Application of Improved Collaborative Filtering in the Recommendation of E-commerce Commodities

D. Chang, H.Y. Gui, R. Fan, Z.Z. Fan, J. Tian

Dan Chang
School of Economics and Management
Beijing Jiaotong University, China
No.3 Shang Yuan Cun, Haidian District, Beijing, China
6787@bjtu.edu.cn

Haoyu Gui
School of Economics and Management
Beijing Jiaotong University, China
No.3 Shang Yuan Cun, Haidian District, Beijing, China
17120610@bjtu.edu.cn

Rui Fan*
School of Economics and Management
Beijing Jiaotong University, China
No.3 Shang Yuan Cun, Haidian District, Beijing, China
*Corresponding author: 17120607@bjtu.edu.cn

Zezhou Fan
School of Economics and Management
Beijing Jiaotong University, China
No.3 Shang Yuan Cun, Haidian District, Beijing, China
17125463@bjtu.edu.cn

Ji Tian
Beijing Research Institute of Automation Machinery Industry Co, Ltd
No.1 Jiaochangkou Street, Xicheng District, Beijing, China
15010118013@163.com

Abstract: Problems such as low recommendation precision and efficiency often exist in traditional collaborative filtering because of the huge basic data volume. In order to solve these problems, we proposed a new algorithm which combines collaborative filtering and support vector machine (SVM). Different with traditional collaborative filtering, we used SVM to classify commodities into positive and negative feedbacks. Then we selected the commodities that have positive feedback to calculate the comprehensive grades of marks and comments. After that, we build SVM-based collaborative filtering algorithm. Experiments on Taobao data (a Chinese online shopping website owned by Alibaba) showed that the algorithm has good recommendation precision and recommendation efficiency, thus having certain practical value in the E-commerce industry.
Keywords: recommendation precision, recommendation efficiency, support vector machine (SVM), collaborative filtering.

1 Introduction

With the rapid development of the Internet and mobile Internet, the E-commerce industry is booming with broad attention from all walks of life. According to Research Report on Market Prospect and Invest Opportunity of E-commerce Industry in China from 2018 to 2023, the overall

transaction scale of China's E-commerce reached 24.1 trillion Yuan in 2017, an increase of 17.4%. With the gradual improvement of the E-commerce industry, it is estimated that its transaction scale will reach 28.4 trillion Yuan in 2018, a year-on-year increase of 17.8% [32]. The flourish of the E-commerce industry has led to the explosion of different kinds of data. And the data, which contain great value, are invisible assets for the E-commerce industry. However, not all data are valuable, so users have to spend much time in extracting useful and specific information from a vast amount of data.

With the increasingly booming information in the E-commerce industry and the extraction of valuable information, recommender systems emerge as the times demand – E-commerce websites begin to solve the problem of "information overload" [3] through recommender systems. Commodity recommender systems can record user's characteristic information and their behavioral information such as purchasing and browsing. By analyzing the information obtained from modeling of user's preference, commodity resources that fit user's potential demand or may interest them will be extracted from the commodity information on E-commerce websites, and then recommended to users. Collaborative filtering, with many advantages such as no need to consider the content of recommendation item, offering novel recommendation, little disturbance while browsing websites, as well as easy to achieve technically, becomes the basic algorithm of recommender systems.

Traditional collaborative filtering has the limitations of recommending only on the basis of single indicator — either user's marks or comments. However, marks and comments should not be studied separately, because inconsistency between marks and comments often exists in real life. For example, a user may give a high mark for a commodity, but a dissatisfied comment at the same time and people's cognition on mark differs. Therefore, traditional collaborative filtering is defective in terms of precision.

Nowadays, it is a research hotspot to combine two or several different methods to solve practical problems to make up for the defects of traditional algorithm. For example, in [30] Zhao et al. combined the lexicographic method and Pareto method to optimize the flight ground support capability of airport. Also, to extend the classical vehicle routing problem, they proposed a time-dependent and bi-objective vehicle routing problem with time windows [29]. Therefore, we pretend to combine SVM and collaborative filtering recommendation to improve the recommendation efficiency and precision.

Before giving comprehensive grades, commodities are divided into positive-feedback commodities and negative-feedback commodities in the paper with SVM, namely commodities users like and dislike. Collaborative filtering recommendation is only implemented on positive-feedback commodities. The original data for recommendation reduce greatly due to the classification in advance, so the improved collaborative filtering increases efficiency compared with the traditional one. Finally, online data on Taobao are used to verify that the improved collaborative filtering promotes recommendation precision and efficiency significantly.

