not only to improve
the quality of recommendation but also to provide an efficient support for explainable recommendations using the trustworthiness aspect. Our solution takes into consideration how an item is recommended as well as why it is recommended at the same time. The proposed method is evaluated on Amazon Instant Video dataset \(^1\) in terms of Root Mean Square Error (RMSE), which is commonly used to measure the differences between the predicted values by the model and the observed values [10].

The rest of the paper is organized as follows: Section II presents the related work. Section III describes the proposed approach. Section IV illustrates the experimental results. Finally, Section V concludes our findings and proposes the plans for future work.

II. RELATED WORK

The idea of explainable recommendation is to develop a recommendation algorithm that can be able to solve the problem of why we recommend items to a particular user by giving him/her an efficient clarification about the recommendation task. Indeed, with the explainable recommendation, the process of recommendation is not yet considered as a black box as before.

Recently, many researchers have tried to make recommendation more understandable and explainable [11]–[17]. For example, Ribeiro et al. [18] presented LIME, an algorithm that is capable of explaining the predictions of any classifier in a faithful way. They also introduced SP-LIME to filter the representative predictions from those redundant and non-representative. Ai et al. [19] provided a knowledge-base embeddings framework for explainable recommendation in which they used a soft matching algorithm to provide explanations about the recommended items. The soft matching algorithm was divided for exploring the explanation paths that exist between the recommended items and a user.

Hou et al. [20] proposed to generate explainable recommendations using aspect-based matrix factorization, which could enhance the prediction of rating with fusion of aspect information. Lin et al. [21] developed a neural network framework, which was capable to generate abductive comments when providing outfit recommendations. They employed a convolutional neural network with a mutual attention function to outfit matching and recurrent neural network with a cross-modality attention function to obtain a concise sentence.

However, to the best of our knowledge, the trustworthiness combined with the collaborative filtering have not been together explored yet. Additionally, no work has been conducted about the trust modeling for Amazon Instant Video dataset. Thus, the contributions of this paper are to define a trustworthiness model as a new explainable recommendation data type, use this model within a collaborative filtering and carry out experiments on Amazon Instant Video dataset.

\(^1\)Amazon product data, http://jmcauley.ucsd.edu/data/amazon/

III. METHODOLOGY AND APPROACH

In this section, we present a new effective model-based trust collaborative filtering for explainable recommendations that claims to help users to get not only the recommendation but also an explanation for each recommended item. It involves users’ similarities and trustworthiness in order to produce recommendation results with their corresponding explanations.

A. Model-based trust for explainable recommendations

To help users know at any time why they get the specific results from the recommender system, we describe a new trustworthiness model. This model can make the recommender systems more efficient in encouraging users accepting the recommended items.

1) Trustworthiness: The trustworthiness characterizes the relationship that exists between the target users and their neighbors by providing a real value. This value specifies the degree to which the active user chooses the items based on his or her neighbors.

The trustworthiness between two users increases or decreases depending on the reaction of the active user. For example, when the active user Bob interacts with items recommended using the neighbor Alice, the trustworthiness between them increases and vice versa.

The trustworthiness between two users \(u \) and \(v \) is defined [22] by Equation 1, where \(R_{u,v} \) denotes the number of items that are really recommended to the active user \(u \) based on the user \(v \), likewise, \(S_{u,v} \) denotes the number of times in which the user \(v \) was selected as neighbor of the user \(u \).

\[
T(u, v) = \frac{R_{u,v}}{S_{u,v}} \tag{1}
\]

2) Explainable recommendation data type: The main problem to be addressed in the context of explainable recommendation is to respond to the question: why a specific item is recommended to a given user? Providing an explanation process can effectively assist those who are using the recommender systems in understanding the reasons behind these recommendations.

To enable efficient generation of useful explanations, we introduce a new data structure called explainable recommendation data type that includes all elements serving for the explanation. Indeed, an explainable recommendation data type is a data structure with five elements \((I, U, R, N, T)\) where \(I \) is the recommended item, \(U \) is the active user, \(R \) is the predicted rating, \(N \) is the number of users sharing the same preferences with the active user and \(T \) is the average trustworthiness; users and items are presented by their identifiers, respectively.

Figure 1 shows that using the explainable recommendation data type, an explanation can be presented to an active user as follows:

**Hi Bob, the item “Home alone” was recommended to you because there are 15 users having the same taste as you with a 95% of average trustworthiness and a 4.5 predicted rating.**
B. Model-based trust explainable recommendation collaborative filtering algorithm

In this work, a collaborative filtering recommendation algorithm based on trust explainable recommendation is proposed. Figure 2 shows the main steps of the proposed algorithm.

**Step 1**: Get the active user and generate the user-item rating matrix. This step should be proceeded by a data preprocessing procedure, which consists of removing items with fewer ratings and managing missing values. In the resulted user-item rating matrix, users are represented by the rows and items by the columns.

**Step 2**: Calculate the similarities between users, this step is computed using the cosine similarity measure. This step generates the user-user similarity matrix. The cosine similarity measure is one of the most popular metrics used in collaborative filtering recommendation algorithms [23], [24]. It enables to compute the angle cosine between two vectors of user’s $u$ and $v$, see Equation 2.

