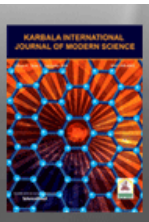

Karbala International Journal of Modern Science

Manuscript 3393

Classifying Items with the Rating Values 3 Using Text Reviews to Improve the Recommendation Accuracy in the Collaborative Filtering Approach

Ali Mohsin Ahmed Al-Sabaawi

Mohsin Hasan Hussein

Muyassar Dalli

Classifying Items with the Rating Values 3 Using Text Reviews to Improve the Recommendation Accuracy in the Collaborative Filtering Approach

Abstract

Collaborative filtering is a common aspect recently used in e-commerce to increase sales and overcome information overload. One significant limitation in collaborative filtering is data sparseness. Several studies have proposed alleviating this issue by utilizing extra information such as users' reviews. However, the researchers have been concerned with using entire reviews irrespective of the users' ratings. This requires extra processing time and might perplex the recommendation decision. In this study, after analyzing the users' ratings and reviews, it was noted that when the rating values are 4 or 5, most of the reviews accompanying these ratings are positive. Otherwise, when the ratings are 1 or 2, the reviews are negative. The rating value 3, however, is a confusing one. It, consequently, might mislead users' preferences. Thus, two points are highlighted. First, analyzing and classifying the entire users' reviews is time-consuming. Second, directly considering the rating value 3 as a like or dislike decreases the model's accuracy. Hence, this method utilizes the users' reviews only in case the rating value is 3. Three scenarios were formulated and fitted to this case. The first scenario considers the rating value 3 as a like. The second scenario regards the same value as a dislike. The third one classifies the users' reviews into positive or negative regarding their content. Additionally, a probabilistic method was applied to compute the recommendation list. The proposed method was implemented on two standard datasets. Eventually, the results of the third scenario outperformed the other two in terms of accuracy.

Keywords

collaborative filtering, text mining, data sparseness, classification

RESEARCH PAPER

Classifying Items With the Rating Values 3 Using Text Reviews to Improve the Recommendation Accuracy in the Collaborative Filtering Approach

Ali M.A. Al-Sabaawi [a], Mohsin H. Hussein [b,c,*], Muyassar Dalli [a]

[a] Software Department, College of Information Technology, Ninevah University, Mosul, Iraq
[b] College of Computer Science and Information Technology, University of Kerbala, Karbala, Iraq
[c] College of Science, University of Warith Al-Anbiyaa, Karbala, Iraq

Abstract

Collaborative filtering is a common aspect recently used in e-commerce to increase sales and overcome information overload. One significant limitation in collaborative filtering is data sparseness. Several studies have proposed alleviating this issue by utilizing extra information such as users' reviews. However, the researchers have been concerned with using entire reviews irrespective of the users' ratings. This requires extra processing time and might perplex the recommendation decision. In this study, after analyzing the users' ratings and reviews, it was noted that when the rating values are 4 or 5, most of the reviews accompanying these ratings are positive. Otherwise, when the ratings are 1 or 2, the reviews are negative. The rating value 3, however, is a confusing one. It, consequently, might mislead users' preferences. Thus, two points are highlighted. First, analyzing and classifying the entire users' reviews is time-consuming. Second, directly considering the rating value 3 as a like or dislike decreases the model's accuracy. Hence, this method utilizes the users' reviews only in case the rating value is 3. Three scenarios were formulated and fitted to this case. The first scenario considers the rating value 3 as a like. The second scenario regards the same value as a dislike. The third one classifies the users' reviews into positive or negative regarding their content. Additionally, a probabilistic method was applied to compute the recommendation list. The proposed method was implemented on two standard datasets. Eventually, the results of the third scenario outperformed the other two in terms of accuracy.

Keywords: Collaborative filtering, Text mining, Data sparseness, Classification

1. Introduction

Collaborative filtering (CF) establishes personalized recommendations by tracing user behavior to determine what the users like or dislike [1–3]. The first task of CF is to identify the users who have similar tastes by checking their ratings [4,5]. The second task is to compute the similarity among users on the basis of their ratings by applying one of the similarity formulas such as Pearson's correlation coefficient (PCC), cosine measure, or Jaccard's coefficient. Finally, according to the similarities among users, the model predicts from the nearest similarity values of users to the targeted user set of items that he/she has not rated yet [6,7]. Thus far, CF has been the most successful approach for creating recommendation system models [8]. Although CF has succeeded in providing an acceptable list of items favorable to users, there are still several noteworthy obstacles in real-world situations. A major challenge is the data sparseness issue. It occurs when the existent rating is less than the total number of items in the rating matrix. It is difficult for CF to make suggestions for users because there are several missing ratings in the rating matrix [9,10]. To alleviate the aforementioned problem and enhance the intelligibility of CF, textual reviews have been utilized. Recently,

Received 25 September 2024; revised 24 December 2024; accepted 29 December 2024.
Available online 18 January 2025

* Corresponding author at: College of Computer Science and Information Technology, University of Kerbala, Karbala, Iraq.
E-mail address: mohsin.h@uokerbala.edu.iq (M.H. Hussein).

