Sustainable Cities and Society 76 (2022) 103373

Contents lists available at ScienceDirect

Sustainable Cities and Society

journal homepage: www.elsevier.com/locate/scs

Three-way Naive Bayesian collaborative filtering recommendation model for smart city

Chunying Zhang [a,b], Xueming Duan [a], Fengchun Liu [c,*], Xiaoqi Li [a], Shouyue Liu [a]

[a] College of Science, North China University of Science and Technology, Tangshan Hebei, China
[b] Key Laboratory of Data Science and Application of Hebei Province, Tangshan, China
[c] College of Qian 'an, North China University of Science and Technology, Tangshan Hebei, China

ARTICLE INFO

Keywords:
Smart city
Three-way decisions
Naive Bayesian
Collaborative filtering
Recommendation system

ABSTRACT

Smart city construction has penetrated all aspects of human life. People can continue to feel the convenience of life brought by new technologies and new services. In order to further optimize the smart city system, aiming at the sparsity of scoring data and the difficulty of two-way decisions to deal with uncertain decisions in the recommendation algorithm, a three-way Naive Bayesian Collaborative Filtering Recommendation model (3NBCFR) for smart city is constructed by integrating Naive Bayesian, three-way decisions and collaborative filtering algorithm. Firstly, we consider the influence of item attributes on user scoring. Naive Bayesian classifier is used to score unrated items, which can effectively fill in the missing values in the score matrix and solve the problem of data sparsity. Secondly, to ensure sustainable recommendations for uncertain goals, the three-way decisions is introduced into the collaborative filtering recommendation system, three-way recommendation rules are formulated, and 3NBCFR is constructed. Finally, the model is applied to the movie recommendation system. The experiment on Movielens shows that compared with the traditional collaborative filtering recommendation algorithm, 3NBCFR algorithm reduces the recommendation cost and improves the recommendation quality. At the same time, it lays a foundation for promoting the construction of smart city.

1. Introduction

Smart city makes use of information and communication technologies to analyze and integrate the key information of the core system of urban operation, respond intelligently to the needs of people's livelihood, and create a better urban life for human beings. Intelligent algorithm is a key technology for the construction of smart city, which has been successfully applied in metallurgy ore blending (Yang et al., 2021), surface roughness prediction (Pan, Wang, Zhou, Yan, & Guo, 2020), drug positioning (Yan, Yang, Kong, Bai, & Li, 2021) and other manufacturing fields. Smart city has also been extensively studied in urban services, such as intelligent monitoring (Wang, Wang, Kumari, Yeh, & Chen, 2020), intelligent distribution (Wang et al., 2021), and urban logistics (Filip, Dominika, & Jozef, 2021). With the development of economy, people's demand for entertainment is increasing. At present, more and more scholars study the needs of people's spiritual life. For example, Gao et al. (2020) considered different types of social media environments and presented a new place recommendation algorithm which is designed based on latent factor model to make contributions to the people's tourism industry. Tang, Tan, Hu, and Geng (2018) discussed the evaluation method of urban Wi-Fi facility spatial service and layout efficiency in smart city network infrastructure planning, which could provide help for People's Daily network needs. Watching movies is also an important way of entertainment. Studying excellent recommendation algorithms to provide people with appropriate and high-quality movies is a part of meeting people's livelihood needs.

To a certain extent, the level of intelligent recommendation service determines the degree of satisfaction of the people in their spiritual life. At present, the research of intelligent recommendation algorithm has been very extensive. For example, Xu et al (2020) proposed a recommendation based on deep user multimodal preference method to capture the recommendation of users and items matched by text and vision. Li and Gong (2020) proposed a new cross-city POI recommendation deep neural network, which integrated deep neural network, transfer learning technology and density-based resampling method into a unified framework. The above recommended algorithms have good results in their respective application fields, but the algorithm process is complex and the learning cost is high. The collaborative filtering (CF) algorithm is

* Corresponding author.
E-mail address: lnobliu@ncst.edu.cn (F. Liu).

https://doi.org/10.1016/j.scs.2021.103373
Received 17 May 2021; Received in revised form 7 September 2021; Accepted 17 September 2021
Available online 29 September 2021

simple and can filter information that is difficult to automatically analyze based on content, such as artwork, music, etc. Therefore, the collaborative filtering algorithm can be lightly applied to the smart recommendation service of the smart city, so that people can enjoy the convenience brought by the smart city.

