CTU Journal of Innovation and Sustainable Development
ISSN 2588-1418 | e-ISSN 2815-6412

DOI:10.22144/ctujoisd.2024.318

Product recommendation for online sales systems based on transaction sessions

Mai Le Bich Tuyen[1], Nguyen Thanh Hai[2], and Tran Thanh Dien[3*]

[1]FPT Can Tho, Viet Nam

[2]College of Information and Communication Technology, Can Tho University, Viet Nam

[3]Can Tho University Publishing House, Can Tho University, Viet Nam

*Corresponding author (thanhdien@ctu.edu.vn)

Article info.

Received 26 Jun 2024
Revised 29 Aug 2024
Accepted 29 Sep 2024

Keywords

Memory-based collaborative filtering, model-based collaborative filtering, online sales systems, transaction sessions

ABSTRACT

The advancement of information technology has influenced and accelerated the growth of many economic fields and industries, including e-commerce. This is one area of information technology that has rapidly developed and gained widespread popularity due to the significant benefits it offers to the community. Today, the need for online shopping is increasing because of its convenience and timesaving, especially for busy people. Historical session data plays an important role in helping sales systems recommend orders to customers that suit their personal preferences. This study proposes models for recommending products on online sales systems based on transaction sessions using memory-based collaborative filtering methods, including user-based and item-based, and model-based collaborative filtering methods including SVD and KNN. The experimental results show that the SVD model has a better rating prediction performance than other techniques. Therefore, it is probably proposed for product recommendations on online sales systems.

1. INTRODUCTION

In today's technological age, online sales markets are increasingly expanding and diverse, attracting the attention of consumers. Searching and choosing the preferred product is a shopping experience and a process of exploration and discovery.

However, given the diversity of products and the constant change in beauty trends, consumers often have difficulty finding suitable products. This creates a great opportunity to develop an intelligent product recommendation system based on the user's transaction session history. This recommendation system does not display products related to the user's shopping history but also focuses on predicting upcoming trends based on previous transactions. This helps optimize the shopping experience, making it easier for users to discover new products and drive sales at the same time. Therefore, e-commerce transactions, in general, and online sales, in particular, need to have a product recommender system that suits and changes according to customer preferences.

Users' need for online shopping is increasing because of convenience and time savings, especially for busy people. Instead of waiting to pay at shopping centers and supermarkets, people can stay at home and choose the items to buy with a few simple steps and then just wait for sellers to deliver the goods. Therefore, product recommendations for online sales systems based on transaction sessions were proposed to build a recommendation model that would make the customer shopping process more convenient and effective.

2. RELATED WORK

Quadrana et al. (2017) proposed RNN (Recurrent Neural Networks) models to personalize for a recommendation system based on users' transaction sessions. They include RNN models with cross-session information transfer and a hierarchical RNN model for relaying latent hidden states of the RNNs across user sessions. The obtained results show large improvements over the session-only RNNs. Meanwhile, Wang et al. (2014) from the University of Kentucky proposed two algorithms of NMF (Non-Negative Matrix Factorization) and SVD (Singular Value Decomposition) for recommending based new data update approach in collaborative filtering and providing both high-level privacy protection and good data utility. Another interesting topic comes from the researchers who built a recommender system that not only provides users with satisfaction but also brings healthy profits to the companies (Ma & Lu, 2014). To deal with the huge amount of data, the authors proposed an effective recommender system to model rich historical data. Initially, consult a new strategy for data splitting to ensure the recommendation system is appropriate for large-scale advertising. Then, a distributed recommendation algorithm is designed. Finally, the two applied effective recommend systems and distributed computing and storage tools to model historical data to improve system efficiency.

To recommend items for users to have interesting experiences on social networking sites. The study of authors (Guan & Lu, 2012) was based on a subset of data taken from KDD Cup 2012 and then used the algorithms on the prediction task which involves predicting whether a user will follow a proposed item to users in social networking services. A study on building a recommendation system based on transaction history is also of interest to readers (Zhang et al., 2018). This study introduces a method of using the self-attention mechanism to infer the item-item relationship through the user's historical interactions. The experiments on datasets in different domains demonstrated that the proposed approach outperforms the state-of-the art.