https://doi.org/10.33640/2405-609X.3393

most online websites such as Flipkart and Amazon have been providing their users with the option of adding text reviews in addition to assigning a numerical rating for their products [11]. Usually, when customers intend to buy a specific product online, they read (0–10) online reviews to decide whether to buy it or not. Online reviews aid customers in judging the quality of items before purchase. Moreover, they make consumers aware of the items even if they do not intend to buy them immediately [11]. Rich information from the text reviews may partially reflect the characteristics and qualities of the products. Exploiting such characteristics can supply suggestions to other users and enhance the recommendation accuracy [10]. In compliance with some recent studies, utilizing the review text in CF can improve the explanatory capacity and lessen the impact of the sparsity issue [12,13]. Most existing models use text reviews to assist the model of interaction between the latent vector of users and items [14–19]. Despite the remarkable progress achieved by the previous models, they have several drawbacks with respect to the architecture of the attention mechanism. They extract the characteristics of all users' reviews to combine the rating values and the text reviews. Note that interpretations of rating scales can vary depending on the user preferences towards a specific product. Usually, if the rating scale is 1–5, 1 and 2 refer to disliked items whereas 4 and 5 indicate liked ones. However, the rating value 3 is a confusing one. Therefore, it is perplexing for the model regarding whether to consider items with the rating values 3 in the like item list or the dislike one. Consequently, it may mislead the model in the recommendation process.

In the literature, researchers might not universally agree on whether a rating of 3 is definitively positive or negative. Yet, they provide insights into how it might be perceived or used in different contexts.

In this study, text reviews were utilized in addition to the rating values to mitigate the sparsity problem. Moreover, it is believed that when users rate an item 4/5, they strongly mean that they like a product. If, on the contrary, their rating is 1/2, they definitely mean that they dislike it. Therefore, it is futile and time-consuming to analyze and classify the reviews when the rating values are (1, 2, 4, or 5). Unlike other studies, only when the rating value is 3, the classification of the text review is activated, and then, the decision of like or dislike is settled on the basis of the text review. If the review is positive, it represents a like. Otherwise, it is negative (dislike). At the beginning, the rating values are converted into 0 or 1 when the ratings were 4, 5, 1, or 2. Later, the text review is classified into positive or negative by using the polarity term level which extracted the text by using the opinion lexicon technique. Regarding the rating value 3, it is classified into 0/1 on the basis of the review classification result. Moreover, the probabilistic technique is used to find the top-n items. Eventually, the precision and recall evaluation measures are used to assess the performance of the proposed method.

The contributions of this study could be summarized as follows:

- Only the user reviews of items whose rating value is 3 are classified to utilize them in the recommendation process.
- The user reviews of items rated 1, 2, 4, and 5 are ignored.
- A comparison is carried out by establishing the other two scenarios that contemplated the rating value 3 as a like and dislike, respectively.
- The evaluation process is carried out by applying the proposed model to two different datasets.

The rest of this paper includes the related works discussed in Section 2. The proposed method is presented in Section 3. In Section 4, the results are discussed. Finally, the conclusions are drawn in the last section.

2. Related work

The most common technique in recommendation systems is CF. One of the major challenges in this technique is data sparsity. Recently, several studies have been dedicated to diminishing this problem [20]. Some of these studies used additional data sources such as social relations and text reviews [10,21]. Some platforms provide users with the ability to establish their relationships. Thus, researchers exploit these relations to improve the prediction and recommendation performance.

Reference [22] proposed the explicit–implicit social recommendation (EISR). The EISR method utilized three information sources incorporated into the PMF method. However, this study did not exploit the implicit rating information. This limitation was resolved in Ref. [12], which integrated four sources of data, namely explicit and implicit social relations as well as explicit and implicit users' feedback. The integration of explicit and implicit relations was achieved but in a different scenario in Ref. [23]. The authors applied the weighted voting technique, which is generally utilized to determine the neighbors' adequate ratings.

Another vital source of data is text reviews. In the past few years, many studies have been conducted to exploit user reviews to improve recommendation accuracy [24–27]. In Ref. [24], the authors used manually designed ontologies to create free texts. The time-consuming domain dependence was the principal problem of this study. In Ref. [25], a latent topic and latent factors were combined to create better vectors of users and items. Another proposed study [26] produced a unified framework to assess both aspect evaluations and the mood of the connected user for more individualized item recommendations. The user text review was analyzed and classified via the lexicon technique to figure out the user features and integrate these features into the tensor factorization method. In Ref. [27], the researchers incorporated user ratings and user reviews and produced an adjustment weight to tune the user text review towards any positive rating. Recently, deep learning has been employed in various aspects such as image processing [28–30] and recommendation systems [20]. Deep learning in text mining was utilized to improve the accuracy of the recommendations [20]. This study involved two steps. The first step was to use multichannel deep learning to extract aspects and generate ratings by computing user polarities. In the second step, tensor factorization was used to compute the predictions.

After reviewing the previous studies, it can be concluded that there is still a serious deficiency in terms of recommendation accuracy. The drawback of these studies is that they adopted user text reviews of all users and they integrated the whole ratings with the whole reviews. This not only perplexed the CF technique but is also time-consuming. To overcome the aforementioned problems and enhance the recommendation accuracy, the proposed method analyzed and classified the user reviews of items with the rating value 3 only.

3. The proposed method

Two objectives are achieved in this study. First, it improves the recommendation quality by integrating users' ratings and reviews of users whose rating value for products is 3. Second, it reduces computational complexity. In addition, utilizing the reviews depended on the rating values. From the experiments, it is observed that the user reviews in most cases were positive when the rating values were 4 or 5. However, the reviews are negative when the rating values are 1 or 2. For example, in the Yelp dataset, after randomly selecting 6863 users whose ratings were 4 or 5, 6648 were positive reviews, while the rest were negative ones. Similarly, from 1676 users whose ratings were 1 or 2, 1590 were negative reviews whereas the rest were positive. In contrast, the users' positive and negative reviews are close when their rating value is 3. The same thing is observed in the Amazon musical instruments dataset.

