

SA-BTM-CF: A Sentiment-Aware Collaborative Filtering Framework for E-commerce Recommendation Systems

Mohammad Ali¹, Jin Xie², Wu Wenhuan³

^{1,2,3} School of Intelligent Connected Vehicle, Hubei University of Automotive Technology, China
Email: mohammadali881998@gmail.com, rudyxie@huat.edu.cn, wuwenhuan5@163.com

*Corresponding Author: Jin Xie

Abstract- The machine behind e-commerce recommendation systems is often based on Collaborative Filtering (CF) algorithms which are highly dependent on numerical star ratings ignoring the fact that user review text is a rich source of semantic information. This constraint increases the issue of data sparsity and lowers the quality of the recommendations. We suggest SA-BTM-CF, a Sentiment-Aware Collaborative Filtering system, which encompasses the Biterm Topic Model (BTM) with a conventional CF. The suggested methodology is a latent topic extractor of review literature and a fine-grained sentiment score generator of latent topic models which are employed to build an enriched user-item rating matrices. Personalized product suggestions are then created by computing user similarity on these sentiment-augmented ratings. Large scale shampoo review data on the JD.com site is experimented with and experimentally validates that SA-BTM-CF can be used to effectively reduce sparsity in data, and it can be used to obtain better recommendations than current baseline algorithms.

Keywords - collaborative filtering, recommendation algorithm, BTM model, text classification, e-commerce.

I. INTRODUCTION

Furthermore, since the Internet is experiencing a stage of high growth, and the amount of information has become a daunting problem to users who want to access meaningful information. Researchers in turn have come up with recommendation algorithms [1] to mitigate this problem. These algorithms include the content-based algorithms, collaborative filtering algorithms and the hybrid algorithms [2]. Traditional techniques use mainly the user rating analysis and the history to come up with recommendations. Nonetheless, when there are few user ratings of the items, the effectiveness of such approaches becomes low resulting in the errors of the recommendations that can be explained by the shortage of data [3].

The traditional recommendation algorithm considers similarity of users in terms of product ratings, which only represents the general levels of satisfaction, but not the disaggregated levels of attribute satisfaction. As an example, in reviews of a shampoo, user A and user B both rate a shampoo at 4 points. It is effective and expensive to user A, but it is reasonably priced and not effective to user B. They have different preferences even though they have the same ratings. Therefore, text reviews are more informative, and it is possible to analyze user preferences more accurately and increase the accuracy of recommendations.

In this work, the review text analysis with the use of the Biterm Topic Model (BTM) has been used to explore the feelings of users about various aspects of products. All products discuss various issues, and one can find the characteristics of products like price, efficacy, and scent. Holistic product rating prediction is achieved by dissecting topics followed by the algorithm predicting the most similar user to the target user when giving personalized recommendations in this manner, which aids in the process of determining the most similar user to the target user.

II. RELATED WORK

1. Recommendation Systems

The history of recommendation systems dates back to the early 1990s, the combination of the knowledge of various fields, including information retrieval, predictive analytics, data management, and market analytics [4],[5]. These systems, as invaluable tools, have grown more and more essential, as more and more users, more services, and data online have been added. They skillfully provide customized recommendations in advance, prior to the purchasing choices of users [6]. A useful recommendation system should address complexities like recommendation accuracy, cold start problems, computation and storage cost needed by large-scale datasets across the recommendation lifecycle [7].

Lee et al. [8] noted that recommendation systems can be more effective when some social connections and historical interactions are introduced in the recommendation algorithms. The unspoken forces of social networks like peer pressure and shared interests direct users with similar implicit regarding similar characteristics to receive similar suggestions. With the advancement of research, new paradigms such as association rule mining based recommendation, Bayesian classification based systems and personalized recommendation services have come up one after the other [9]. Personalized recommendation systems are critical in the context of e-commerce of improving the user experience and business growth. Collaborative filtering (CF) algorithm is one of the commonly used methods of recommendation and is highly popular nowadays owing to its capacity to utilize the data on user-item interactions. There are two broad methods of CF algorithms, namely user-based and item-based.

The algorithms of User-based Collaborative Filtering (UCF) work on the principle of finding certain users who share similar preferences or behavior patterns and providing them recommendations on the basis of the items that were liked by similar users. The assumption is that the past history tends to be concurring with the future [10]. UCF algorithms generally rely on calculating the similarity of pairs of users and do so with a range of different measures, some of which include cosine similarity or Pearson correlation coefficient [11]. This is then followed by the generation of recommendations, which is done by establishing the most similar users (nearest neighbors) and recommending items that these neighbors have rated highly or interacted with.