Other authors used the Bayesian Personalized Ranking Matrix Factorization combined with the research to propose predicting user preferences based on feedback (Thai-Nghe & Tan-Phong, 2014). The results show that it is possible to integrate the model built with the above algorithm into the existing music recommendation systems. The research proposed by Thu and Nghi (2016) proposed the application of the collaborative filtering method of the recommender system with keyword search index functionality (Elastic Search) to the search function. The results show that an optimal recommender system based on search keywords, document names, and book borrowing history helps improve the efficiency of document lookup at libraries.

According to Thai-Nghe et al. (2022), techniques play an important role in recommendation systems to support users in finding suitable products in online systems. The authors proposed a detailed architecture of a deep matrix factorization for recommendation that is also compared with the standard matrix factorization model. The experimental results show that the deep matrix factorization model can work well for recommendations in online e-commerce systems.

Nguyen et al. (2023) proposed a recommendation system for a clothing online sale system based on analyzing context-based and collaboration-based methods. The algorithms, including the K-nearest neighbor's algorithm (KNN), singular value decomposition (SVD), non-negative matrix factorization (NMF), and matrix factorization (MF) were deployed for the comparison. From the experimental results, the authors proposed a content-based memory-based method using Word2vec + IDF and a collaboration-based model-based method using the SVD algorithm to deploy on the sales system.

The internet has become a one-stop shop for daily necessities, with food being a prime example of items that can be enhanced by recommendation systems. Recognizing this trend, researchers from the Maharashtra Institute of Technology explored the potential of such systems (Panicker et al., 2016). Their project aims to develop a recommendation engine for online grocery shopping, designed to suggest products that align with consumers' preferences and needs. The system also proposes to offer personalized shopping carts, taking into account individual customer profiles and past purchase behavior to some degree.

A dynamic recommendation system (DRS) for e-commerce has been proposed by two Indian researchers. This model aims to address challenges related to customer ratings and reviews in online retail (Kulkarni & Gulavani, 2022). The DRS integrates multiple approaches, including market basket analysis, frequent item set mining, topselling product data, and personalized customer preferences. By combining these elements, the system offers a sophisticated solution to enhance product recommendations in the digital marketplace.

Other researchers proposed a novel approach combining an online food marketplace with a recommendation system (Lee et al., 2020). Their multi-phase product recommendation system aims to understand consumer behavior by analyzing order histories and recurring purchase patterns. Instead of traditional collaborative filtering techniques, the researchers employed Recurrent Neural Networks (RNNs), which excel at processing sequential data. Experiments using real-world online food market data showed that this proposed system outperformed additive filtering-based methods in a multi-stage context, demonstrating improvements in both accuracy and diversity of recommendations. The results suggest that the system's ability to closely examine customer orders and purchasing trends makes it suitable for suggesting multi-stage products. This approach aligns with the researchers' long-term objectives of reducing food waste, improving nutritional intake, and streamlining the shopping experience while incorporating regular customer feedback.

Researchers collaborated on developing a recommendation system for fashion brands with both online and brick-and-mortar presence (Eldemerdash et al., 2023). Their novel approach, called K-RecSys, builds upon traditional item-based collaborative filtering methods while accounting for specific domain characteristics. K-RecSys incorporates click-stream data from online products and weighted offline sales data to reflect both online and in-store customer preferences. Experimental results demonstrated that this method outperforms existing systems in terms of product clicks and sales in online shopping environments. Moreover, K-RecSys's alternative recommendations showed higher adoption rates compared to additional suggestions. Aggarwal et al. (2024) explore the potential of integrating contextual information into intelligent recommendation systems. Their study focuses on improving the online shopping experience within the fashion industry through these enhanced systems.

This study proposes session-based recommendation models based on both methods of memory-based memory collaborative filtering (CF) including user-based CF and item-based CF, and model-based collaborative filtering including singular value decomposition (SVD) and k-nearest neighbors (KNN). From the obtained results, we recommend the model with the lowest errors, high accuracy and short training time.

3. METHODS

This study was conducted with the hope of providing the best answer to customers' questions, "What should I buy when there are thousands of products on a sales website?" or proposing an effective solution to improve the situation when customers have experienced the frustration of receiving product recommendations that have not related to their preferences. The study proposed a recommender system model based on memory-based collaborative filtering and model-based collaborative filtering using transaction session data.