Collaborative filtering is the most common recommendation algorithm, which was proposed by Goldberg, Nichols, Oki, and Terry, (1992). The idea is to calculate the list of neighbor users based on the score matrix, predict the favorite items of users, and recommend the items with high ratings to users. Collaborative filtering algorithm has achieved great success in the recommendation systems. In the actual recommended scenario, the quality and reliability of recommendation are reduced if the user only evaluates a few items, the score matrix obtained at this time is relatively sparse when the user and the project scale are huge. Scholars have proposed some solutions for the sparsity of scoring data. Ren, Zhang, and Zhang (2020) used the attribute information of users or items and the user-item score matrix to construct a decision table, and used rough set theory to extract rules to alleviate the problem of data sparsity. Zheng, Hu, Yang, and Yj (2019) classified users firstly, and then used rough set attribute reduction methods to eliminate some items, which solved the problem of sparsity to a certain extent. Since rough sets can process uncertain information, the above methods can effectively alleviate the problem of data sparsity. Naive Bayesian (NB) classifier is a classification method using probability and statistics knowledge. It mainly utilizes Bayesian' theorem to predict the maximum possible category of samples or objects. It can not only deal with uncertain problems, but also is simple and efficient, with strong probability expression ability. Therefore, compared with rough sets, it is more effective to use Naive Bayesian to solve the sparsity problem of score data.

In practical problems, insufficient and inaccurate information will lead to the uncertainty of recommendation. For recommendation problems, if only two strategies, namely recommendation and non-recommendation, are considered, the model may fall into difficulties, leading to recommendation errors and unsustainable recommendation. Three-way decisions is a method to deal with uncertain decisions. By introducing delayed decision, the decision domain is divided into three parts, which can continuously deal with uncertain recommendation results and effectively alleviate the mechanization problem of traditional two-way recommendation algorithm. Therefore, adding three-way decisions into collaborative filtering can effectively solve the problem of low recommendation efficiency.

In order to solve the problem of sparsity of score data and uncertain information cannot be solved by two-way decisions. First, according to the impact of item attribute characteristics on user ratings, the Naive Bayesian classifier is used to score unrated items, and the missing values in the score matrix are filled in to alleviate the problem of data sparseness. Secondly, three-way decisions is introduced into collaborative filtering recommendation system, construct three-way Naive Bayesian recommendation models. Finally, in order to ensure that the algorithm is suitable for smart city construction, the three-way Naive Bayesian collaborative filtering recommendation model (3NBCFR) is applied to movie recommendation, so that people can enjoy the convenience of technology while enjoying life.

The contributions of this paper are as follows,

(1) In view of the sparseness of scoring data, the existing scoring information and item attributes are analyzed, and the Naive Bayesian is applied to the data filling of the score matrix, and a score prediction model based on the Naive Bayesian classifier is constructed. It makes full use of the information contained in the score matrix to fill in the scoring data reasonably and efficiently.
(2) According to the minimum risk decision process of Bayesian decision theory, three-way recommendation cost matrix is established and three-way recommendation rules are constructed. It solves the problem that the two-way decisions cannot deal with the uncertain decision, so that the decision is more in line with the way of human thinking, and the recommendation behavior is more intelligent.
(3) The score prediction model based on the Naive Bayesian classifier and the three-way recommendation rules are merged to construct 3NBCFR. Experiments show that the model has higher recommendation quality and lower recommendation cost and it is beneficial to the construction of smart city.