Compared to UCF, Item-based Collaborative Filtering (ICF) algorithms pay attention to determining similar items relying on the user interactions or rating [12]. The logic behind ICF is that people will gravitate towards the things that are close to what they have liked or even been in contact with in the past. Computing the similarity between items, ICF algorithms derive similarities between the items, which is usually done with the cosine similarity or adjusted cosine similarity [13]. Recommendations are produced by proposing items that are very similar with those that the user had shown interest in or rated positively in the past.

2. BTM Topic Model

The BTM, first introduced by Yan et al. [14] in 2013, provides an effective topic modeling framework that finds latent topics in short texts, e.g. product reviews or user comments. The BTM can effectively determine underlying

topics and their distributions in a corpus by modeling the patterns of co-occurrence of word pairs (biterms).

The combination of BTM with recommendation systems has been the focus of the great attention in the recent years in the area of information retrieval and the personalized recommendations. This is a new trend that is based upon the understanding that the traditional recommendation systems do not always pay due attention to the rich semantic information enclosed in the textual data and thus they are unable to offer accurate and relevant recommendations.

The work of Wang and colleagues [15] is one of the first ones on this topic that introduces a new method of collaborative filtering on the basis of the BTM. Through the topic modeling into the recommendation process, the suggested approach is able to capture the semantic relationship between items and users hence, achieving better recommendation accuracy. Based on this base, there has been recent work that has investigated the extensions and applications of the BTM in recommendation systems. As an example, Sun et al. [16] suggested a hybrid recommendation strategy that will be based on the combination of both BTM and matrix factorization methods to overcome the cold-start issue and promote the diversity of recommendations.

Based on this, it was also possible to develop a context-sensitive recommendation model through the combination of the BTM with data related to user-item interaction also, which offers customized recommendations taking into consideration both user preferences and contextual information. Furthermore, there have been attempts of using the BTM to use domain-specific recommendation tasks. Indicatively, Liu et al. [17] used the BTM to a healthcare recommendation system, in which text user feedbacks, as well as, medical records underwent analysis to determine individualized treatment recommendations.

III. RECOMMENDATION MODEL

1.Data Pre-processing

The product's user reviews were acquired and subjected to preprocessing through several steps. Initially, instances of repeated reviews by a single user within a brief timeframe were eliminated. Subsequently, reviews with fewer than five words were excluded to maintain recommendation precision due to their limited informational value. Lastly, irrelevant or superfluous words such as auxiliary and onomatopoeic terms, as well as function words, were filtered out from the reviews. The Chinese word segmentation tool "jieba"[18] was then employed to segment the text and generate a refined dataset

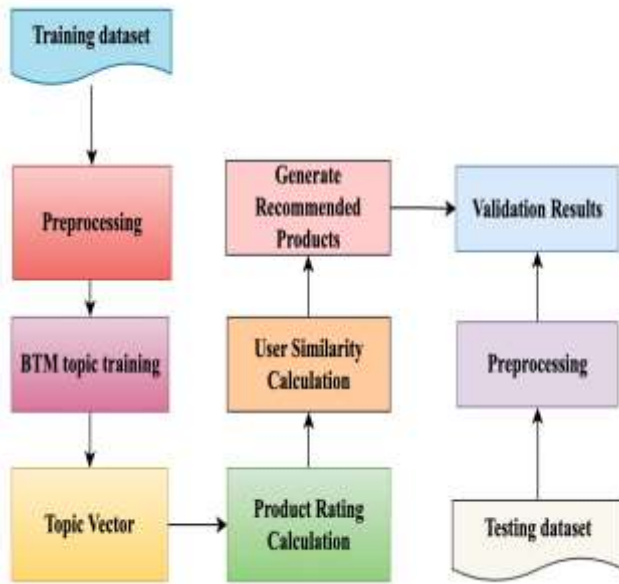


Fig. 1. The overall framework of BTM-CF.

