International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 07 | July - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

Detection and Correction of Natural Noise in the Recommendation System

Rahul Kumar Deo
M. Tech
Department of Computer Science and Engineering
R. V. S. College of Engineering and Technology, Jamshedpur, Jharkhand, India
Jharkhand University of Technology, Ranchi.

Akansha Sinha
M. Tech
Department of Computer Science and Engineering
R. V. S. College of Engineering and Technology, Jamshedpur, Jharkhand, India.
Jharkhand University of Technology, Ranchi.

Deobrata Kumar
HOD
Department of Computer Science and Engineering
R. V. S. College of Engineering and Technology, Jamshedpur, Jharkhand, India
Jharkhand University of Technology, Ranchi.

Abstract

Recommender systems have become a mainstream research field in IT. Recommender systems act as a supporting tool for helping users trying to obtain information that best fits their preferences and needs, in scenarios where the information overload is an important drawback. The most popular version of RS (recommender systems), learns from preferences of the user about a predefined set of known items, and predicts the preference degree about unknown items. With this goal, many applications have been built to recommend different types of items like music, books, TV shows, movies, jokes, news, scientific papers, web pages, and so on. They have covered diverse areas like e-commerce, e-learning, e-services, tourism, and software engineering. The two most used techniques in the development of recommender systems are the content-based and the collaborative filtering. Content-based recommender systems, suggest items with similar features to those that the user chose in the past. On the other hand, collaborative filtering recommender systems recommend items that other similar users liked in the past.

1. Introduction

The main objective of this is to detect and correct Natural Noise in the Recommendation System. This aims to deliver a novel algorithm (approach) to find out the inconsistent ratings that are involuntarily introduced by users and then correct them instead of eliminating them from the dataset. The two most used techniques in the development of recommender systems are the content-based and the collaborative filtering. Content-based recommender systems, suggest items with similar features to those that the user chose in the past. On the other hand, collaborative filtering recommender systems recommend items that other similar users liked in the past.

2. Literature Survey

R.Y. Toledo and Martínez have proposed an approach to detect and correct natural noise by using only the current ratings in the dataset without needing any additional information.

Pengyu Wang and Zhu have proposed a scheme based on fuzzy theory to manage the natural noise in the Recommendation System. Collaborative filtering (CF) as one of the most popular recommendation models has been widely used in research and practice. The basic idea of CF is to search the N most similar neighbors for a given user by similarity measures, and then produce predictions for the user according to neighbors' opinions. Thus, the neighbor selection is a critical step for CF, which determines the quality of

recommendation results.

J.M. Pujol and Oliver come up with a new hybrid approach where users are required to rate the same items again to check whether the previous ratings contain natural noises, which can avoid the influence of accidental factors in the rating process.

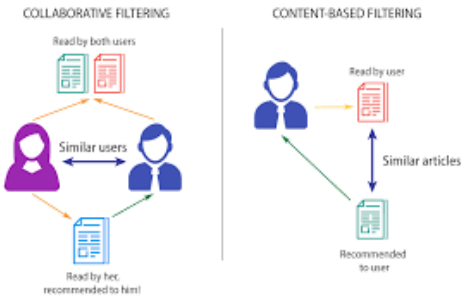
Pham and Jung recommend an interactive system is built to provide users with some guidance in time, and even some experts are invited to examine the historical ratings and judge whether natural noises in ratings according to their professional experience. Although these methods can avoid some possible noises, they may be hard to implement in real applications due to extra time, labor, and resource costs.

Bag and Tiwari proposed a method which only uses the features of rating data to process natural noise and possesses the advantage of easy deployment in RS. In these methods, the ratings in RS are firstly classified into three categories (i.e., low, medium, and high ratings). Then, the users in RS are also divided into three categories (i.e., positive, average, and negative users) according to their historical ratings. Similarly, items are divided into preferred, av-preferred, and no-preferred items.

R. Yera and Martínez proposed a novel approach to improve the ability of detecting natural noise where the researchers do not group each rating to a single class but create the membership function to compute the membership degree that the rating belongs to different categories. The membership function provides a basic way to describe the fuzzy profiles of users and items. Then, the noise detection rules are proposed based on the user and item profiles. But these methods still adopt the strategy that re-predict a new rating to correct noise, which is time-consuming.

3. Architecture

The architecture employed in the development of recommender systems are the collaborative filtering and the content-based filtering.