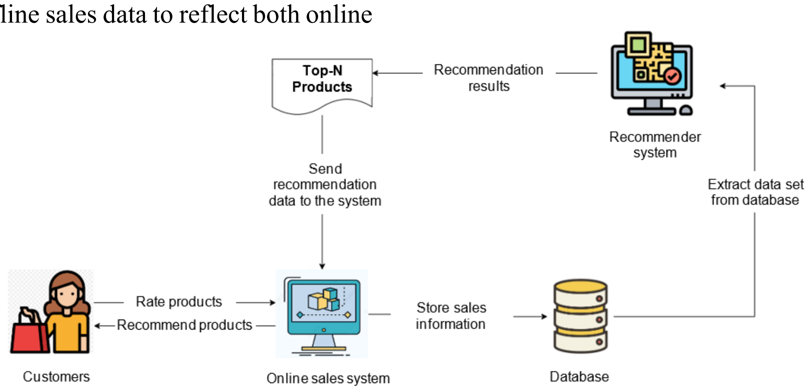

Figure 1. Proposed online sales recommender system

The online sales recommender system is described in Figure 1. The system receives data from the online sales system that has collected product ratings from customers when they have bought their products. The data set is retrieved from the database and passed through the recommender system. The returned result is the top-N products that can be recommended for the customers.

The working process of the recommendation system is described as Figure 2. In this figure, the implementation processing of the recommendation system is carried out through the following steps:

Step 1. Data set is converted from database to readable machine learning libraries, mainly in CSV file format.

Step 2. Data analysis is carried out from customer ratings about the product.

Step 3. Data preprocessing is implemented, such as duplicate handling, data loss handling, etc. before applying data to algorithms to conduct model training.

Step 4. The model is trained. Then, we back up and continue to the next step.

Step 5. Predicting customer ratings for products based on the trained model.

Step 6. Returning a list of Top-N products with predicted ratings arranged in ascending order to produce a list of product recommendation.

This study uses the CF recommendation method based on two approaches: (1) memory-based CF and (2) model-based CF.

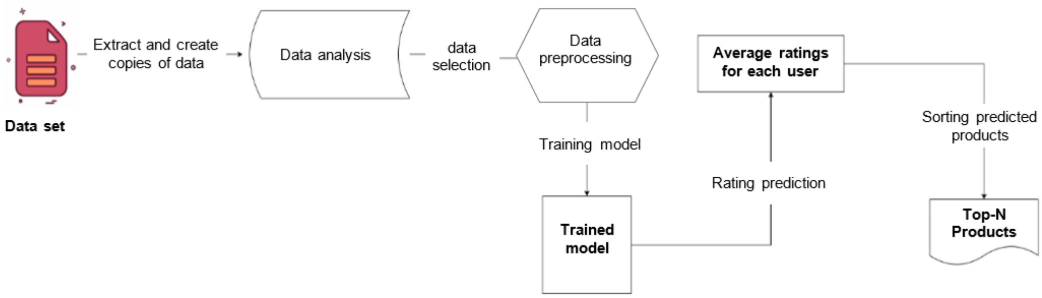

Figure 2. The implementation processing of the recommendation system

Memory-based CF: It includes user-based CF and item-based CF. The method of user-based CF is based on the assumption that behaviorally similar users prefer similar products. Therefore, to make recommendations for a user, the recommender system finds users with similar behavior and uses information about their ratings to make recommendations. On the other hand, the method of item-based CF assumes that similar products are evaluated similarly by all users. The system finds products identical to the reviewed product and uses information about their reviews to make recommendations.

There are several measures to calculate similarity. The method of cosine distance computation used in this study, with the following formula:

$$\text{Cosine (A,B)} = \cos(\theta) = \frac{A.B}{||A||\,||B||} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\ \sqrt{\sum_{i=1}^{n} B_i^2}}$$

In there:

A.B is the dot product between vectors A and B.

||A|| and ||B|| is the norm of vector A and vector B, respectively.

This study uses Euclidean distance to measure the similarity between two vectors representing users or products to help generate rating predictions based on purchase history information. In the n-dimensional space, dimensions corresponding to users and products are represented. Each user or product vector in this space may contain information about related features, such as purchase history, ratings, or other attributes. These vectors are then used to predict a product rating based on the user transaction history or vice versa. The smaller the Euclidean distance, the more similar the two vectors are.