The fundamental idea of this study was to ignore the user reviews when the user ratings were 1, 2, 4, or 5. It is believed that it is superfluous and time-consuming to analyze the text review and determine whether it was positive or negative when the rating value was not 3. It is noted that there is a semi-matching between the reviews and the ratings (1/2/4/5). Thus, to achieve the aforementioned objective, we propose the method displayed in Fig. 1 below.

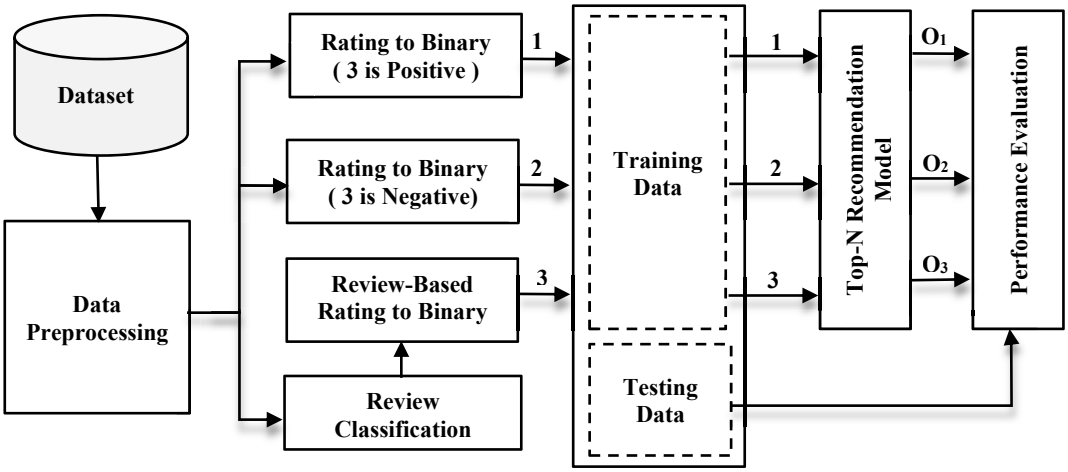

Fig. 1. Proposed method.

3.1. Pre-processing

Two standard datasets were used in this study. The following procedures were applied at this step:

a) Because of memory size, the users were randomly filtered according to their ratings.
b) Duplicated data was removed.
c) Items with less than two ratings were removed. It was unnecessary to use such items as they were considered cold-start items and were out of the scope of this study.
d) Text cleaning: This process involved extracting the raw text from the input data and converting it into the desired encoding format by eliminating all non-textual elements such as markups and metadata.
e) Text pre-processing: This involved several tasks including tokenization, stop word removal, lowercasing, stemming, and lemmatization.

3.2. Exploiting the data resources

Two resources were utilized in this study, namely, user ratings and user reviews. For the user reviews, the polarity term density technique was used. The objective of analyzing the sentiment-laden text was to assort words into either positive or negative in a particular text, which was measured using the polarity term density in the sentiment analysis. The density of these divisive terms served as a basis for these statistics, which helped to identify how strongly a text expressed a mood. The polarity term was extracted using a sentiment lexicon.

With this technique, the ratio of the document polarity level was computed. This ratio was calculated by counting the positive and negative terms and then finding the difference between them. Next, they were divided over the length of the text to select the maximum polarity level value. On the basis of this technique, the user reviews were classified into positive or negative (0/1) using sentiment analysis. After calculating the positive and negative polarity densities, the sentiment score (overall polarity) value is calculated and used to determine whether a review is positive or negative. In this study, the sentiment score threshold value is 0.5 and 0.3 for Yelp and Amazon datasets respectively. The sentiment score was determined by trial and error to choose the best score in terms of accuracy.

Additionally, the rating values were converted into 0 or 1. For converting ratings into 0 or 1, we followed three scenarios. The equations below show the process.

First scenario:

$$r = \begin{cases} 1, & r \geq 3 \\ 0, & otherwise \end{cases} \quad (1)$$

Second scenario:

$$r = \begin{cases} 0, & r \geq 3 \\ 1, & otherwise \end{cases} \quad (2)$$

Third scenario:

$$r = \begin{cases} 1, & r > 3 \\ 0, & r < 3 \\ based\ on\ review, & r = 3 \end{cases} \quad (3)$$

Where r indicates the rating value.

3.3. Dividing the datasets

Usually, in any machine learning model, datasets are divided into two parts: training and testing. In this study, two training sizes were used, namely, 80 % and 75 % of the entire rating for each user, whereas the rest was the test part.

3.4. Computing top-n items

In this study, a probabilistic method was applied to compute the top-n items of the target user. First, the co-occurrence of each item was computed on the basis of users who rated the same items. In other words, for each pair of items, the number of users who rated this pair was computed. At the end of this step, an array size number of items × number of items was created. Second, the probability of each item was estimated by dividing the occurrence value of each item over the total value of the entire occurrence values of the created array (item × item). Equation (4) depicts the process. In this step, 0 and 1 refer to the probability of an item.

$$partial\ prob = a/b \quad (4)$$

Where a indicates the occurrence of each item. While b refers to the total occurrence of entire items.

Thereafter, for the target user, the final probability was computed by multiplying the partial probability of each user rating a specific item as shown in Equation (5).

$$final\ prob = \prod_i^n partial\ prob_i \quad (5)$$

Where n denotes the number of users who rated a specific item.