2   Literature review

Collaborative filtering is most commonly used in recommender system. It was first proposed by scholars such as Goldberg in 1992, and implemented in TaPestry — the experimental mail system, enabling the system to extract user-interested and effective email list [6]. Mainstream collaborative filtering today can be divided into user-based collaborative filtering and item-based collaborative filtering. At present, collaborative filtering is the research focus in the academic circle. As traditional collaborative filtering has drawbacks such as cold start, data sparsity and poor extensibility, in order to come up with better solutions to these problems, scholars now focus on the improvement of traditional collaborative filtering.

For example, in [7] Guo et al. proposed a novel method called "Merge" to incorporate social trust information, and supplement user preference by merging users' trusted neighbor ratings. In [8] Hu et al. integrated time information into collaborative filtering similarity measure in collaborative filtering algorithm, and designed a hybrid personalized random walk algorithm; Yong-ping Du et al. [5] proposed item-based RBM, and used deep and multilayer RBM network structure to solve the problem of data sparsity; Sedhain et al. [20] generalized matrix algebra framework, and they doesn't need the target user's data when the side information is available ; Jian Wei et al. [25] put forward two models on the basis of a framework based on tight-coupling collaborative filtering and the in-depth study into neural network; A. Murat Yagci et al. [26] focused on frequent co-occurrence items and proposed SASCF to eliminate the cold start of the system; Su Hongyi et al. [22] proposed a new algorithm involving time decay factor in the CF algorithm, and deployed time weights on the MapReduce parallel computing framework ; Xiuju Liu et al. [13] presented a new algorithm of CF-ISEGB, and took the influence sets of current e-learning groups into consideration to effectively solve problems caused by sparse data sets.

Besides, recommendation precision and recommendation efficiency are also two important indicators to assess collaborative filtering. Therefore, many scholars have improved the algorithm by promoting its recommendation precision and efficiency. For example, Mehrbakhsh Nilashi et al. [17] provided the probability of precise recommendation by considering users' preference in many aspects of the items, and introduced vague method to eliminate the uncertainty of users' preference. In [18] Mahdi Nasiri et al. promoted the predictive accuracy of collaborative filtering by initialized factor matrix. Feng Zhang et al. [27] designed linear time algorithm to calculate similarity, thus reducing the time of assessment. In [18] Zhongya Wang et al. calculated PB-level data by parallel computing and proposed effective collaborative filtering based on multi-GPU. Kasra Madadipouya in [14] added location factors to the traditional film collaborative filtering, and improved the accuracy and the quality of recommendation in practical application. Nicola Barbieri in [2] proved that basic probability framework is useful in the generation of the recommendation list, and enhanced the accuracy of recommendation. In [10] Gai Li et al. proposed a new model named PPMF by using RankRLS to solve the problem of low recommendation accuracy and the high cost. With improved algorithms above, scholars have promoted recommendation precision and efficiency. Therefore, it is of significance to improve traditional recommendation algorithm from the perspectives of recommendation precision and recommendation efficiency.

The collaborative filtering in the paper adopts the classification model of SVM. As an implementation method of statistical theory in practice, SVM maps input variable on high-dimensional space by nonlinear mapping, then constrains questions through quadratic optimization and finds out the optimal classification hyperplane, thus maximizing the distance between data and the optimal classification hyperplane. The core concept is to reduce the error of classification to the greatest extent. Many scholars at home and abroad use SVM classifier to classify some data and establish prediction model. Huayu Li et al. separated data into positive and negative feedbacks by a SVM-like task [11]. Uricar et al. in [23] made classification with SVM after figuring out deep characteristic data represented by people's facial expressions, thus predicting their age, gender and smiles. In [15] Asha S. Manek et al. combined feature extraction with SVM and made sentimental classification among online film reviews to predict the popularity of a film. Anish Jindal et al. in [9] combined DT with SVM to precisely predict electricity theft, reducing false alarms greatly. A.S. Ahmad et al. in [1] used the Least Square Support Vector Machine (LSSVM) to forecast electrical energy consumption of buildings. In [21] Selakov et al. used the combination of PSO and SVM to forecast short-term electrical load according to the significant temperature variations. Dongwen Zhang et al. in [25] used SVM in sentimental classification to extract the valuable information.