The formula of the Euclidean distance for n-dimensional space is as follows:

$$\text{distance} = \sqrt{\sum_{i=1}^{n} (x_{2i} - x_{1i})^2}$$

Model-based CF is often used in mathematical models to make predictions. This study used a collaborative filtering algorithm based on K-nearest neighbors (KNN) and the method of single-value decomposition or SVD.

The KNN algorithm is a widely used technique in recommender systems. The basic principle of collaborative filtering is to predict user preferences by identifying other users with similar preferences and using their ratings to make recommendations. The KNN-based collaborative filtering algorithm is a type of collaborative filtering that assigns ratings to items by leveraging the ratings of the K users most similar to the target user. The KNN algorithm measures the similarity between users using a distance measure. It then identifies the K users most similar to the target user and calculates their weighted average rating for each item. These weights are determined using the similarity score between the target user and each of the K users. The predicted rating for an item is a weighted average of the ratings of the K users.

The KNN algorithm is a simple supervised machine learning algorithm that can be used to solve both classification and regression problems. It has the advantage of being easy to implement and understand, but the disadvantage of becoming slow as the data usage size is increasing. The formula of KNN is as follows:

$$\hat{r}_{ui} = \frac{\sum_{v \in N_i^k}(u)\, \mathrm{sim}(u, v).\, r_{vi}}{\sum_{v \in N_i^k}(u)\ \mathrm{sim}(u, v)}$$

In there:

$\hat{r}_{ui}$: expected rating of customer for product i

k: number of neighbors, in this study using k=10 for comparison

sim(u,v): similarity of customer u to customer v

r_{vi}: Rating of customer for product i

In SVD, singular values are often represented in a diagonal matrix $\sum$. It represents the "magnitude" of the corresponding degenerate components in the analysis. A usually refers to the evaluation matrix. SVD is an algorithm that decomposes a matrix X into the product of three matrices: U (matrix containing user information), $\sum$ (matrix containing diagonal degenerate values) and V^T (matrix containing product information). To build a model based on the SVD algorithm, we have A as the evaluation matrix, m as the number of rows and n as the number of columns. Then the formula is as follows:

$$A = U \sum V^T$$

Where:

U: User matrix - latent features ($m \times m$)

$\sum$: A diagonal matrix of sigma values ($m \times n$)

V^T: Product matrix - lantent features ($n \times n$)

The formula to calculate each matrix is as follows:

Matrix U: contains eigencolumn vectors of AA^T.

Matrix $\sum$: contains the sigma values on the main diagonal, which is the square root of the positive eigenvalues of AA^T.

Matrix V^T: contains the eigenvectors of A^TA.

Prediction process: after finishing training, we get the values and two optimal decomposition matrices. The prediction process is the same as factorizing the matrix and using the above formula.

3.1. Data description

This study uses the Amazon Ratings dataset, which contains more than 2 million reviews related to beauty products sold on Amazon.com. This dataset includes customer ratings, Amazon product ASINs, and review scores (from 1 to 5). It has three typical columns: UserId, ProductId, and Rating, as described in Table 1. This information provides diverse data about user reviews and experiences for different products.

Table 1. Samples of each attribute in the data set

Attribute	UserId	ProductId	Rating
# Sample	1,210,271	249,274	2,023,070

Table 2 shows us the total number of ratings for each scale of 5 and through that, we can see that 5/5 is the rating most rated by customers.

Table 2. Rating scores for each scale

Ratings	5	4	3	2	1
# Sample	1,248,721	307,740	183,784	169,791	113,034

3.2. Evaluation metrics

This study uses different techniques for product recommendation for online sales system based on transaction sessions, specifically memory-based CF including user-based CF and item-based CF, and model-based CF including SVD and KNN. In this study, two common measures, RMSE and MAE, are used to evaluate the models. For SVD and KNN, we run 10-fold cross-validation to evaluate the performance of the model and compare the two models with each other.

4. EXPERIMENTAL RESULTS

As described in Table 1, the data set used for this study has 1,210,271 customers and 249,274 products with a total of 2,023,070 reviews. The data set is divided into 80% as the training set with 1,618,456 and 20% as the test set with 404,614.