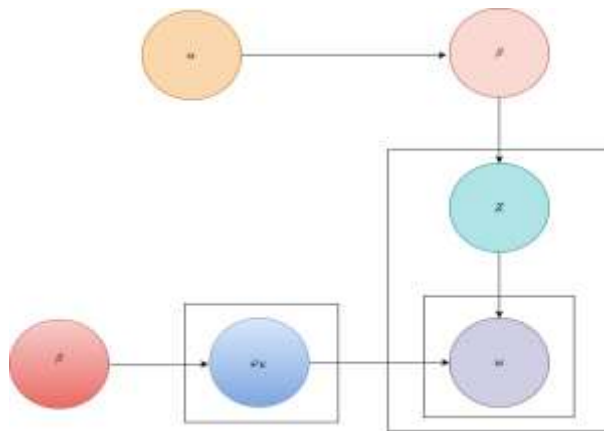


Fig. 2. BTM model training process framework

In the figure above, K is the number of artificial topics, θ is the topic distribution over the text corpus, ϕ is the term distribution over the topic, $\text{Mult}()$ is the multi-term distribution, and $\text{Dir}()$ is the Dirichlet distribution. Z is the Dirichlet distribution of the topic θ . BTM building steps are as follows:

2. Construct BTM Topic Model

Use The conventional approach to constructing a topic model primarily focuses on evaluating term significance within documents. However, when dealing with limited corpora, gauging term importance becomes challenging, leading to sparse data issues. Recognizing the limitations of LDA in modeling short texts, the BTM topic model emerged. The BTM model's concept revolves around capturing pairs of terms occurring together in the same context within the training dataset. The overall framework of BTM-CF is shown in fig. 1. Categorization of terms within this model is contingent upon their co-occurrence frequency. Additionally, the BTM model assumes a mixture distribution of multiple topic models to generate the entire training set, with each topic being generated by a global topic distribution. D represents the training set corpus, $W = w_1, w_2, \dots, w_N$ represents the biterm set obtained from the training set corpus, and w_1 represents a biterm unit $(q_{i,1}, q_{i,2})$. Assume that α and β are Dirichlet prior parameters. The whole process of model building using the BTM model is shown in Fig.2.

- Step 1: The word distribution $\phi K \sim \text{Dir}(\beta)$ under a topic is sampled from the parameter β by Dirichlet distribution;
- Step 2: Sampling the global topic distribution $\theta \sim \text{Dir}(\alpha)$ from the Dirichlet distribution of α ;
- Step 3: Select the topic Z from the common parameter θ of the short text corpus, and serve $Z \sim \text{Mult}(\theta)$;
- Step 4: Suppose that a word pair in the corpus is $b, b = (w_i, w_j)$. Take the words w_i, w_j from the topic Z extracted above, and make them obey $w_i, w_j \sim \text{Mult}(\phi Z)$;
- Step 5: Select the topic Z from the common parameter θ of the short text corpus, and serve $Z \sim \text{Mult}(\theta)$.

3. Product rating calculation

The Within the matrix of topics and vocabulary, only a subset of words associated with each topic exhibits polarity in sentiment. Utilizing the HowNet sentiment lexicon, the determination of positive, negative, and neutral sentiments for the words within each topic is facilitated. Upon identification of positive sentiment words within a topic, the sentiment score of the topic is incremented by one.

Conversely, the sentiment score of a topic decrease by one for each negative sentiment word detected. Neutral words do not

contribute to the sentiment score calculation. This process iterates until all words within topics are assessed, resulting in the derivation of the sentiment score corresponding to each topic. In the document-topic matrix, each document contains a different probability of each topic, and each document has a different rating. The rating $S_{ui,xh}$ of user u_i for commodity x_h is calculated as

$$S'_{ui,xh} = \sum_{k=0}^K g_{ui,xh}^k \times p_{ui,xh}^k \quad (1)$$

where $1 \leq i \leq N$, N denotes the total number of users; $1 \leq h \leq H$, where H denotes the total number of commodities; $k(k = 0, 1, 2, 3, \dots, K)$ represents the k th topic of the document in the document-topic matrix; K is the number of topics in the document. $g_{ui,xh}$ represents the sentiment score of user u_i 's review text for item x_h on the K th topic; $p_{ui,xh}$ represents the probability that the review text of user u_i for item x_h contains topic k .