3.1 Collaborative filtering

Collaborative filtering is based on the similarity matrix. For user-based collaborative filtering, the user similarity matrix will consist of some distance metric that measures the similarity between any two pairs of users. Likewise, the item-similarity matrix will measure the similarity between any two pairs of items. Collaborative filtering is a method of making predictions about the interests of users by analyzing the taste of users which are similar to the said user. The CF techniques are mainly categorized into 2 types namely Memory based approach and Model based approach.

Memory-Based Collaborative Filtering can be divided into two main sections: user-item filtering and item-item filtering. A user-item filtering takes a particular user, find users that are similar to that user based on similarity of ratings, and recommend items that those similar users liked. In contrast, item-item filtering will take an item, find users who liked that item, and find other items that those users or similar users also liked. It takes items and outputs other items as recommendations.

In Model-Based Collaborative Filtering approach, CF models are developed using machine learning algorithms to predict user's rating of unrated items.

3.2 Content based filtering

In content-based filtering, item features are used to suggest other items that match the user's preferences, based on past actions or user feedback.

3.3 Noise in Recommendation system

The inconsistencies introduced (intentionally or unintentionally) in the datasets of the recommender system (RS) is termed as noise. The noise in data will affect the accuracy and performance of the RS. Generally, the inconsistencies (noise) are of two types:

(a) Malicious noise: It is associated with noise intentionally introduced by an ex- ternal agent to bias recommender results.

(b) Natural noise: These are involuntarily introduced by users, and that could potentially affect the recommendation result.

Handling malicious noise is a well-developed research area but natural noise is a very recent topic of research and has received much less attention. The natural noise is produced by different sources: while the malicious noise is usually associated with user profiles that match certain patterns, the natural noise identification is more difficult because it tends to appear in several ways different from each other. Due to these reasons, techniques for processing both types of noise must be different.

With regards to natural noise, several authors have suggested that a rating never should be appreciated as a ground-truth value, because the process of eliciting preferences is intrinsically noisy. With this perspective, Amatriain et al. and Pham and Jung outlined two possible reasons for the appearance of natural noise in recommendation datasets: 1. the user preferences change over time. 2. the user's inherent imprecision for eliciting ratings.

3.4 Managing Natural Noise

The major focus of a recommender system is to improve the prediction accuracy, while natural noise moderates the performance of the system. The ground assumption of the user-item rating matrix is often violated due to inconsistent user behavior in providing ratings on similar items. There are two main categories of the noise correction methods:

(a) Classification based method: It classify the ratings into different categories (e.g in 3 classes or 5 classes) according to the boundary/threshold between different pair of classes.

(b) Fuzzy based method: It uses a fuzzy set to further describe to what degree that each rating value belongs to a category (i.e., low, medium, or high rating), which is more reasonable than the classification-based methods. However, the membership functions in the fuzzy-based schemes may cause different rating values having the same fuzzy profiles. After detecting noisy ratings, some methods remove the noising ratings directly, and some methods correct noisy ratings with average rating values. However, these methods may lead to some new problems, such as data sparsity and correcting noise insufficiently.

4. Methodology

The approach proposed in this report classifies the users and items into 5 different classes. Users are classified in to strongly-negative, negative, average, positive and strongly-positive classes and in the same way items are classified into strongly-not- preferred, not-preferred, average-preferred, preferred and strongly-preferred. The threshold values for classification are decided by employing statistical parameters mean (μ) and standard deviation (σ):

(a) k1 :- between strongly-negative and negative = $\mu - x1\sigma$

(b) k2 :- between negative and average = $\mu - x2\sigma$

(c) v1 :- between average and positive = $\mu + x3\ \sigma$

(d) v2 :- between positive and strongly-positive = $\mu + x4\ \sigma$

The threshold values are not taken as constant for all the users and items rather it is calculated using proposed statistical formula for each item and users separately to carry out the experiment. The values of x1, x2, x3 and x4 are determined by experimenting with various values but these are constant for all.

Initially, the users are separated based on the threshold boundaries into 5 classes. Then, the cardinality of rating metrics is used to classify ratings in to different classes. Those users which does not fall in any of the categories are put in the variable class (means the behavior of these users can't be determined). Also, the items are separated based on the threshold boundaries into 5 classes. Then, the

cardinality of rating metrics is used to classify ratings in to different classes. Those items which does not fall in any of the categories are put in the variable class (means the behavior of these items can't be determined).

4.1 Correction of Natural Noise

Once the users and items are classified in to different classes. The natural noise is corrected based on the proposed algorithm. The detailed procedure of the noise rating correction is presented in Figure.