First, we use the two techniques of user-based CF and item-based CF. To evaluate the model, we choose products that have been rated at least 500 times and we get 174 products with appropriate ratings.

In order to balance computational economy and performance estimate reliability at the start of the project, we select ten times out of a possible 20. While additional folds (e.g., 20) can be computationally expensive without delivering projected significant improvement, fewer folds (e.g., 5) can result in more variance in performance measures. In order to balance computational economy and performance estimate reliability at the start of the project, we select ten times out of a possible 20. While additional folds (e.g., 20) can be computationally expensive without delivering projected significant improvement, fewer folds (e.g., 5) can result in more variance in performance measures.

A 500 rating criterion can help strike a compromise between rejecting products with inadequate data and having enough data to generate accurate forecasts. Five hundred choices will help prevent overfitting. Additionally, by concentrating on goods that have received a lot of feedback, the functioning of the suggested system is made more stable and reliable.

From the filtered data, we evaluate the model and obtain the error of the two models as shown in the following table:

Table 3. MAE and RMSE measure comparison of User - Based CF

Measure	User – Product	User – Ratings
MAE	4.1849	1.6181
RMSE	4.3732	1.7365

Table 4. Measure comparison of Items - Based CF

Measure	User – Product	User – Ratings
MAE	4.1849	1.6181
RMSE	4.3732	1.7365

From Table 3 and Table 4, we can see that the MAE and RMSE of the two methods are similar. Therefore, dividing data with ratings columns (user-ratings and product-ratings) gives a more accurate model than dividing the remaining data. Therefore, we took measurements by dividing the data with an evaluation method to compare with the two model-based collaborative filtering methods in the next step.

Selecting a product with 300 ratings can still have less information than one with 500 or more. Fewer ratings may potentially miss user preferences, which could affect the model's predicted stability and accuracy. User-rated splits can use more data points from individual reviews, allowing the model to acquire more information from dense data sets. This means that even if we decrease the threshold to 300 reviews, a user-rated splits-based model may still outperform a user-rated splits-based model.

Next, two model-based collaborative filtering are selected including SVD and KNN. The two models based on these two algorithms are both built with 10-fold. The KNN model has a training of about 96 hours, while the SVD model is about 3 hours faster than the KNN. The results of these two models are shown in Table 5:

In Table 5, it can be observed that the model using the SVD algorithm has better performance than the KNN one in the model-based collaborative filtering method. From those results, we take the average value of the measures of the SVD and KNN models to compare with the memory-based collaborative filtering method.

To prevent overfitting and inappropriate models, set the k value for both the SVD and KNN models to 10. A value larger than ten may result in an unsuitable or subpar model. A value of 10 is the ideal choice for both models. For each k, the two models' MAE values primarily rise while the RMSE falls. Training for the KNN model takes four days, whereas the SVD model needs three hours each day.

Table 6 shows a measurement comparison of the KNN, SVD, and User-Based CF models in the user-rating and Item-Based CF in the product rating. From the results, we can see that the SVD is the model with the lowest errors.

Table 5. MAE and RMSE measure comparison of KNN and SVD

Model / Fold	KNN		SVD	
	MAE	**RMSE**	**MAE**	**RMSE**
1	18.8289	4.339	0.7890	1.8350
2	15.4114	3.925	0.7692	0.8938
3	8.5588	2.9255	0.7973	0.8810
4	4.3531	2.0864	0.7870	0.8631
5	2.5833	1.6072	0.7736	0.8554
6	1.5716	1.2553	0.7623	0.8604
7	1.1116	1.0643	0.7600	0.8598
8	0.8248	0.9081	0.7876	0.8520
9	0.6426	0.8016	0.7639	0.8485
10	**0.5624**	**0.7499**	**0.7942**	**0.8449**
AVG	5.4485	1.9662	**0.7784**	**0.9593**

Table 6. Comparison of the MAE and RMSE average of four methods

Model	User-Based CF	Item-Based CF	SVD	KNN
MAE	1.6181	1.6181	**0.7784**	5.4485
RMSE	1.7365	1.7365	**0.9593**	1.9662

With the Amazon data set as mentioned, the experimental results give two measures MAE = 0.7784 and RMSE = 0.9593 of the SVD model. It can be seen that the average of the two error degrees is the lowest among the four models according to the two methods of collaborative filtering. In addition, experimental results also show that the SVD model has fast training time and high accuracy performance when using large data sets, so SVD is the optimal choice for this research.