The structure of this paper is as follows. The Section 2 introduces the related works of the collaborative filtering recommendation algorithm and the three-way decisions theory. In Section 3, we use the Naive Bayesian classifier to score the unrated items, fill in the missing values in the score matrix, and build a scoring prediction model based on the Naive Bayesian. Section 4 introduces three-way decisions into the collaborative filtering recommendation system, formulates three-way recommendation rules, and builds 3NBCFR. In Section 5, we apply 3NBCFR to the movie recommendation system to verify the effect on recommendation quality and recommendation cost. Finally, the content of this paper is summarized and the future works are prospected.

2. Related works

2.1. Smart city

With the development of the economy, information technology and innovative concepts are used all over the world to connect and integrate urban systems and services, improve the efficiency of resource utilization, optimize urban management and services, and accelerate the sustainable development of cities, so as to improve the quality of life of citizens and build smart city. Smart city has penetrated human life. Fong, Aghamohammadi, Ramakreshnan, Sulaiman, & Mohammadi, 2019 summarized the outdoor thermal comfort conditions in the existing tropical marine environment, improved the shortcomings, and promoted the sustainable livability of cities in tropical regions. Nathan & Owusu, 2020 analyzed the concept of smart houses based on the socioeconomic characteristics of Sub-Saharan Africa and discussed how to adopt these houses sustainably in the region. In order to solve the problem of low level of traditional air pollution prediction methods, Liu and Zhang (2021) used long-term and short-term memory (LSTM) to evaluate the air pollution quality prediction in smart city, and proposed Staked Auto-Encoder model (LSTM-SAE) assisted by long-term and short-term memory. In the construction of smart city, recommendation algorithm is an important algorithm, which is an indispensable part of intelligent construction of smart city. For example, Soheil, Haneen, & Dominique, 2020 analyzed the negative effects brought by the control of autonomous driving vehicle system, and put forward some recommendations to optimize the urban traffic system. Romain, Siddharth, & Peter, 2020 studied the transmission of COVID-19, and the Swedish government chose a recommendation-based approach to implement some restriction to ensure the safety of citizens and improve the capacity of public services in the city.

2.2. Collaborative filtering

Collaborative filtering recommendation algorithm is currently one of the most mature and popular personalized recommendation technologies. It has been successfully applied to movie recommendation (Zhang et al., 2019), service recommendation (Yao, Wang, Li, & Rodrigues 2021 ; Zhang, Yin, Wu, He, & Zhu 2020) and other fields. Data sparsity is a common problem in collaborative filtering. Faced with the problem of data sparsity, there are many different solutions, such as the filling directly, which fills the sparse matrix according to the weighted average, mode, and median of the rows and columns of the user-item matrix to improve the sparsity of data (Breese, Hecherman, & Kadie, 1998). The new similarity method: making full use of various useful information contained in user rating data to calculate the similarity between users or items (Choi & Suh, 2013). The content-based method: by establishing user description files and item content files, match the content of the items that best match the user's description to the user (Gao, Qi, Liu, & Liu 2008).

The above methods seem to be able to solve the problem of data sparsity, but there are problems such as large prediction deviations and limited use methods. In order to effectively solve the problem of data sparsity, many scholars have done further research. For example, Wang and Liu (2020) preprocessed the original score matrix according to the user's rating difference. They constructed the user category preference matrix from the obtained user item score matrix and item type matrix, reflecting the user's interest preferences, and alleviating the sparseness of the data. Finally, a collaborative filtering recommendation algorithm with optimized clustering is proposed. Huynh, Phan, Pham, Pham, and Ismail, (2020) alleviated the problem of data sparseness by constructing the homologous relationship between users or items calculated by context attributes, and they put forward a collaborative filtering recommendation based on context similarity.