User classes→ Item classes↓	U_{sn}	U_n	U_a	U_p	U_{sp}
I_{snp}	Modify user rating with k_1	No modification	No modification	No modification	No modification
I_{np}	No modification	Modify user rating with $(k_1+k_2)/2$	No modification	No modification	No modification
I_{ap}	No modification	No modification	Modify user rating with $(k_2+v_1)/2$	No modification	No modification
I_p	No modification	No modification	No modification	Modify user rating with $(v_1+v_2)/2$	No modification
I_{sp}	No modification	No modification	No modification	No modification	Modify user rating with v_2

5. Results and Comparisons

This shows the rating c ying the proposed re-classification method. Furthermore, the performance of the proposed algorithm is compared with relevant approaches. Out of 943 users, 332 users get classified into strongly-positive class, 18 users get classified into positive class, 48 into average class, 11 into negative class, 1 into strongly- negative class and 533 users are of variable behavior, we can't determine any class for them. The user distribution among various classes is shown in Figure 4.1.

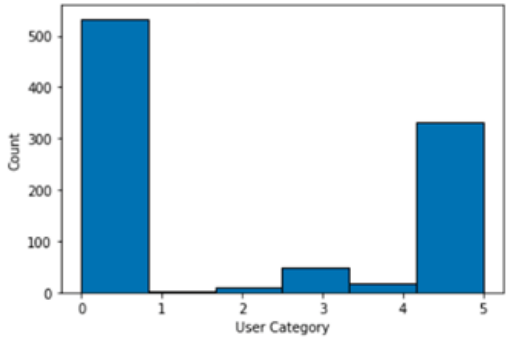


Fig 3. User Category distribution

Out of 1032 items, 661 are of variable class, 321 of average-preferred class, 21 of preferred class and 29 of strongly-preferred class. The distribution of different items i.e., books, movies etc. among various classes is shown in Figure 4. In the S. Bag and Tiwari paper, the users and items are classified in to 3 classes. More users and items are got classified in one of the categories in this case but the performance of the recommender system is less in comparison to the algorithm. 880 inconsistencies are detected in the dataset and corrected according to the proposed method.

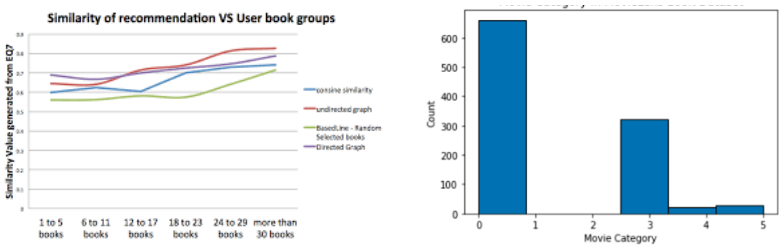
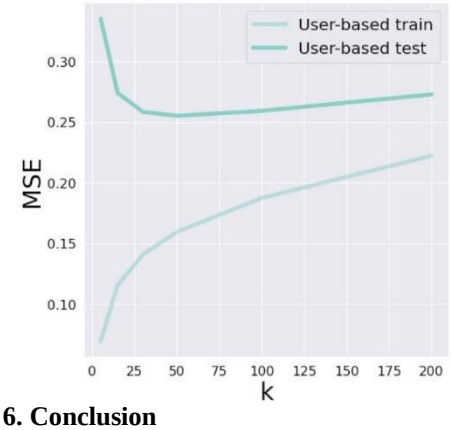
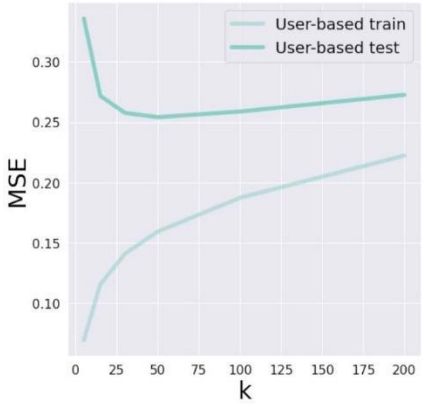
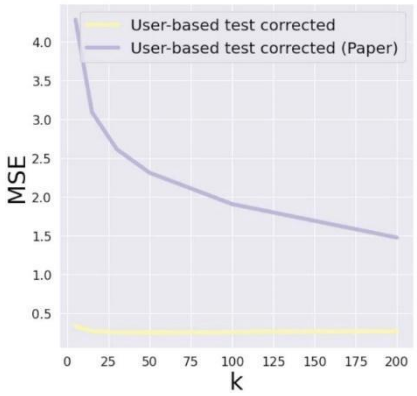