**Step 3**: Select the most similar neighbors of the active user to be considered in the calculation of the prediction.

**Step 4**: Predict the rating values of the active user based only on the selected neighbors with their ratings. The prediction function is defined in Equation 3, where $u$ represents the active user, $k$ represents the item, $s(u)$ represents the list of neighbors and $cos()$ is the cosine similarity function.

**Step 5**: Select the top $N$ items of users and calculate the trustworthiness using Equation 1.

**Step 6**: Use the explainable recommendation data type with the corresponding values to generate recommendation results with their explanations, see Figure 1.
\[ \cos(u, v) = \frac{u \cdot v}{\|u\| \cdot \|v\|} \tag{2} \]

\[ P(u, k) = \frac{\sum_{u_a \in S(u)} \cos(u, u_a)(r_{ua} - \bar{r}_u)}{\sum_{u_a \in S(u)} \cos(u, u_a)} \tag{3} \]

IV. EXPERIMENTS

The algorithms presented in this paper are implemented in Python where the experimental platform configuration is Intel Core i7-8550U CPU @ 1.80GHz x 8, 8GB memory. The used operating system is Ubuntu 18.04 LTS.

The experiments are conducted on the dataset called Amazon Instant Video \(^2\) from Amazon. Recently, this dataset was extensively used in the domain of recommender systems. It contains 37126 reviews about 1685 products or videos from Amazon made by 5130 reviewers, where each of anonymous users and items have \(k\) reviews. The conventional user-item rating matrix corresponds here to the reviewer-video rating matrix.

In these experiments, two different methods are considered. The first one is the collaborative filtering with the Model-based trust explainable recommendations, while the second one is the collaborative filtering without the Model-based trust explainable recommendations. The RMSE shown in Equation 4 is applied to evaluate the prediction accuracy for both methods.

\[ \text{RMSE} = \sqrt{\frac{1}{n} \sum_{u=i}^{n} |r_{ui} - \bar{r}_{ui}|^2} \tag{4} \]

Table I shows the values of RMSE along the neighbor sizes that are ranging from 10 to 80 for the two methods. As we can observe, the collaborative filtering with the Model-based trust explainable recommendations and the collaborative filtering without the Model-based trust explainable recommendations have identical experimental results and the optimal size of neighbor of each method can be easily obtained. The RMSE decreases as neighbor size increases from 10 to 30 and then there is no real improvement. Therefore, the optimal size of neighbors is 30. From Table I, it can be concluded that obtaining the same results of prediction accuracy for the collaborative filtering with and without the Model-based trust explainable recommendations means that the explainable recommendations benefited the users of more trustworthiness and explanation without causing a negative influence on the performance of the recommender system.

\(^2\) Amazon product data, http://jmcauley.ucsd.edu/data/amazon/

<table>
<thead>
<tr>
<th>Method</th>
<th>Number of neighbors</th>
<th>RMSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collaborative filtering with the model-based trust explainable recommendation</td>
<td>10</td>
<td>1.0444</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>1.0391</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>1.0389</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>1.0392</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>1.0403</td>
</tr>
<tr>
<td></td>
<td>60</td>
<td>1.0416</td>
</tr>
<tr>
<td></td>
<td>70</td>
<td>1.0427</td>
</tr>
<tr>
<td>Collaborative filtering without the model-based trust explainable recommendation</td>
<td>80</td>
<td>1.0435</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>1.0444</td>
</tr>
<tr>
<td></td>
<td>20</td>
<td>1.0391</td>
</tr>
<tr>
<td></td>
<td>30</td>
<td>1.0389</td>
</tr>
<tr>
<td></td>
<td>40</td>
<td>1.0392</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>1.0403</td>
</tr>
<tr>
<td></td>
<td>60</td>
<td>1.0416</td>
</tr>
<tr>
<td></td>
<td>70</td>
<td>1.0427</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>1.0435</td>
</tr>
</tbody>
</table>

V. CONCLUSION AND FUTURE WORK

The development of the technology of recommender systems has changed the strategies of many companies providing online services. Hence, these strategies are constantly changing since the emergence of the explainable recommendation. Recommender systems with the features of explainable recommendation have become more transparent and trustworthy. Due to the importance of building these systems with such features, we proposed a new effective model-based trust collaborative filtering for explainable recommendations considering users’ similarities and trustworthiness to generate recommendation results with their explanations.

The experiments that we have conducted on the Amazon Instant Video dataset demonstrate that prediction results are the same for the two collaborative filtering methods, with and without the model-based trust explainable recommendations; implying that the trustworthiness ensured by our approach is useful and it has no negative effects on the recommendation quality.

However, for future work, we are looking forward to investigate other methods such as clustering algorithms with the principle component analysis \([25]\) and collaborative filtering recommendations based on dimensionality reduction that would be coupled with a clustering method. This clustering \([26]\) may improve both the recommendations and explanations. Additionally, we are aiming at conducting other experiments with big datasets to confirm the effectiveness of our method.

REFERENCES