The above papers study the problem of data sparsity in collaborative filtering is a two-way decisions in a deterministic environment. When problems such as insufficient knowledge and insufficient data lead to uncertain decisions, the two-way decisions cannot make accurate recommendations. Three-way decisions is a tool to deal with uncertainty problems, people introduce a delayed decision, the universe divided into three and the decision to continue. Therefore, it is necessary to join the three-way decisions in collaborative filtering.

2.3. Three-way decisions

Three-way decisions is a tool proposed by Yao Yiyu to deal with uncertain information (Yao, 2012). Its main idea is to divide the domain of discourse into three, and to formulate different decision strategies for each domain, which conforms to human thinking and cognition. Three-way decisions is an extension of the traditional two-way decisions. Taking into account the uncertain factors in the decision process, the three-way decisions regard non-commitment decision or delayed decision as the third decision behavior when the information is insufficient. The three-way decisions has been widely used in time scheduling (Li, Tian, Chen, & Liang, 2020), complex networks (Yang & Li, 2020), combinatorial optimization (Zhang, Cheng, Zhao, Wang, & Xia, 2021), dynamic recommendation (Liu & Ye, 2020) and other fields.

In the 1990s, Yao Yiyu proposed an extended model of probabilistic rough set (i.e., decision-theoretic rough sets) (Yao & Wong, 1992) to solve the problem of the lack of fault tolerance and risk cost sensitivity in the classical rough set theory to deal with the classification and decision problems. He introduced the Bayesian minimum risk decision process, and provided a set of numerical calculation method system based on the decision risk cost for the determination of the probability threshold in the rough set of decision (Yao, 2011). In addition, he explained that the three-way decisions model based on probabilistic rough set is better than the two-way decisions model and the three-way decisions model based on classical rough set under any decision conditions (Yao, 2010, 2009). Jackowski (2018) provided a loss function characterized by conditional risk, loss function and minimum risk decision rule based on Bayesian decision theory. Setting the loss function as the evaluation function of three-way decisions to construct a three-way decisions model based on rough set of decision theory. Hu, Zhang, and Yu, (2019) chose the uncertainty measurement method based on margin strategy as the evaluation function to measure the uncertainty of unlabeled samples, and then divided the unlabeled samples into three different domains: positive domain, negative domain, and boundary domain. Different samples in different regions were processed differently, and an active learning method based on three-way decisions theory was proposed. Xue, Wang, Liu, Zhu, and Xue, (2016) combined the probability map and built a Bayesian network to calculate the conditional probability distribution function, and proposed a three-way decisions model based on the probability map. They respectively integrated the set pair information granule and interval concept lattice with the three-way decisions theory, and respectively proposed a three-way decisions model based on the set pair information granule space and the concept of the interval three-way decisions space, and achieved good research results (Wang, Zhang, & Liu, 2016; Zhang, Wang, Li, & Liu, 2016).

Applying the three-way decisions to the recommendation algorithm can effectively increase accuracy. From the perspective of the three-way decisions space, the interval concept lattice theory and the three-way decisions theory were combined, and the interval three-way decisions space theory was proposed (Li, Chen, Liu, & Liu, 2020). In the interval three-way decisions space, the positive domain, negative domain, and boundary domain were divided by extending the concept of interval three-way decisions. Then, the decision loss function and three-way decisions rules were extracted. Finally, experiments to verify that the three-way recommendation models were suitable for product recommendation problems. In order to reduce the recommendation cost of the traditional collaborative filtering algorithm and solve the problem of single scoring information of the algorithm, a three-way granular recommendation algorithm based on collaborative filtering was proposed (Qin & Zhang, 2020). Zhang et al. (2017) applied the three-way decisions to the recommendation system and minimized the average cost by adjusting the thresholds of different behaviors, and a regression-based three-way recommendation system was proposed.

Aiming at the sparseness of data, in order to avoid problems such as high recommendation cost, inaccurate prediction results, and poor practicability caused by mechanically supplementing data, the existing scoring information and item attributes are analyzed, and the Naive Bayesian classifier is used to predict missing data in the score matrix. In order to deal with the uncertainty problem, three-way decisions is introduced, and the 3NBCFR is constructed.