The model produced three different outputs on the basis of the input scenarios. Thereafter, the top-n

probabilities were recommended to the target user. Then, the model was evaluated to assess the top-n recommendation items by using the precision and recall metrics.

4. Experimental study

This section is dedicated to answering the following research questions: 1) What is the performance of using the review items with a rating value of 3? 2) What is the performance of ignoring the user reviews when the corresponding ratings are 1, 2, 4, and 5? 3) What is the performance of considering the items with a rating value of 3 as likes?

This section is devoted to evaluating the performance of the proposed method. The dataset is described and then the metrics are displayed in the following subsection. The results and discussion are presented in the last subsection.

4.1. Dataset preparation

Two standard datasets were used in this study, namely Yelp and Amazon musical instruments. Yelp is an online platform for user reviews of local businesses (approximately 200,000 businesses). It is generally used in different academic areas such as natural language processing, machine learning, and recommendation systems. To avoid memory consumption, we kept the users who had at least four ratings.

Amazon Musical Instrument (MI) involves a collection of products available on Amazon for musical instruments. It includes a set of information such as brand, price, and category. It also incorporates information that is exploited in academic research such as user reviews and user ratings, which can be used in recommendation systems. In this study, only users with eight ratings were considered (see Table 1).

4.2. Evaluation metrics

Two metrics were used in this study: precision and recall. These metrics are commonly used in CF to evaluate the recommendation accuracy. They compute the ratio of the relevant items by utilizing the confusion matrix as follows:

Table 1. Datasets statistics.

Dataset	MI	Yelp
# Users	1428	6324
# Items	899	4116
# Reviews	10261	10001
Sparsity	99.2 %	99.96 %

$$precision = \frac{number\ of\ relevant\ items}{length\ of\ recommendation} \tag{6}$$

$$recall = \frac{number\ of\ relevant\ items}{entire\ relevant\ items} \tag{7}$$

4.3. Results and discussion

The datasets were divided into training and testing parts: 80 % of the entire rating of each user was randomly dedicated to the training part and the rest to the testing part. In this study, 5-fold and 4-fold cross-validation techniques were used to validate the model. The backbone of the proposed method is that the reviewing text of the rating values 4 and 5 is in most cases positive. It is negative, though, when the rating values are 1 and 2. Tables 2 and 3 show the details of this contention.

The experiment was implemented in three scenarios. In the first scenario, we assumed that the rating value 3 was a like along with the rating values 4 and 5. In the second, it was presumed that the rating value 3 was a dislike along with the rating values 1 and 2. The third option was to evaluate the

Table 2. Statistics of positive and negative reviews in Amazon MI.

No. of all 4 or 5 ratings	4 or 5 positives	4 or 5 negatives
9022	8804	218
No. of all 1 or 2 ratings	1 or 2 positives	1 or 2 negatives
467	76	391
No. of all 3 ratings	3 positive	3 negative
772	439	333

Table 3. Statistics of positive and negative reviews in Yelp.

No. of all 4 or 5 ratings	4 or 5 positives	4 or 5 negatives
6863	6648	215
No. of all 1 or 2 ratings	1 or 2 positives	1 or 2 negatives
1676	86	1590
No. of all 3 ratings	3 positive	3 negative
1461	879	582

Table 4. Recommendation performance for the Yelp dataset 80 % training.

	Metrics	@5	@10	@20	@30
3 is dislike	Precision	96.12403	92.4242	83.8461	80.3921
	Recall	19.2579	22.3034	22.4637	22.7452
3 is like	Precision	99.2207	97.9434	94.9748	93.1707
	Recall	45.1764	46.371	46.5732	47.3841
3 based on the review	Precision	**99.6835**	**99.2709**	**98.6825**	**96.5**
	Recall	**53.4265**	**53.5269**	**54.0311**	**54.3509**

The boldface indicates best results of each column.

Table 5. Recommendation performance for the Amazon MI dataset 80 % training.

	Metrics	@5	@10	@20	@30
3 is dislike	Precision	81.4388	79.2401	77.9898	76.5492
	Recall	49.6491	51.1383	55.2252	55.8510
3 is like	Precision	84.2201	81.7872	79.1825	77.3132
	Recall	72.2118	74.2524	74.8547	76.3485
3 based on	Precision	**84.77**	**82.1065**	**79.7661**	**77.3356**
the review	Recall	**74.2452**	**77.5**	**79.203**	**81.6924**

The boldface indicates best results of each column.

Table 6. Recommendation performance for the Yelp dataset 75 % training.

	Metrics	@5	@10	@20	@30
3 is dislike	Precision	95.3051	91.4242	82.1803	80.4332
	Recall	10.0909	21.0909	21.5302	22.1452
3 is like	Precision	99.4767	98.5530	96.1244	93.8635
	Recall	43.2034	44.3158	45.1039	45.6721
3 based	Precision	**99.8457**	**99.7721**	**98.5667**	**97.2133**
on review	Recall	**44.7183**	**45.8822**	**45.8999**	**47.4256**

The boldface indicates best results of each column.

Table 7. Recommendation performance for the Amazon MI dataset 75 % training.

	Metrics	@5	@10	@20	@30
3 is dislike	Precision	79.4069	77.5641	76.8939	74.9627
	Recall	48.6869	48.4	51.5930	52.7660
3 is like	Precision	82.7809	82.4218	81.3166	77.6860
	Recall	**72.1130**	74.2212	**74.8493**	75.9535
3 based	Precision	**84.2105**	**83.4841**	**81.7427**	**80.6387**
on review	Recall	71.2264	**74.4395**	75.4802	**76.4427**

The boldface indicates best results of each column.

rating value 3 on the basis of the classification of the review for that item. Tables 4 and 5, show the results of recommendation for 80 % training, whereas Tables 6 and 7 show the results for 75 % training for top-5, 10, 20, and 30. The tables depict the precision and recall results for the aforesaid scenarios.