As SVM can do well in predicting data, scholars at home and abroad combine it with collaborative filtering to predict items or products users may like and then recommend them to users. In order to solve the problem that recommender system is easy to be attacked by shilling, Wei Zhou et al. in [31] combined SVM with TIA and used borderline SMOTE to relieve class-imbalance, thus detecting shilling in the system. Lifang Ren et al. in [19] combined SVM with collaborative filtering to improve prediction precision as far as possible, which meant that SVM was used to filter out services users might dislike, and services listed on top N would be recommended according to preference . Problems related to recommendation precision and efficiency are generally caused by data missing, and scholars have proposed different solutions to these problems. In [16] Mehrbakhsh Nilashi et al. made the SVM and multi-criteria collaborative filtering (MC-CF) as a combination to improve the recommendation precision. Yeounoh Chung et al. in [4] used the SVM to find personalized experts, then these experts will be recommended to different users. In [12] Zhan Li et al. used multiple-kernel SVM to recommend new videos, which can relieve the problems of data sparsity and item cold start . Therefore, SVM-based collaborative filtering proposed in the paper is of significance in promoting recommendation precision and efficiency.

3   SVM-based collaborative filtering

3.1   Commodity information acquisition

In order to verify the actual effect of improved collaborative filtering on E-commerce industry, Python-based Scrapy is adopted in the paper to acquire online commodity information on Taobao, mainly including commodity name, commodity information and user's comments on it. Commodities on Taobao are graded by a 5-star marking system, and in the paper, it is shifted into a 5-point marking system. In addition, according to a huge amount of comments, Taobao translates them into characteristic value of commodities by semantic analysis, such as good stuff, fast logistics, etc. In the experiment, 3,4000 pieces of data are acquired and part of them are demonstrated in Table 1.

3.2   SVM-based classification

The data set in the experiment mainly includes marks and characteristic values of commodities. A 2500-dimension vector set is built based on the data, and the characteristic value which is nonexistent will be filled with 0. Suppose that data set of commodities on Taobao is $\{(x_i, y_i) | i = 1, 2, ...n\}$; the data set of commodities available for recommending is $\{x_i | i = 1, 2, ..., n\}$; $x_j = (x_{j1}, x_{j2}, ..., x_{jk})$ and $x_i = (x_{i1}, x_{i2}, ..., x_{ik})$ are characteristic attributes of set i and set j; $y_i \epsilon \{-1, 1\}$ is the output type. $y_j = -1$ means the commodity has negative feedback; $y_j = 1$ means the commodity has positive feedback. SVM builds classification model with commodity data set $\{(x_j, y_j) | j = 1, 2, ...n\}$ to find optimal hyperplane $g(x) = \langle w \cdot x \rangle + b = 0$. The corresponding optimization of SVM in the experiment is:

$$\max_{\alpha} L(w, b, \alpha) = \sum_{j=1}^{n} \alpha_j - \frac{1}{2} \sum_{j=1}^{n} \alpha_i \alpha_q y_j y_q K(x_j x_q) \, s.t. \sum_{j=1}^{n} y_j \alpha_j = 0; 0 \leq \alpha_j \leq C \qquad (1)$$

$k(\cdot)$ is the radial basis function; optimal solution α_j^* can be obtained by optimization in formula (3-1) and thus figuring out the solution to the original question. $w^* = \sum_{j=1}^{n} \alpha_j^* y_j x_j$. Therefore, the classification decision function of optimal hyperplane definition can be expressed

Table 1: Data of commodity information

No.	Name	Price	Mark	Characteristic Values
1	Vero Moda 2018 autumn new chic knitwear	599	4.9	comfortable (179) pretty (134) I like it (85) I haven't wear it (59) good color (42) standard size (39)
2	Lin Shi Mu Ye cloth sofa bed	5960	4.9	skilled installation (1394) not bad quality (888) fast logistics (484) cost-effective (363) high price/performance ratio (104) no color difference (42) bad quality (6)
3	loose jeans for man in fall and winter	468	4.8	great quality (6681) comfortable (4132) suitable size (2816) good service (2585) good to wear (2366) cheap and fine (1996) thin cloth (613)
4	NTMPBINS suitcase with universal wheel	1658	4.8	good quality (721) good service (273) good style (270) fast logistics (222) no color difference (195) smooth wheels (122) fair quality (51)
5	excerpts from Historical Records with annotations	298	4.9	not bad (79) clear printing (61) good quality (54) cost-effective (50) thick paper (49) fast logistics (40)
6	2018 autumn new black dress with paillettes and tassels	513	4.7	good quality (193) beautiful (192) brighten your skin (192) soft cloth (153) non-deformation (135) new style (127) fair quality (39)

as below:

$$f\left(x\right) = sgn\left(g\left(x\right)\right) = sgn\left(\sum_{j=1}^{n} \alpha_j^* y_j k\left(x_j, x\right) + b^*\right) \qquad (2)$$