The maximum and minimum standardized formula is used to make the user's score for the product between (1, 5), and the standardized user u_i score for the product x_h is calculated by

$$S_{ui,xh} = \frac{S'_{ui,xh} - S'_{ui,min}}{S'_{ui,max} - S'_{ui,min}} \quad (2)$$

Here $S_{ui,min}$ represents the minimum rating of the user u_i for the purchased item, and $S_{ui,max}$ represents the maximum rating of the user u_i for the purchased item. The sentiment scale ranges from 1 to 5, representing levels of satisfaction from complete dissatisfaction to total contentment. A higher rating indicates greater user satisfaction, while a blank entry denotes the absence of user feedback. Consequently, a matrix of user ratings for the product is generated. According to the user's rating on all items, calculate the average rating $\overline{S}_{ui}$ of user u_i on all rated items while using the following formula

$$\overline{S}_{ui} = \frac{1}{H} \sum_{h=1}^H S_{ui,xh} \quad (3)$$

4. User Similarity Calculation

The similarity between two users is calculated using the cosine similarity method. $\text{sim}(u_i, u_j)$ represents the similarity between user u_i and user u_j , which is calculated as

$$\text{sim}(u_i, u_j) = \frac{\sum_{h=1}^H S_{ui,xh} \times S_{uj,xh}}{\sqrt{\sum_{h=1}^H (S_{ui,xh})^2} \times \sqrt{\sum_{h=1}^H (S_{uj,xh})^2}} \quad (4)$$

where, $S_{ui,xh}$ represents the rating of user u_i on item x_h , and $S_{uj,xh}$ represents the rating of user u_j on item x_h .

The degree of similarity between two users increases as their similarity value grows. Users are then arranged in a descending order based on this measure. Consequently, the top user exhibits greater similarity to the target user, thereby forming a group of users with similar characteristics.

5. Generate Target Recommended Product

According to the Top - N strategy, the top N users are selected from the similar user group as the nearest neighbor users. According to the nearest neighbor users, the rating of the target user to the recommended product is predicted. $g_{uo,xh}$ represents the predicted rating of target user u_o for the recommended item x_h as given by

$$g_{uo,xh} = \frac{\sum_{u_m \in M} \text{sim}(u_o, u_m) \times (S_{u_m,xh} - \overline{S}_{u_m})}{\sum_{u_m \in M} \text{sim}(u_o, u_m)} \quad (5)$$

Here M represents the set of all users in the similar user group who have rated the recommended item x_h . $\text{sim}(u_o, u_m)$ denotes the target similarity between user u_o and user u_m . $S_{u_m,xh}$ is user u_m 's rating of the recommended item x_h ; $\overline{S}_{u_m}$ represents the mean of the ratings given by user u_m to all of its rated items.

IV. EXPERIMENT AND ANALYSIS

1. Experimental Dataset

In this experiment, the main dataset will consist of the review text of 15 high-rated shampoo brands that appear on JD and has 123,844 product reviews that have been left by 6,787 users. After preprocessing [19], the dataset consisted of 88,726 review of products written by 5,213 users, and the average number of words per product review was 56. Then, the dataset was randomly split into the training and test sets in the 8 to 2 ratio. They have been largely done by matrix factorization techniques [20] applied in tasks where the dataset is rated in similar format. The survey on the recommendation landscape by Adomavicius and Tuzhilin [21] validates the considerations of scalability and sparsity management to be at the center of such large-scale experimental settings.

2. Compare Experimental and Evaluation Metrics

Headings, In order to verify the accuracy of the recommendation method, the BTM-CF algorithm is compared with the following baseline algorithms. Topic models such as LDA [22] and broader recommender system frameworks [23]

serve as important reference points for evaluating collaborative filtering advances:

- Content-based Recommendation Algorithm (CB).
- Traditional collaborative filtering recommendation algorithm (CF).

The evaluation metrics used in this study:

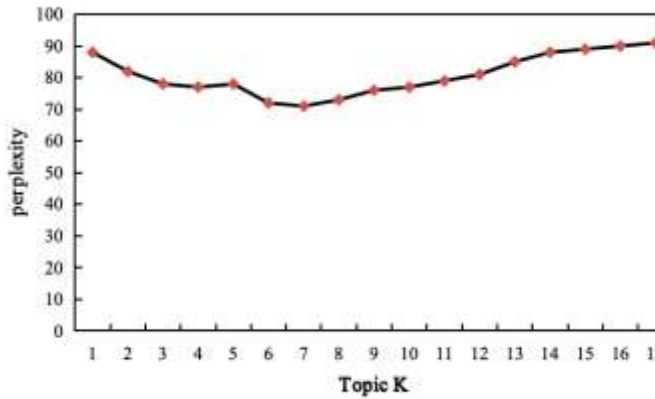


Fig. 3. Relationship between the number of topics K and the perplexity.