Note that there was an obvious superiority when we utilized the classification of the user review for the rating value 3 in all of the datasets in the precision metric. The precision values for the last scenario were better than the other scenarios in all of the recommended items (top 5, 10, 20, and 30). An additional point was that the precision values decreased when the list of recommendation items increased. For example, in Table 4, when the recommendation is top-5, the precision value is 99.68. It then starts to decrease gradually until it reaches the minimum value (96.5) at top-30. The same situation appears in Table 5. In contrast, the recall values increase when the length of the recommendation list does. According to the above results, when some users rate an item 3, this means that they like the product. Yet, when some other users give the same rate value, they dislike it. Therefore, review texts in these cases are essential to exactly recognize the user's opinion about the product.

In Tables 6 and 7, it can be seen that the precision values when the training size is 75 % are slightly better than the results in Tables 4 and 5. The reason for this is that the number of relevant items (testing part) increases and, consequently, the precision improves. On the opposite, the recall results in Tables 4 and 5 exceed the results of Tables 6 and 7. The justification of this preference is that when the training size is 75 %, it means the entire relevant items (in the testing part) are bigger than the relevant items compared with the training size in Tables 4 and 5. This causes a reduction in the results of the recall metric.

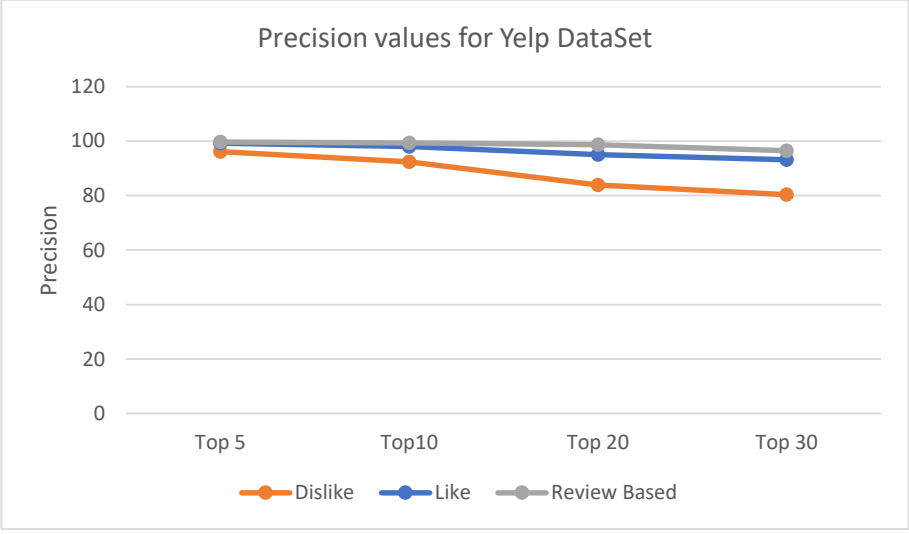

Fig. 2. Precision results for the Yelp dataset of 80 % training.

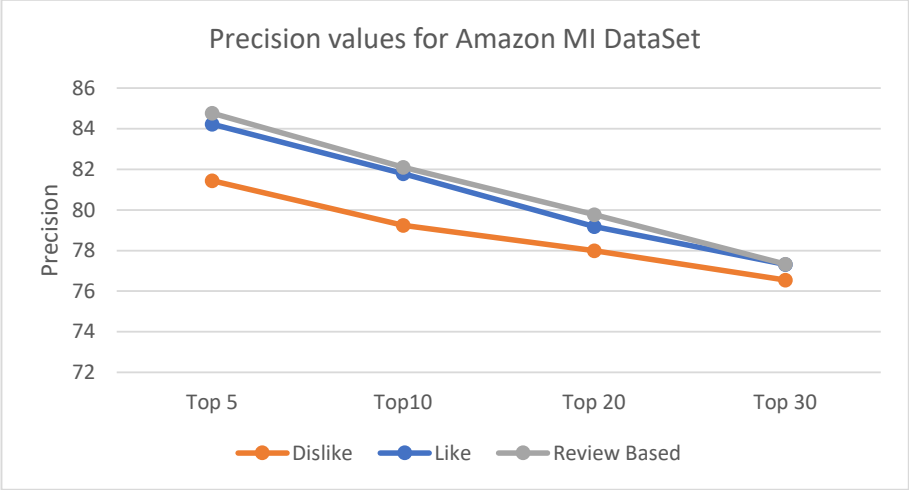

Fig. 3. Precision results for the Amazon MI dataset 80 % training.

Figs. 2 and 3 show the results of the precision metric in Yelp and Amazon MI, respectively. As seen in these figures, the worst results were recorded when directly considering the rating value 3 as a dislike. This means that most users who rated 3 liked the product. In addition, directly considering the rating value 3 as a like yields pretty results. However, the third scenario which utilized text reviews in the case of the rating value 3 produced the best results among all three considered scenarios. The same concept can be seen in Fig. 3. The figures confirm our idea that using user reviews only when the rating value is 3 leads to enhancing the recommendation accuracy and reducing the processing time.