According to formula (2), all commodities can be divided into two categories. When $f(x) = -1$, the commodity has negative feedback; when $f(x) = +1$, the commodity has positive feedback. Commodities fall into two categories through the classification of SVM. Then the data representing users' dislike are eliminated and only data representing users' affection are reserved.

3.3   Comprehensive grade calculation

After the classification with SVM, the paper in this part will only make sentimental analysis of commodities with positive feedback and obtain quantified sentimental intensity, and then solve sentimental intensity after sentimental analysis and commodity marks with the method of weighted average, finally obtaining comprehensive grades. Sentimental matching algorithm is adopted to match commodity comments with the ontology base and work out corresponding sentimental intensity. Steps for computing the sentimental intensity of commodity comments are as follows: Firstly, normalize the word frequency of characteristic values in commodity comments as showed in Table 2.

Table 2: Normalization of word frequency of commodity characteristic values

ID	1	2	3	4	5	6	7	8	9
Description	good stuff	good service	fast logistics	good content	official edition	good packaging	clear printing	high-quality paper	fair quality
Word Frequency	480	207	129	117	86	84	72	70	34
Normalization	0.3752	0.1618	0.1008	0.0914	0.0672	0.0656	0.0562	0.0547	0.0265

Next, use sentimental words matching algorithm to match commodity characteristic values with ontology base to obtain sentimental intensity and polarity of the corresponding characteristic value, and the formula to calculate the comprehensive sentimental intensity of a commodity's comments is:

$$intensity\,(f_i) = K_1 P\,(w_{i1})\,T\,(w_{i1}) + ... + K_n P\,(w_{in})\,T\,(w_{in}) \tag{3}$$

w_{ij} represents sentimental words in commodity comments; $P\,(w_{i1})$ and $T\,(w_{i1})$ represent sentimental intensity and tendency of sentimental words respectively. When the polarity of $w_{ij} = 1$, $T\,(w_{i1}) = 1$. When the polarity of $w_{ij} = 0$,$T\,(w_{i1}) = 0$. When the polarity of $w_{ij} = 2$, $T\,(w_{i1}) = -1$. k_n is the corresponding normalized value of the characteristic value n . Finally, calculate the comprehensive grade of the commodity with formula (4) $\alpha = 5$; g_i is the mark of the commodity:

$$CE = \alpha f_i + (1 - \alpha)\,g_i \tag{4}$$

3.4   Similarity calculation and recommendation

The data processed in Table 3 are saved in matrix $R\,(U, I)$. U represents the number of users in the recommender system; I represents the number of items in the recommender system; r_{ij} is the mark of item I_j given by user u_i , representing users' affection for the item.w_{ij} is the similarity between item I_i and item I_j . Cosine similarity is adopted in the paper to measure the similarity between users or items. It measures similarity by the included angle between vector quantities.

As it does not take users' different rating scales into consideration, the cosine similarity in the modeling is improved by deducting users' average mark $\overline{r_u}$. Similarity w_{ij} between item I_i

Table 3: Example of comprehensive grade calculation-programming Python-from introduction to practice

Comment	good stuff	good service	fast logistics	good content	official edition	good packaging	clear printing	high-quality paper	fair quality	
Sentimental Classification	PH	PH	PA	PH	PH	PH	PH	PH	NN	
Intensity	3	7	7	5	4	6	5	3	9	
Polarity	1	1	1	1	1	1	1	1	2	
Assistant Sentimental Classification	0	0	PH	0	0	0	0	0	0	
Intensity	0	0	3	0	0	0	0	0	0	
Polarity	0	0	1	0	0	0	0	0	0	
Frequency	0.3752	0.1618	0.1008	0.0914	0.0672	0.0656	0.0562	0.0574	0.0265	
Sentimental Mark	1.1256	0.1326	0.46368	0.457	0.2688	0.3936	0.281	0.1641	-0.477	
comprehensive sentimental intensity										3.80938

and item I_j can be expressed as

$$w_{ij} = \frac{\sum_{u \epsilon U(i,j)}(r_{ui} - \overline{r_u})(r_{uj} - \overline{r_u})}{\sqrt{\sum_{u \epsilon U(i)}(r_{ui} - \overline{r_u})^2}\sqrt{\sum_{u \epsilon U(j)}(r_{uj} - \overline{r_u})^2}}. \tag{5}$$

Comprehensive grades of some commodities and users are firstly calculated with the formula in (3), as showed in Table 4. Then the similarity between commodities is calculated according to formula (5), and part of the results are demonstrated in Table 5.