Mean Absolute error (MAE) : reflects how similar the predicted rating of the recommendation algorithm is to the actual rating, as shown in the following equation

$$MAE = \frac{\sum_{(i,j) \in E} |u_{ij} - u'_{ij}|}{|E|} \quad (6)$$

Where $u_{i,j}$ represents the actual rating of item j by user i . Let $u'_{i,j}$ denote the predicted rating of user i for item j . $|E|$ denotes the total number of predicted ratings. MAE is a standard evaluation metric in deep learning-based recommender systems [24] and has been adopted for benchmarking review-text collaborative filtering [25]

F1-Score: The precision P and recall R of the classification model are integrated, as shown in Equations (7) to (9):

$$P = \frac{TP}{TP+FP} \quad (7)$$

$$R = \frac{TP}{TP+FN} \quad (8)$$

$$F1 - \text{value} = \frac{2 \times P \times R}{R+P} \quad (9)$$

In this context, TP denotes instances where positive text samples are correctly classified as such, while FN refers to cases where positive text samples are incorrectly classified as

negative. Conversely, TN signifies accurate predictions of negative text samples, whereas FP denotes instances where negative text samples are erroneously classified as positive.

V. EXPERIMENTAL RESULTS AND ANALYSIS

The BTM topic model is applied to analyze user comment data, with parameters estimated using the Gibbs sampling method. The Dirichlet prior parameter $\alpha = 50/T$ and Dirichlet prior parameter $\beta = 0.01$ are set, and the optimal topic number K is determined according to the perplexity method. Through model training, document-topic matrix and topic-word matrix are obtained. The relationship between the number of topics K and perplexity is shown in Figure 3, and it can be seen that when the number of topics $K = 7$, the perplexity of the BTM topic model is the smallest. This is consistent with neural collaborative filtering architectures [26] that similarly optimize latent factor dimensions. The resulting topic-word distributions also permit aspect-level sentiment interpretation [27], confirming the model's semantic interpretability.

Table I. Topic-Word Probability Distribution Of Partial User Review Data

TOPIC	VOCABULARY	PROBABILITY
TOPIC1	GOOD, EXPRESS, EXPRESS FAST, CONVENIENT AND QUICK	0.428, 0.040, 0.031, 0.027
TOPIC2	FAMOUS BRAND, TRUST, BRAND, OLD BRAND	0.354, 0.121, 0.087, 0.401
TOPIC3	BUYBACK, QUALITY, AFFORDABLE, VALUE FOR MONEY	0.272, 0.120, 0.085, 0.046
TOPIC4	PACKING, EXQUISITE, LOGISTICS IN GOOD CONDITION, IN GOOD CONDITION	0.149, 0.090, 0.074, 0.067

Table 1 displays the probability distribution of topic- specific words derived from a subset of user review data analyzed using the BTM topic model. Additionally, Table 2 presents the distribution of topics within the same user review dataset as inferred by the BTM topic model. The algorithm BTM-CF extracts document-topic probability and topic-word distribution from user reviews via the BTM topic model, alongside acquiring user ratings. The process of extracting and summarizing topic-representative words mirrors established opinion mining methods [28] and is theoretically grounded in probabilistic topic modelling surveys [29].

Table II. Document-Topic Probability Distribution Of Partial User Review Data

DOCUMENTATION	TOPIC	PROBABILITY
DOC1	[7,3,5,6,4,2,1]	0.355, 0.270, 0.196, 0.072, 0.006, 0.005, 0.004
DOC2	[5,7,1,6,3,4,2]	0.287, 0.137, 0.074, 0.063, 0.005, 0.004, 0.003
DOC3	[3,7,6,5,4,2,1]	0.201, 0.153, 0.009, 0.009, 0.006, 0.005, 0.003
DOC4	[4,2,7,6,5,3,1]	0.535, 0.419, 0.006, 0.005, 0.004, 0.002, 0.001
DOC5	[6,2,1,5,4,3,7]	0.304, 0.296, 0.170, 0.087, 0.066, 0.056, 0.005

Figure 4 illustrates the recommendation outcomes of BTM-CF, traditional collaborative filtering (CF), content-based methods (CB) and LDA-based collaborative filtering (LDA-CF). Comparative analysis reveals the superior performance of BTM-CF, with notably smaller MAE values and enhanced rating prediction accuracy compared to its counterparts. Similar MAE reductions have been reported in deep learning recommendation studies [30] and in sentiment-aware systems applied to social media data [31], underscoring the benefit of integrating textual signals into rating prediction.