Figs. 4 and 5 depict the results of the recall metric for the Yelp and Amazon MI datasets, respectively. In these figures, again, the outperformance of the third scenario is shown. Moreover, the results increased gradually when the recommended list increased. For example, the results are 53.42 and 54.35 in the review-based scenario when the list size is 5 and 30, respectively. The same concept was repeated in all of the other scenarios. Figs. 6 and 7 show the results of precision metrics when the training size is 75 %. It can be seen that the results of the first and third scenarios (rating value 3 as like and depending on the users' reviews respectively) are better than the results of the second scenario. Moreover, the results of the third scenario are again better than those of the first scenario, which supports our view. In other words, utilizing users' reviews when the rating value is 3 provides the best results compared with other procedures.

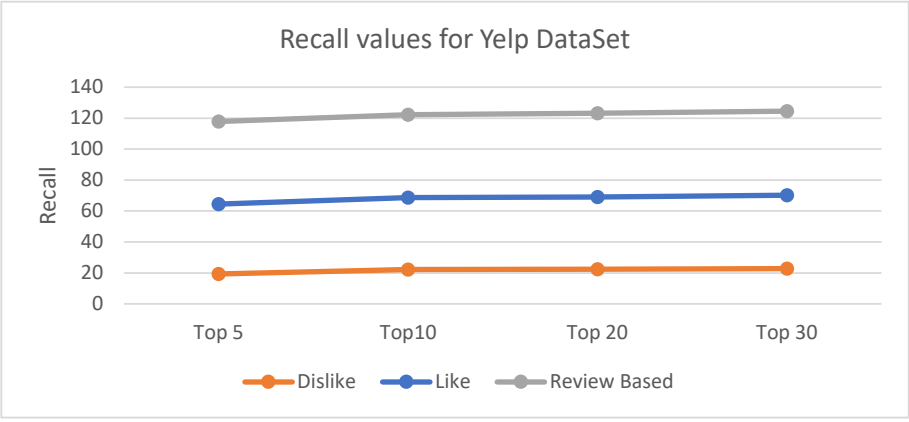

Fig. 4. Recall results for the Yelp dataset 80 % training.

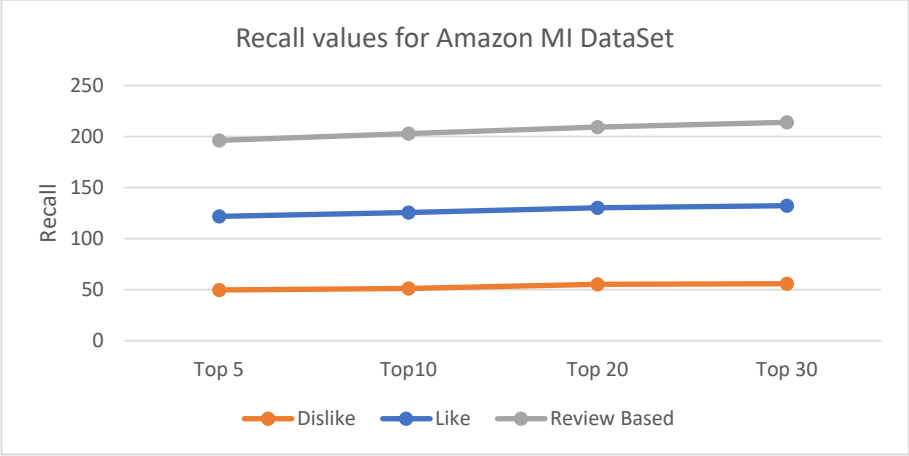

Fig. 5. Recall results for the Amazon MI dataset 80 % training.

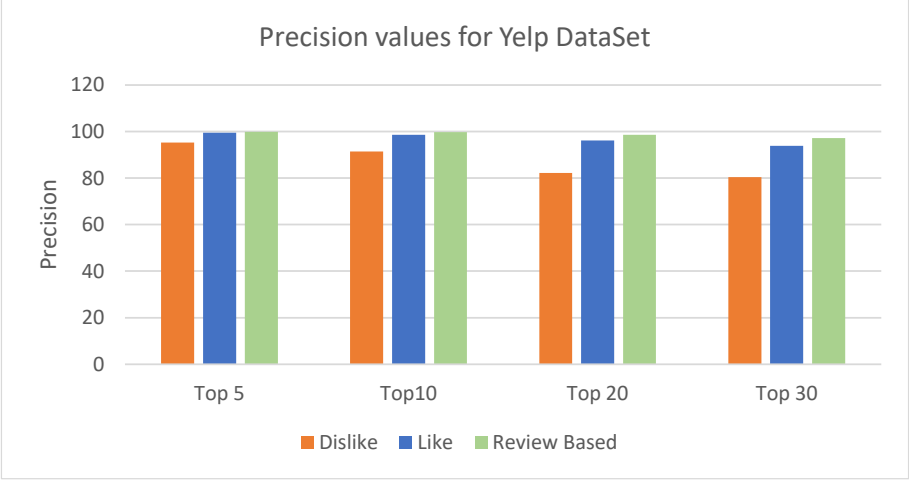

Fig. 6. Precision results for the Yelp dataset 75 % training.