3. Score prediction model based on Naive Bayesian

For the sparsity of user score, we use the existing scoring information and item attributes for each user to construct the Naive Bayesian classifier to predict the missing data in the score matrix R. In this section, we mainly introduce the knowledge of collaborative filtering and build a score prediction model based on Naive Bayesian.

3.1. Collaborative filtering

The user, item and scoring information in the recommendation system is the most basic data model. Let $U = \{u_1, u_2, \cdots, u_n\}$ be the user set, $V = \{v_1, v_2, \cdots, v_m\}$ be the item set, and the user's rating function for items is expressed as

$$R : U \times V \to V_R \cup \{0\} \tag{1}$$

In the formula, $R = (r_{ij})_{n\times m}$ is the score matrix of $n\times m$, n is the number of users, m is the number of items, r_{ij} is the rating of u_i on v_j, V_R is the set of ratings of the items by users, such as $V_R = \{1, 2, 3, 4, 5\}$.

The common measures used to calculate the similarity between users are shown below.

1) Cosine similarity (COS)

$$sim(i,j)^{COS} = \frac{\sum_{k\in V_{ij}} r_{ik} \times r_{jk}}{\sqrt{\sum_{k\in V_{ij}} r_{ik}^2} \times \sqrt{\sum_{k\in V_{ij}} r_{jk}^2}} \tag{2}$$

2) Adjusted cosine similarity (ADCOS)

$$sim(i,j)^{ADCOS} = \frac{\sum_{k\in V_{ij}}\left(r_{ik}-\overline{r}_k\right)\times\left(r_{jk}-\overline{r}_k\right)}{\sqrt{\sum_{k\in V_{ij}}\left(r_{ik}-\overline{r}_k\right)^2}\times\sqrt{\sum_{k\in V_{ij}}\left(r_{jk}-\overline{r}_k\right)^2}} \tag{3}$$

3) Pearson Correlation Coefficient (PCC)

$$sim(i,j)^{PCC} = \frac{\sum_{k\in V_{ij}}\left(r_{ik}-\overline{r}_i\right)\times\left(r_{jk}-\overline{r}_j\right)}{\sqrt{\sum_{k\in V_{ij}}\left(r_{ik}-\overline{r}_i\right)^2}\times\sqrt{\sum_{k\in V_{ij}}\left(r_{jk}-\overline{r}_j\right)^2}} \tag{4}$$

In the formulas, $sim(i,j)$ denotes the similarity between users u_i and u_j. V_{ij} denotes the set of items rated by users u_i and u_j. r_{ik} and r_{jk} denotes the score of users u_i and u_j on item v_k, respectively. $\overline{r_i}$, $\overline{r_j}$and $\overline{r_k}$ denote the average score of u_i, $u_j$and v_k respectively.

The formula (4) is used to calculate the similarity value between users. We select the first l neighbor users of the target user t_i and predict t_i's score on unrated item v_j.

$$r_{ij}^{'} = \overline{r}_i + \frac{\sum_{x=1}^{l} sim(i,x)\times\left(r_{xj}-\overline{r}\right)}{\sum_{x=1}^{l} sim(i,x)} \tag{5}$$

In the formula, $r_{ij}^{'}$ denotes the predicted score of t_i to v_j, r_{xj} denotes the score of the neighbor users l to v_j, $\overline{r_i}$ and $\overline{r_x}$ denotes the average score of users t_i with neighbor user u_x.

3.2. Score prediction model based on Naive Bayesian

In practical applications, the sparseness of the matrix continues to increase with the increasing data of users and items. Therefore, the Naive Bayesian classifier is used to score the items that have not been rated by users, and the matrix is filled.