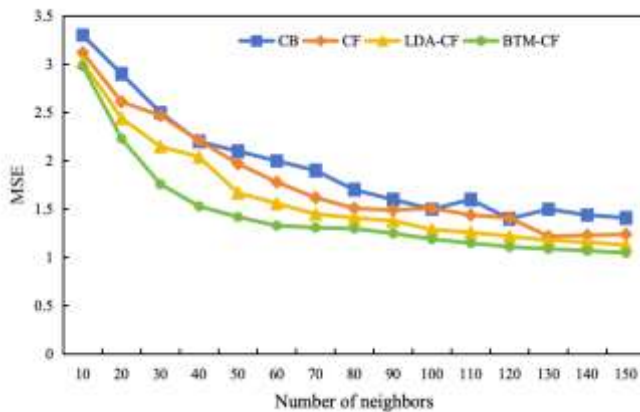


Fig. 4. Comparison of MAE values of different models.

Utilizing the cosine similarity technique, we derive the recommended items for the target user. In this study, we compare the BTM-CF algorithm with the content-based recommendation algorithm (CB), the traditional collaborative filtering recommendation algorithm (CF), and the LDA based collaborative filtering algorithm (LDA-CF). The detailed experimental findings are presented in the table 3. Figures 5

through 7 display the accuracy, recall rate, and F1-score of the recommended models across different topics, respectively.

Figures 5 and 6 illustrate that when the number of recommended items exceeds 10, the recall rate of BTM-CF surpasses that of other baseline models. Additionally, BTM-CF achieves an average accuracy rate that is 2.95% higher than CF and an average recall rate that is 2.01% higher. The experimental results in Figure 7 further demonstrate that the BTM-CF algorithm outperforms baseline algorithms in terms of F1-score. Specifically, BTM-CF achieves higher accuracy on the dataset, exceeding CB, CF, and LDA-CF by an average of 3.25%, 0.97% and 0.56%, respectively. These F1 improvements are in line with findings from comprehensive sentiment analysis surveys [32] and comparative evaluations of short-text topic modelling techniques [33]. Our analysis reveals across-the-board enhancements in BTM-CF performance compared to the other methods, under identical conditions and datasets, resulting in superior recommendation accuracy.

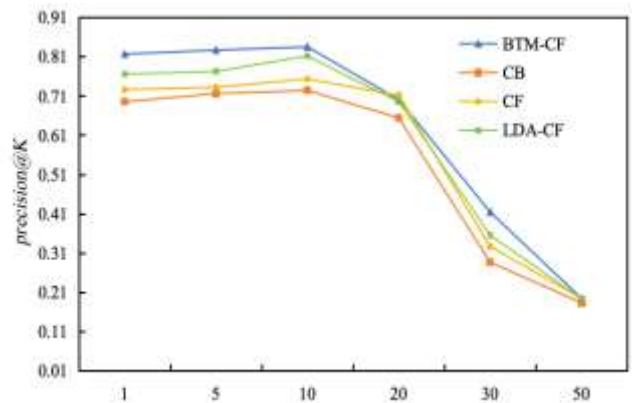


Fig. 5. Top K recommendation comparison of Precision of different models.