Table 4: Comprehensive grades of some commodities

commodity|user	user1	user2	user3	user4	user5
Python** 1	4.57	2.18	3.18	0.60	3.56
suit*** 2	3.35	1.88	0.00	4.58	4.92
children's shoes**** 3	1.88	0.00	2.10	4.24	0.67
Anchor**** 4	0.00	1.50	3.94	0.00	1.38
nuts*** 5	1.16	2.96	1.58	4.14	4.85

SVM-based collaborative filtering obtains k nearest neighbors of the item to be recommended by calculating the similarity between items, and then use item similarity and users' records to figure out the popularity grade of the item to be recommended by weighted calculation, finally forming a recommendation list according to the rank of grades. User u's grade on item I_j can be expressed as:

$$P_{uj} = \sum_{I_i \epsilon I_u \bigcap S(I_j,k)} w_{ji}r_{ui} \tag{6}$$

Table 5: Similarity between commodities

	Similarity
Python**,suit***	-0.15
Python**,children's shoes****	0.29
Python**,Anchor****	-0.50
Python**,nuts***	0.15
suit***,children's shoes****	0.21
suit***,Anchor****	-0.80
suit***,nuts***	0.70
children's shoes****,Anchor****	-0.28
children's shoes****,nuts***	0.02
Anchor****,nuts***	-0.26

L_u represents the set of items user u like in the records; $S(I_j, k)$ represents the "k" item sets that are most similar to I_j . SVM is used to build positive-feedback set and negative-feedback set $L_i = \{L_i | r_{ij} = 1\}$ is the positive-feedback set; $DL_i = \{L_i | r_{ij} = -1\}$ is the negative-feedback set; $B_i = \{L_i | r_{ij} = 0\}$ is the unselected set.

In SVM-based collaborative filtering, the training set $X = \{(x_i, y_i) | x_i \epsilon L_i \bigcup DL_i, y_i \epsilon \{-1, 1\}\}$, test set $TX = \{tx_i \epsilon B_i\}$ .y_i is the classification tag of x_i; when $x_i \epsilon L_i$; $y_i = 1$ when $x_i \epsilon DL_i$, $y_i = -1$. The main steps of SVM-based collaborative filtering are as follows: Initialization: training set $X = \Phi$, test set $TX = \Phi$,

- use Pytho-based Scrapy to acquire commodity information on Taobao.

- use SVM to divide commodities into positive-feedback commodities and negative-feedback commodities, and build training set X and test set TX .

- use training set X to train SVM classifier.

- use classifier to classify test set, filtering out negative items and reserving positive items.

- make weighted calculation of marks and comments of positive-feedback commodities to get comprehensive grades.

- use $f(x)$ to predict the popularity grades of the items, and rank them according to the predicted grades, forming the final recommendation list.

4   Experiment results and analysis

4.1   Data preparation

The experiment adopts Python-based Scrapy to acquire data of online commodity information on Taobao, which include 7 categories of commodities (clothing, books, appliances, digital products, mobile phones, shoes and bags) and about 34,000 pieces of detailed comments. There are 4000 pieces of data for each category, among which 2500 pieces are taken as the training set and the rest are used for testing. Hardware : Thinkpad E445, 3.3GHZ, 4GB RAM.

4.2   Assessment indicator

Predictive accuracy P represents the probability that the user may like an item in the recommendation list, which can show the accuracy of the recommender system. The formula to calculate predictive accuracy of recommender system is as follow:

$$P = \frac{1}{m}\sum_{u=1}^{m} P_u = \frac{1}{m}\sum_{u=1}^{m} \frac{|RL_u \bigcap TL_u|}{N} \qquad (7)$$

Recall rate R represents the proportion of items users like in the recommendation list, which can show users' satisfaction degree with the recommendation results. The higher the recall rate is, the higher satisfaction degree users have.