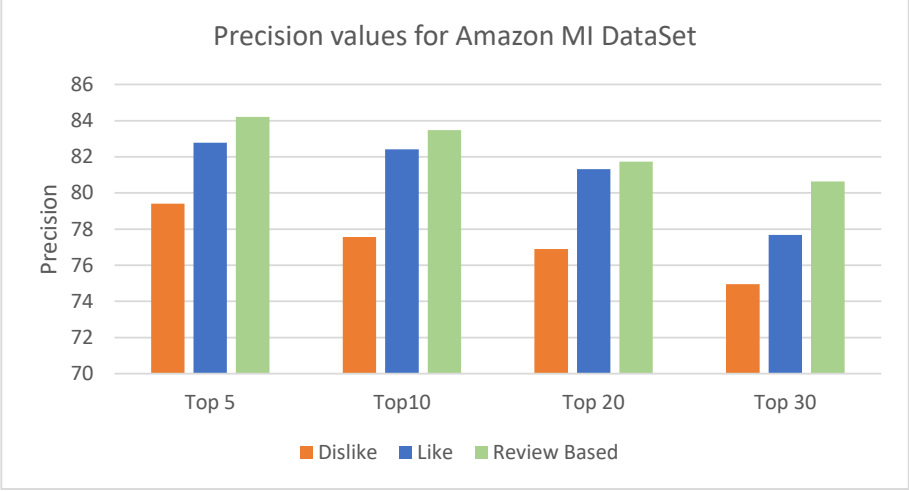

Fig. 7. Precision results for the Amazon MI dataset 75 % training.

5. Conclusions

Analyzing and understanding all user reviews is time-consuming. According to our experiment, most ratings (1, 2, 4, and 5) expressed user preferences clearly without utilizing the reviews accompanying these ratings. The only exception was items with the rating value 3, which was neither positive nor negative. Thus, user reviews were used to determine the user preferences only for items with the rating value 3. The study revealed that items with the rating value 3 did not precisely mean that the user liked or disliked the product. Moreover, the study diminished the processing time by analyzing and classifying the reviews of the items with the rating value 3 only. The study also recorded that the highest recommendation accuracy was achieved when the user opinions were extracted from these items.

The limitation of this work is that some reviews are ambiguous so it is hard to figure out and classify the users' preferences. Therefore, for a further future quest, new models such as deep learning techniques will expectedly have the ability to classify such reviews more precisely. Moreover, another factor can be added such as the purchasing time of a particular product. Finally, other metrics can be added to evaluate the proposed model.

Ethical information

The used datasets are public and free web access and dedicated for research studies and the manuscript does not contain any studies involving human participants or animals that require ethical approval.

Funding

No funds.

Conflicts of interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgements

The authors would like to thank the Presidencies of the University of Kerbala and Ninevah University for their Moral Support and for providing the laboratories to accomplish the practical part of the research.

References

[1] Y. Koren, R. Bell, Advances in collaborative filtering, in: F. Ricci, L. Rokach, B. Shapira, P.B. Kantor, eds., Recommender Systems Handbook, Springer, Boston, MA. 2011, pp. 145–186, https://doi.org/10.1007/978-0-387-85820-3_5.

[2] I.A. Al-Mani, A.M.A. Al-Sabaawi, M.H. Hussien, A review paper of model based collaborative filtering techniques, in: International Conference on Data Science and Intelligent Computing (ICDSIC), IEEE, Karbala, Iraq. 2022, pp. 52–57, https://doi.org/10.1109/ICDSIC56987.2022.10076148.

[3] F.T. Hussien, A.M.S. Rahma, H.B. AbdulWahab, Recommendation systems for e-commerce systems an overview, in: Journal of Physics: Conference Series, Volume 1897, Sixth International Scientific Conference for Iraqi Al Khwarizmi Society (FISCAS), IOP, Cairo, Egypt. 2021, pp. 1–14, https://doi.org/10.1088/1742-6596/1897/1/012024.

[4] D. Margaris, C. Vassilakis, Improving collaborative filtering's rating prediction quality by considering shifts in rating practices, in: Proceedings of the 19th IEEE Conference on Business Informatics (CBI 2017), IEEE, 2017, pp. 158–166, https://doi.org/10.1109/CBI.2017.24.

[5] W. Zhou, W. Han, Personalized recommendation via user preference matching, Inf. Process. Manag. 56 (2019) 955–968, https://doi.org/10.1016/j.ipm.2019.02.002.

[6] S. Ahmadian, M. Meghdadi, M. Afsharchi, A social recommendation method based on an adaptive neighbor selection mechanism, Inf. Process. Manag. 54 (2018) 707–725, https://doi.org/10.1016/j.ipm.2017.03.002.

[7] A.M.A. Al-Sabaawi, H. Karacan, Y. Yenice, A novel overlapping method to alleviate the cold-start problem in recommendation systems, Int. J. Software Eng. Knowl. Eng. 31 (2021) 1277–1297, https://doi.org/10.1142/S0218194021500418.

[8] M.K. Najafabadi, A. Mohamed, C.W. Onn, An impact of time and item influencer in collaborative filtering recommendations using graph-based model, Inf. Process. Manag. 56 (2019) 526–540, https://doi.org/10.1016/j.ipm.2018.12.007.

[9] A.M.A. Al-Sabaawi, H. Karacan, Y. Yenice, SVD++ and clustering approaches to alleviating the cold-start problem for recommendation systems, Int. J. Innov. Comput. Inf Control 17 (2021) 383–396, https://doi.org/10.24507/ijicic.17.02.383.

[10] G. Adomavicius, R. Sankaranarayanan, S. Sen, A. Tuzhilin, Incorporating contextual information in recommender systems using a multidimensional approach, ACM Trans. Inf. Syst. 23 (2005) 103–145, https://doi.org/10.1145/1055709.1055714.

[11] S. Saumya, P.K. Roy, J.P. Singh, Review helpfulness prediction on e-commerce websites: a comprehensive survey, Eng. Appl. Artif. Intell. 126 (2023) 107075, https://doi.org/10.1016/j.engappai.2023.107075.