5. CONCLUSION

In this study, we propose approaches for building models recommending products for online sales systems based on transaction sessions using memory-based CF methods including user-based CF and item-based CF, and model-based CF methods including SVD and KNN. The experimental results show that the SVD model has good rating prediction performance compared to other techniques in the lowest errors, high accuracy and short training time; thereby, it is suitable for product recommendations for online sales systems.

However, this study has some limitations that need to be improved in the future. The study needs to compare with many other techniques and models, especially machine learning and deep learning models. Additionally, larger transaction session data needs to be collected in order to accurately evaluate the models and integrate them into sales systems for assessing practical effectiveness.

REFERENCES

Aggarwal, D., Sharma, D., & Saxena, A. (2024). Enhancing the online shopping experience of consumers through artificial intelligence. *International Journal of Information Technology and Computer Engineering*, *4*(1), 1-5. https://doi.org/10.55529/ijitc.42.1.5

Eldemerdash, D., shikh, K., & Abou-Ghali, M. (2023). Fashion recommendation system and its impact on consumers' purchase decision making. *International Design Journal*, *13*, 195-206. https://doi.org/10.21608/idj.2023.276188

Kulkarni, M., & Gulavani, S. (2022). Role of recommender system for selling products in online shopping. *International Journal of Humanities and Social Science*, IRJHISIC2203087, 660-666.

Lee, K., Choi, Moon, & Kim, D. (2020). A multi-period product recommender system in online food market based on recurrent neural networks. *Sustainability*, *12*, 969. https://doi.org/10.3390/su12030969

Guan, L., Lu, H. (2012). Recommend items for user in social networking services with CF. *2012*

International Conference on Computer Science and Service System (pp. 1347-1350). https://doi.org/10.1109/CSSS.2012.340

Ma, J., & Lu, Y. (2014). A recommend system for modelling large-scale advertising. *International Conference on Cyberspace Technology (CCT 2014)* (pp. 1-4). https://doi.org/10.1049/cp.2014.1291

Nguyen, H. T., Ngo, V. H., & Dien, T. T. (2023). Context-based and collaboration-based product recommendation approaches for a clothes online sale system. In T. D. L. Nguyen & J. Lu (Eds.), *Machine Learning and Mechanics Based Soft Computing Applications* (pp. 41-52). Springer Nature Singapore. https://doi.org/10.1007/978-981-19-6450-3_6

Panicker, S., Mane, P., Pakhale, C., Fulzele, S. and Rathi, A. (2016). Online grocery recommendation system. *International Journal for Scientific Research & Development*, *4*(4), 318-321.

Quadrana, M., Karatzoglou, A., Hidasi, B. and Cremonesi, P.(2017). Personalizing Session-based Recommendations with Hierarchical Recurrent Neural Networks. *In Proceedings of the Eleventh ACM Conference on Recommender Systems* (pp. 130-137). https://doi.org/10.1145/3109859.3109896

Thai-Nghe, N., & Tan-Phong, N. (2014). Building a music recommendation based on implicit feedbacks. *CTU Journal of Science*, *34*, 81-91. (in Vietnamese).

Thai-Nghe, N., Thanh-Hai, N. and Dien, T. T. (2022). Recommendations in e-commerce systems based on deep matrix factorization. *Future Data and Security Engineering. Big Data, Security and Privacy, Smart City and Industry 4.0 Applications* (pp. 419-431). Springer Nature Singapore, Singapore, pp. 419-431.

Thu, T. N. M., & Nghi, H. Q. (2016). Recommender system for assisting document search. *CTU Journal of Science*, *43*, 126-134 (in Vietnamese).

Wang, X., Zhang, J., Lin, P., Thapa, N., Wang, Y., & Wang, J. (2014). Incorporating auxiliary information in collaborative filtering data update with privacy preservation. *International Journal of Advanced Computer Science and Applications*, *5*(4), 224-235. https://doi.org/10.14569/IJACSA.2014.050432

Zhang, S., Tay, Y., Yao, L., & Sun, A. (2018). *Next item recommendation with self-attention*. Arxiv. https://arxiv.org/abs/1808.06414