The formula to calculate the recall rate of recommender system is as follow:

$$R = \frac{1}{m}\sum_{u=1}^{m} R_u = \frac{1}{m}\sum_{u=1}^{m} \frac{|RL_u \bigcap TL_u|}{TL_u} \qquad (8)$$

F–Measure is used to assess the overall recommending performance of the algorithm.

A larger F-Measure means the stronger recommending ability of the algorithm. The formula to calculate F-Measure of the recommender system is as follow:

$$F = \frac{1}{m}\sum_{u=1}^{m} F_u = \frac{1}{m}\sum_{u=1}^{m} \frac{2*P_u*R_u}{P_u + R_u} \qquad (9)$$

RL_u is user u's recommendation item set; TL_u is the set of items user u like in the test set; "U" is the recommended item number; $RL_u = N$. P, R and F are three main indicators to assess the algorithm. A larger F factor means the stronger recommending ability of the algorithm.

4.3   Results and analysis

(1)*Comparative analysis with traditional recommendation algorithm*

For the fairness and objectivity of the experiment, data recommendation in the paper is carried out according to traditional item-based recommendation algorithm and SVM-based collaborative filtering respectively, so as to compare the effect of recommendation. Traditional item-based recommendation algorithm is on the basis of neighborhood k . In order to be fairer and more objective, the recommended performance of algorithm with different neighborhood k will be analyzed in this part to choose the optimal k value. Figure 1 is the recommended performance of item-based recommendation algorithm with different k values. Three lines in the figure represents predictive accuracy, recall rate and F-Measure with different k values. It can be seen from the figure that the optimal recommended performance exists when $k = 10$, so $k = 10$ is selected in the subsequent experiment.

Figure 2, 3 and 4 represent the variation tendency of predictive accuracy P , recall rate R and F-Measure of SVM-based collaborative filtering and traditional recommendation algorithm respectively when the recommended item number N is different. It can be seen from the figures that when $N < 20$, predictive accuracy P , recall rate R and F-Measure of SVM-based collaborative filtering are obviously better than those of traditional recommendation algorithm. Therefore, SVM-based collaborative filtering has a big advantage over traditional recommendation algorithm when $N < 20$, especially when $N = 5$.

In real situation of recommending, due to users' limited time and energy, too much recommendation is meaningless to users, and will affect their shopping experience on the contrary. Therefore, what should be considered is the situation when N is relatively small. When $N < 15$
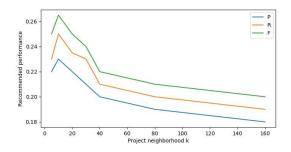
Figure 1: Recommended performance of item-based recommendation algorithm with different k values
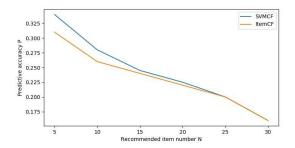

Figure 2: Predictive accuracy P of two models with different N values

, the recommended performance of SVM-based collaborative filtering is better than that of traditional recommendation algorithm, thus illustrating that when N is relatively small, SVM-based collaborative filtering can achieve better recommended performance.

To compare the efficiency of the two algorithms, during the performance comparison of traditional recommendation algorithm and SVM-based collaborative filtering in the paper, the experiment is divided into training stage and predication stage, and then time consumption of the algorithm in data set ML-100K is calculated respectively. Training time of traditional recommendation algorithm and SVM-based collaborative filtering is 33.89 seconds and 11.23 seconds respectively; test time of them is 250.01 seconds and 90.08 seconds respectively, thus demonstrating SVM-based collaborative filtering is more efficient.

(2) *Analysis of the recommended Performance of improved collaborative filtering*

Table 6 and Figure 5 are the recommended performance of SVM-based collaborative filtering
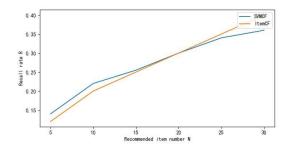

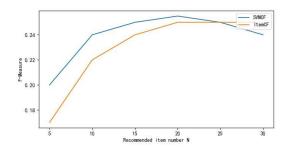Figure 3: Recall rate R of two models with different N values

Figure 4: F-Measure of two models with different N values

with different N (number of different recommended items).