Fig 4. Item Category distribution

The corrected and original rating matrix is used with top-k collaborative filtering approach to verify the improvement in recommender system after correcting natural noise and I got the significant improvement in the performance. The plot of Mean square error (MSE) for different value of k is shown in figure. I also implemented the code for the algorithm proposed in S. Bag and Tiwari paper for the same dataset to check the improvements of newly designed approach and I got the significant difference in MSE of both approaches. Both the approach is also compared for k = 50 for exact value of MSE and the outcome is shown in figures.







6. Conclusion

The basic objective of this work is to improve the efficiency of recommendation generation in noisy and sparse scenarios. All the previous algorithms based on classification approach proposed by various researchers divided the users and items in to 3 classes while this thesis proposes a new way of classification and threshold parameters calculation. The users and items are classified into 5 categories and the threshold parameters are calculated for each and every user and item using statistical parameter instead of taking it as constant as proposed in previous literature. This is a key selling point of this algorithm.

The hesitating behavior, where user oscillates between negative, average and positive classes, and does not fall into a specific one is considered. The intuition that I got by doing this is that as the number of classes increases, the probability of hesitating behavior minimizes but at the same time the number of users and items going into the variable class increases. So, this is a scope of improvement in this thesis. The cardinality principle can be looked to be modified so that we can reduce the variable users. But even after this behavior, we can clearly see the significant improvement in the Recommender system.

The work can be extended in future to study the variable number of classification classes for users and items based on the shape and behavior of dataset.

7. References

1. D. Cosley, S.K. Lam, I. A. J. K. J. R.: 2003, Is seeing believing? how recommender system interfaces affect users' opinions.
2. J. Castro, R. Y. and Martínez, L.: 2018, A fuzzy approach for natural noise management in group recommender systems, Expert System with Applications.
3. J.M. Pujol, N. T. and Oliver, N.: 2009, Rate it again: Increasing recommendation accuracy by user re-rating ACM Conference on Recommender Systems.
4. Pengyu Wang, Yong Wang, L. Y. Z. and Zhu, H.: 2020, An effective and efficient fuzzy approach for managing natural noise in recommender systems, Information Sciences.
5. Pham, H. and Jung, J.: 2013, Preference-based user rating correction process for interactive recommendation systems Multimedia Tools Applications.
6. R. Yera, J. C. and Martínez, L.: 2016, A fuzzy model for managing natural noise in recommender systems, Applied Soft Computing.
7. R. Yera, J. C. and Martínez, L.: 2020, Natural noise management in recommender systems using fuzzy tools, Computational Intelligence for Semantic Knowledge Management.
8. R.Y. Toledo, Y. M. and Martínez, L.: 2015, Correcting noisy ratings in collaborative recommender systems, Knowledge Based Systems.

9. S. Bag, S. Kumar, A. A. and Tiwari, M.: 2019, A noise correction-based approach to support a recommender system in a highly sparse rating environment, Decision Support Systems.
10. C. De Maio et al. RSS-based e-learning recommendations exploiting fuzzy FCA for knowledge modeling Appl. Soft Comput.
11. J. Noguera et al.A mobile 3D-GIS hybrid recommender system for tourism Inf. Sci.
12. S. Schiaffino et al.Building an expert travel agent as a software agent Expert Syst. Appl.
13. G. Castellano et al. Newer: a system for neuro-fuzzy web recommendation Appl. Soft Comput.
14. P.H. Abreu et al. Improving a simulated soccer team's performance through a memory-based collaborative filtering approach Appl. Soft Comput.
15. P. Sánchez et al. A fuzzy model to evaluate the suitability of installing an ERP system. Inf. Sci.
16. M.G. Vozalis et al. Using SVD and demographic data for the enhancement of generalized collaborative filtering Inf. Sci.
17. J. Bobadilla et al. Recommender systems survey Knowl.-Based Syst.
18. V. López et al. An insight into classification with imbalanced data: empirical results and current trends on using data intrinsic characteristics Inf. Sci.
19. L. Zadeh Fuzzy sets Inf. Control
20. L. Martínez et al. An overview on the 2-tuple linguistic model for computing with words in decision making: extensions, applications and challenges Inf. Sci.