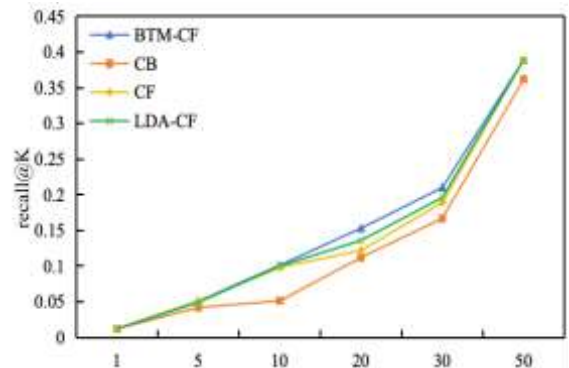


Fig. 6. Top K recommendation comparison of Recall of different models

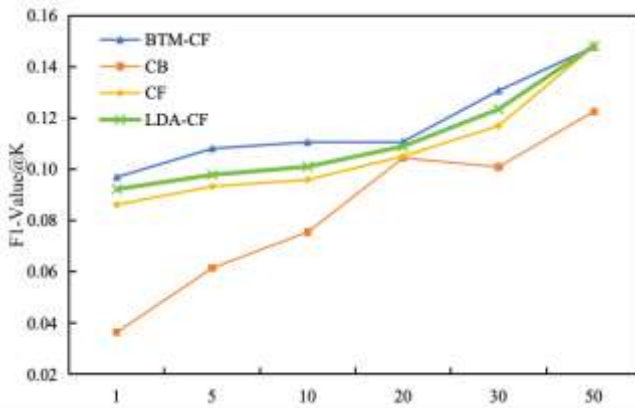


Fig. 7. Top K recommendation comparison of F1-value of different models.

Table Iii. Comparison Of Evaluation Results Of Review Data Sets

Model	Precision	Recall	F1-Score
CB	0.544	0.124	0.084
CF	0.575	0.144	0.108
LDA-CF	0.599	0.147	0.112
BTM-CF	0.632	0.153	0.117

VI. CONCLUSIONS

The collaborative filtering recommendation algorithm presented in this paper is based on the BTM topic model. Compared to conventional approaches, the BTM-CF algorithm uses the review text to thoroughly examine the preferences of the user concerning the topic of products. It has the advantage of making users and items attribute sets richer through the use of the topic model to improve the precision of the recommendations. It has been shown in experiments that the BTM-CF algorithm is good in terms of precision, recall, and F1-value. It is now known that hybrid systems that combine textual semantics with collaborative signals tend to be more effective than rating-based systems as demonstrated in the comprehensive recommender systems handbooks [34]. The current issue of sparse data [35] could be dealt with by sentiment-enhanced user-

item matrices, as it has been shown in this paper, as a viable solution to this particular problem, which is practical and scalable. In the future, research is to be conducted in order to enhance accuracy of text topic mining. The possible solution is to use temporal and spatial data of review texts to analyze topics more efficiently, which will further improve the accuracy of the recommendations of the algorithm

REFERENCES

1. N. Nassar, A. Jafar, and Y. Rahhal, "A novel deep multi-criteria collaborative filtering model for recommendation system," *Knowledge- Based Systems*, vol. 187, p. 104811, 2020.
2. R. Prasad and V. V. Kumari, "A categorical review of recommender systems," *International Journal of Distributed and Parallel Systems*, vol. 3, no. 5, p. 73, 2012.
3. C. Yu and L. Huang, "Clucf: a clustering cf algorithm to address data sparsity problem," *Service Oriented Computing and Applications*, vol. 11, pp. 33–45, 2017.
4. R. Chen, Q. Hua, Y.-S. Chang, B. Wang, L. Zhang, and X. Kong, "A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks," *IEEE access*, vol. 6, pp. 64 301–64 320, 2018.
5. D. Godbole and M. Narnaware, "A survey on personalized service recommendation systems," *International Journal of Engineering Research & Technology*, vol. 15, no. 2, pp. 616–20, 2016.
6. N. S. Bhosale and S. S. Pande, "A survey on recommendation system for big data applications," *Data Mining and Knowledge Engineering*, pp. 42–44, 2015.
7. Q. Zhang, R. Guo et al., "Prediction of mechanical properties of cold rolled steel using fusion of multi-head attention," *IAENG International Journal of Computer Science*, vol. 51, no. 1, 2024.
8. Y.-L. Lee, T. Zhou, K. Yang, Y. Du, and L. Pan, "Personalized recommender systems based on social relationships and historical behaviors," *Applied Mathematics and Computation*, vol. 437, p. 127549, 2023.
9. G. Mihai, "Recommendation system based on association rules for distributed e-learning management