Table 6: Recommended performance of SVM-based collaborative filtering with different N

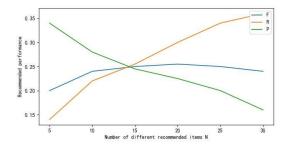| N (number of different recommended items)	5	10	15	20	25	30
F	0.2	0.24	0.25	0.255	0.25	0.24
R	0.14	0.22	0.255	0.3	0.34	0.36
P	0.34	0.28	0.245	0.225	0.2	0.16

Figure 5: Recommended performance of SVM-based collaborative filtering with different N

It can be seen from the table and figure that with the increase of N, predictive accuracy P starts to decline and eventually levels off; recall rate R starts to increase dramatically; F-Measure shows the trend of increasing first and then decreasing, and finally levels off. These phenomena indicate that recommended performance does not always have a positive correlation with N. Therefore, an appropriate N should be selected to improve the recommended performance of SVM-based collaborative filtering. When $N = 15$, F and P have relatively high values, which demonstrates that the recommender system has the best effect at this point, and too many items for recommendation will be a burden for users on the contrary. Therefore, the algorithm in the paper has the optimal overall recommended performance when $N = 15$.

5 Conclusion

The efficiency of recommending commodities to users in the E-commerce industry is decided by the scale of recommended items and the performance of the recommendation algorithm. In the paper, SVM is adopted at first to classify items, and then items users may dislike are filtered out by negative-feedback information to reduce the scale of recommended items, which can not only significantly increase recommendation efficiency, but also decrease the disturbance of these

items on recommendation and promote recommendation precision. After figuring out positive-feedback commodities, marks and comments are taken into consideration to obtain comprehensive grades by weighted average, thus making it more objective and indirectly improving precision. Finally, verification is conducted with the online data in the E-commerce industry. Therefore, the algorithm is of certain practical value for the E-commerce industry. Further studies can be carried out in the future with regard to how to further improve recommendation precision and efficiency of the recommendation algorithm in the situation of more recommended items.

Funding

Author contributions

The authors contributed equally to this work.

Conflict of interest

The authors declare no conflict of interest.

Bibliography

[1] Ahmad, A.S.; Hassan, M.Y.; Abdullah, M.P. et al. (2014). A review on applications of ANN and SVM for building electrical energy consumption forecasting, *Renewable and Sustainable Energy Reviews*, 33, 102-109, 2014.

[2] Barbieri, N. (2013). An Analysis of Probabilistic Methods for Top-N Recommendation in Collaborative Filtering, *Machine Learning & Knowledge Discovery in Databases-European Conference*, DBLP, 2013.

[3] Cheng, Q; Wang X; Yin, D. et al. (2015); The New Similarity Measure Based on User Preference Models for Collaborative Filtering, *IEEE International Conference on Information & Automation*, IEEE, 2015.

[4] Chung, Y; Jung, H.W.; Kim, J. et al. (2013). Personalized Expert-Based Recommender System: Training C-SVM for Personalized Expert Identification, *International Workshop on Machine Learning and Data Mining in Pattern Recognition*, Springer, Berlin, Heidelberg, 2013.

[5] Du, Y,-P.; Yao, C.-Q.; Huo, S.-H. et al. (2017). A new item-based deep network structure using a restricted Boltzmann machine for collaborative filtering, *Frontiers of Information Technology & Electronic Engineering*, 18(05), 658-666, 2017.

[6] Goldberg, D.; Nichols, D.; Oki, B.M. et al. (1992). Using Collaborative Filtering to Weave an Information Tapestry, *Communications of the ACM*, 35(12),61-70,1992.

[7] Guo, G.; Zhang, J.; Thalmann, D. (1992). Merging trust in collaborative filtering to alleviate data sparsity and cold start, *Knowledge-Based Systems*, 35, 57-68, 2014.

[8] Hu, Y.; Peng, Q.; Hu, X. et al. (1992). Time Aware and Data Sparsity Tolerant Web Service Recommendation Based on Improved Collaborative Filtering, *IEEE Transactions on Services Computing*, 8(5), 782-794, 2015.

[9] Jindal, A.; Dua, A.; Kaur, K. et al. (2016). Decision Tree and SVM-Based Data Analytics for Theft Detection in Smart Grid, *IEEE Transactions on Industrial Informatics*, 12(3), 1005-1016, 2016.

[10] Li, G.; Ou, W. (2016). Pairwise probabilistic matrix factorization for implicitfeedback collaborative filtering, *Neurocomputing*, 204, 17-25, 2016.

[11] Li, H.; Hong, R.; Lian, D. et al. (2016). A Relaxed Ranking-Based Factor Model for Recommender System from Implicit Feedback, *IJCAI*, 1683-1689, 2016.