[12] A.M.A. Al-Sabaawi, H. Karacan, Y. Yenice, Two models based on social relations and SVD++ method for recommendation system, Int. J. Interact. Mob. Technol. 15 (2021) 70–87, https://doi.org/10.3991/ijim.v15i01.17751.

[13] G. Ling, M.R. Lyu, I. King, Ratings meet reviews, a combined approach to recommend, in: Proceedings of the 8th ACM Conference on Recommender Systems, Association for Computing Machinery, Silicon Valley, CA, USA. 2014, pp. 105–112, https://doi.org/10.1145/2645710.2645728.

[14] J. McAuley, J. Leskovec, Hidden factors and hidden topics: understanding rating dimensions with review text, in: Proceedings of the 7th ACM Conference on Recommender Systems, Association for Computing Machinery, Hong Kong, China. 2013, pp. 165–172, https://doi.org/10.1145/2507157.2507163.

[15] W. Liu, Z. Lin, H. Zhu, J. Wang, A.K. Sangaiah, Attention-based adaptive memory network for recommendation with review and rating, IEEE Access 8 (2020) 113953–113966, https://doi.org/10.1109/ACCESS.2020.2997115.

[16] Z.Y. Khan, Z. Niu, A. Yousif, Joint deep recommendation model exploiting reviews and metadata information, Neurocomputing 402 (2020) 256–265, https://doi.org/10.1016/j.neucom.2020.03.075.

[17] R. Catherine, C. William, Transnets: learning to transform for recommendation, in: Proceedings of the Eleventh ACM

Conference on Recommender Systems, Association for Computing Machinery, Como, Italy. 2017, pp. 288–296, https://doi.org/10.1145/3109859.3109878.
[18] I. Islek, S.G. Oguducu, A hierarchical recommendation system for E-commerce using online user reviews, Electron. Commer. Res. Appl. 52 (2022) 101–131, https://doi.org/10.1016/j.elerap.2022.101131.
[19] H.Y. Lam, G.T.S. Ho, D.Y. Mo, V. Tang, Responsive pick face replenishment strategy for stock allocation to fulfil e-commerce order, Int. J. Prod. Econ. 264 (2023) 108976, https://doi.org/10.1016/j.ijpe.2023.108976.
[20] D. Aminu, S. Naomie, R. Idris, O. Akram, Recommendation system exploiting aspect-based opinion mining with deep learning method, Inf. Sci. 15 (2019) 1279–1292, https://doi.org/10.1016/j.ins.2019.10.038.
[21] H.J. Oudah, M.H. Hussein, Exploiting textual reviews for recommendation systems improvement, in: International Conference on Data Science and Intelligent Computing (ICDSIC), IEEE, Karbala, Iraq. 2022, pp. 70–74, https://doi.org/10.1109/ICDSIC56987.2022.10076074.
[22] W. Reafee, N. Salim, A. Khan, The power of implicit social relation in rating prediction of social recommender systems, PLoS One 11 (2016) 1–20, https://doi.org/10.1371/journal.pone.0154848.
[23] H.J. Oudah, M.H. Hussein, Exploiting social trust via weighted voting strategy for recommendation systems improvement, Karbala Int. J. Mod. Sci. 9 (2023) 226–236, https://doi.org/10.33640/2405-609X.3295.
[24] N. Jakob, S.H. Weber, M.C. Muler, I. Gurevych, Beyond the stars: exploiting free-text user reviews to improve the accuracy of movie recommendations, in: Proceedings of the 1st International CIKM Workshop on Topic-Sentiment Analysis for Mass Opinion (TSA '09), Association for Computing Machinery, New York, NY, USA. 2009, pp. 57–64, https://doi.org/10.1145/1651461.1651473.
[25] W. Zhang, J. Wang, Integrating topic and latent factors for scalable personalized review based rating prediction, IEEE Trans Knowl. Data Eng. 28 (2016) 3013–3027, https://doi.org/10.1109/TKDE.2016.2598740.
[26] Q. Diao, M. Qiu, C.-Y. Wu, A.J. Smola, J. Jiang, C. Wang, Jointly modeling aspects, ratings and sentiments for movie recommendation (JMARS), in: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD -14), Association for Computing Machinery, New York, NY, USA. 2014, pp. 193–202, https://doi.org/10.1145/2623330.2623758.
[27] P. Liu, S. Joty, H. Meng, Fine-grained opinion mining with recurrent neural networks and word embeddings, in: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP-2015), Association for Computational Linguistics. 2015, pp. 1433–1443, https://doi.org/10.18653/v1/D15-1168.
[28] X. Wu, D. Hong, J. Chanussot, UIU-net: U-net in U-net for infrared small object detection, in: IEEE Transactions on Image Processing 32, 2023, pp. 364–376, https://doi.org/10.1109/TIP.2022.3228497.
[29] X. Wu, D. Hong, J. Chanussot, Convolutional neural networks for multimodal remote sensing data classification, in: IEEE Transactions on Geoscience and Remote Sensing 60, 2022, pp. 1–10. http://doiorg/10.1109/TGRS.2021.3124913.
[30] D. Hong, B. Zhang, H. Li, Y. Li, J. Yao, C. Li, M. Werner, J. Chanussot, A. Zipf, X.X. Zhu, Cross-city matters: a multimodal remote sensing benchmark dataset for cross-city semantic segmentation using high-resolution domain adaptation networks, Remote Sens. Environ. 299 (2023) 113856, https://doi.org/10.1016/j.rse.2023.113856.