- systems,” *Acta Universitatis Cibinensis. Technical Series*, vol. 67, no. 1, pp. 99–104, 2015.
10. J. S. Breese, D. Heckerman, and C. Kadie, “Empirical analysis of predictive algorithms for collaborative filtering,” *arXiv preprint arXiv:1301.7363*, 2013.
11. P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, “GroupLens: An open architecture for collaborative filtering of netnews,” in *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, 1994, pp. 175–186.
12. B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, “Item-based collaborative filtering recommendation algorithms,” in *Proceedings of the 10th international conference on World Wide Web*, 2001, pp. 285–295.
13. M. Deshpande and G. Karypis, “Item-based top-n recommendation algorithms,” *ACM Transactions on Information Systems (TOIS)*, vol. 22, no. 1, pp. 143–177, 2004.
14. X. Yan, J. Guo, Y. Lan, and X. Cheng, “A biterm topic model for short texts,” in *Proceedings of the 22nd international conference on World Wide Web*, 2013, pp. 1445–1456.
15. J. Wang, L. Chen, L. Li, and X. Wu, “Bittm: A core biterm-based topic model for targeted analysis,” *Applied Sciences*, vol. 11, no. 21, p. 10162, 2021.
16. Y. Sun, K. Loparo, and R. Kolacinski, “Conversational structure aware and context sensitive topic model for online discussions,” in *2020 IEEE 14th International Conference on Semantic Computing (ICSC)*. IEEE, 2020, pp. 85–92.
17. Y. W. Liu, S. B. Xiao, T. Wang, and S. C. Shi, “Building chinese sentiment lexicon based on hownet,” *Advanced Materials Research*, vol. 187, pp. 405–410, 2011.
18. Q. Lei, H. Li, and Y. Chen, “How does chinese segmentation strategy effect on sentiment analysis of short text?” in *2021 IEEE 2nd International Conference on Pattern Recognition and Machine Learning (PRML)*. IEEE, 2021, pp. 234–240.
19. B. Zhang, D. Yang, Y. Liu, and Y. Zhang, “Graph contrastive learning with knowledge transfer for recommendation,” *Engineering Letters*, vol. 32, no. 3, 2024.
20. Y. Koren, R. Bell, and C. Volinsky, “Matrix factorization techniques for recommender systems,” *Computer*, vol. 42, no. 8, pp. 30–37, 2009.
21. G. Adomavicius and A. Tuzhilin, “Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 6, pp. 734–749, 2005.
22. D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent Dirichlet allocation,” *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.
23. J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, “Recommender systems survey,” *Knowledge-Based Systems*, vol. 46, pp. 109–132, 2013.
24. S. Zhang, L. Yao, A. Sun, and Y. Tay, “Deep learning based recommender system: A survey and new perspectives,” *ACM Computing Surveys*, vol. 52, no. 1, pp. 1–38, 2019.
25. M. Srifi, A. Oussous, A. Ait Lahcen, and S. Mouline, “Recommender systems based on collaborative filtering using review texts—A survey,” *Information*, vol. 11, no. 6, p. 317, 2020.
26. X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, “Neural collaborative filtering,” in *Proceedings of the 26th International Conference on World Wide Web*, 2017, pp. 173–182.
27. K. Schouten and F. Frasincar, “Survey on aspect-level sentiment analysis,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 3, pp. 813–830, 2016.
28. M. Hu and B. Liu, “Mining and summarizing customer reviews,” in *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004, pp. 168–177.
29. H. Jelodar, Y. Wang, C. Yuan, X. Feng, X. Jiang, Y. Li, and L. Zhao, “Latent Dirichlet allocation (LDA) and topic modeling: models, applications, a survey,” *Multimedia Tools and Applications*, vol. 78, no. 11, pp. 15169–15211, 2019.
30. A. Da’u and N. Salim, “Recommendation system based on deep learning methods: a systematic review and new directions,” *Artificial Intelligence Review*, vol. 53, no. 4, pp. 2709–2748, 2020.
31. S. Kumar, K. De, and P. P. Roy, “Movie recommendation system using sentiment analysis from microblogging data,” *IEEE Transactions on Computational Social Systems*, vol. 7, no. 4, pp. 915–923, 2020.
32. M. Birjali, M. Kasri, and A. Beni-Hssane, “A comprehensive survey on sentiment analysis: Approaches, challenges and trends,” *Knowledge-Based Systems*, vol. 226, p. 107134, 2021.

33. R. Albalawi, T. H. Yeap, and M. Benyoucef, "Using topic modeling methods for short-text data: A comparative analysis," *Frontiers in Artificial Intelligence*, vol. 3, p. 42, 2020.
34. C. C. Aggarwal, *Recommender Systems: The Textbook*. Cham: Springer International Publishing, 2016.
35. H. Yin, Q. Wang, K. Zheng, Z. Li, and X. Zhou, "Overcoming data sparsity in group recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 7, pp. 3447–3460, 2022.