[12] Li, Z.; Peng, J.Y.; Geng, G.H. et al. (2015). Video recommendation based on multi-modal information and multiple kernel, *Multimedia Tools and Applications*, 74(13), 4599-4616, 2015.

[13] Liu, X. (2017). A collaborative filtering recommendation algorithm based on the influence sets of e-learning group's behavior, *Cluster Computing*, 1–11, 2017.

[14] Madadipouya, K. (2015). A Location-Based Movie Recommender System Using Collaborative Filtering, *Computer Science*, 5, 2015.

[15] Manek, A.S.; Shenoy, P.D.; Mohan, M.C. et al. (2017). Aspect term extraction for sentiment analysis in large movie reviews using Gini Index feature selection method and SVM classifier, *World Wide Web*, 20, 135-154, 2017.

[16] Nilashi, M.; Ibrahim, O.B.; Ithnin, N. et al. (2015); A multi-criteria recommendation system using dimensionality reduction and Neuro-Fuzzy techniques, *Soft Computing*, 19(11), 3173-3207, 2015.

[17] Nilashi, M.; Ibrahim, O.B.; Ithnin, N. (2014). Multi-criteria collaborative filtering with high accuracy using higher order singular value decomposition and Neuro-Fuzzy system, *Knowledge-Based Systems*, 60(2), 82-101, 2014.

[18] Nasiri, M.; Minaei, B. (2016). Increasing prediction accuracy in collaborative filtering with initialized factor matrices, *Journal of Supercomputing*, 72(6), 2157-2169, 2016.

[19] Ren, L.; Wang, W. (2017). An SVM-based collaborative filtering approach for Top-N web services recommendation, *Future Generation Computer Systems*, S0167739X17300389, 2017.

[20] Sedhain, S.; Sanner, S.; Braziunas, D. et al. (2014). Social collaborative filtering for cold-start recommendations, 345-348,2014.

[21] Selakov, A.; Cvijetinovi, D.; Milovi, L. et al. (2014). Hybrid PSO–SVM method for short-term load forecasting during periods with significant temperature variations in city of Burbank, *Applied Soft Computing*, 16, 80-88, 2014.

[22] Su, H.; Lin, X.; Yan, B. et al. (2015). The Collaborative Filtering Algorithm with Time Weight Based on Map Reduce, *International Conference on Big Data Computing & Communications*, Springer, Cham, 2015.

[23] Uricar, M.; Timofte, R.; Rothe, R. et al. (2016); Structured Output SVM Prediction of Apparent Age, Gender and Smile From Deep Features, *2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016.

[24] Wang, Z.; Liu, Y.; Chiu, S. (2016). An efficient parallel collaborative filtering algorithm on multi-GPU platform, *The Journal of Supercomputing*, 72(6), 2080-2094, 2016.

[25] Wei, J.; He, J.; Chen, K. et al. (2017); Collaborative filtering and deep learning based recommendation system for cold start items, *Expert Systems with Applications*, 69,29-39,2017.

[26] Yagci, A.M.; Aytekin, T.; Gurgen, F.S. (2017). Scalable and adaptive collaborative filtering by mining frequent item co-occurrences in a user feedback stream, *Engineering Applications of Artificial Intelligenceert Systems with Applications*, 58,2017.

[27] Zhang, F.; Gong, T.; Lee V.E. et al. (2016). Fast algorithms to evaluate collaborative filtering recommender systems, *Knowledge-Based Systems*, 96(C), 96-103, 2016.

[28] Zhang, D.W.; Xu, H.; Su, Z. et al. (2015). Chinese comments sentiment classification based on word2vec and SVM perf, *Expert Systems with Applications*, 42(4), 1857-1863, 2015.

[29] Zhao, P.X.; Gao, W.; Han, X. et al. (2019). Bi-objective collaborative scheduling optimization of airport ferry vehicle and tractor, *International Journal of Simulation Modelling*, 18(2), 355-365,2019.

[30] Zhao, P.X.; Luo, W.H.; Han, X. (2019). Time-dependent and bi-objective vehicle routing problem with time windows, *Advances in Production Engineering & Management*, 14(2), 201-212,2019.

[31] Zhou, W.; Wen, J.; Gao, M. et al. (2015). A Shilling Attack Detection Method Based on SVM and Target Item Analysis in Collaborative Filtering Recommender Systems, *International Conference on Knowledge Science*, 2015.

[32] [Online].http://www.askci.com/reports/20180201/0946472814827719.shtml.