Let φ be the scoring level threshold, and the preference matrix $\Delta = (\delta_{ij})_{n\times m}$ can be calculated according to the score matrix R, where

$$\delta_{ij} = \begin{cases} 1 & if\ r_{ij}\geq\varphi \\ 0 & otherwise \\ -1 & if\ 0<r_{ij}<\varphi \end{cases} \tag{6}$$

1, 0, −1 indicate like, unknown and dislike respectively. Set the scoring level threshold φ to 3, when $r_{ij}\geq 3$, $\delta = 1$, which means u_i likes v_j; when $0<r_{ij}<3$, $\delta = -1$, which means u_i doesn't like v_j; in other cases, $\delta = 0$, which means u_i does not score v_j.

Given a user set $U = \{u_1,u_2,\cdots,u_n\}$, an item set $V = \{v_1,v_2,\cdots,v_m\}$and an item attribute set $A = \{a_1,a_2,\cdots,a_h\}$. For each item $v_j = \{g_1^{(j)},g_2^{(j)},\cdots,g_h^{(j)}\}$, $g_i^{(j)}$ denotes the value of item v_j in attribute a_i. The user's historical rating data is used as training set, and a Naive Bayesian model is established for each user to predict the probability that the user likes unrated items. Given the score threshold φ, the items rated by the user are divided into two categories, that is like (X) and dislike $(\overline{X})$. Based on Bayes theorem, the posterior probability $P(X|v_j)$ can be written as

$$P\left(X|v_j\right) = \frac{P(X)P\left(v_j|X\right)}{P\left(v_j\right)} \tag{7}$$

In the formula, $P(X)$ is the prior probability, $P(v_j|X)$ is the class conditional probability, $P(v_j)$ is a constant. The expression of the Naive Bayesian model is

$$P\left(X|v_j\right)\propto P(X)\prod_{i=1}^{h}P\left(g_i^{(j)}|X\right) \tag{8}$$

In the formula, h denotes the number of attributes, $g_i^{(j)}$ denotes the value of v_j on the ith attribute a_i. In formula (10), the estimation formula of $P(X)$ and $P(v_j|X)$ is

$$P(X) = \frac{|X|}{|U|} \tag{9}$$

$$P\left(g_i^{(j)}|X\right) = \frac{\left|m\left(a_i,g_i^{(j)}\right)\cap X\right|}{|X|} \tag{10}$$

In the formula, $|X|$ is the number of items that the user likes, $|U|$ is the total number of items in the training set, $m(a_i,g_i^{(j)})\cap X$ denotes the set of items in X whose attribute a_i is $g_i^{(j)}$.

Using Laplace estimation, the prior probability $P(X)$ and class conditional probability $P(g_i^{(j)}\big|X)$ are

$$P(X) = \frac{1+|X|}{|\Theta|+|U|} \tag{11}$$

$$P\left(g_i^{(j)}|X\right) = \frac{1+\left|m\left(a_i,g_i^{(j)}\right)\cap X\right|}{|a_i|+|X|} \tag{12}$$

Where, Θ represents the set of user preference states, and $|\Theta|$ represents the number of user preferences.

Aiming at the sparseness of user ratings, a Naive Bayesian classifier is constructed to predict the missing data in the user-item score matrix for each user using the existing scoring information and item attributes. The detail of algorithm is as follows

Algorithm 1: Score prediction model based on Naive Bayesian
Input: Score matrix R, User set $U = \{u_1,u_2,\cdots,u_n\}$, Item set $V = \{v_1,v_2,\cdots,v_m\}$
Output: Filled score matrix R
1:Determine the category label of item v_j according to the score threshold φ
2:**for** $u_i\in U$ **do**
3: Build Naive Bayesian classifier
4: **for** $X\in\Theta$ **do**
5: **for** $a_j\in A$ **do**
6: calculate prior probability$P(X)$and class conditional probability$P(g_i^{(j)}\big|X)$
7: **end for**
8: **end for**
9: Classify the item v_j to be predicted and calculate the probability $P(X|v_j)$
10: Predict u_i score for v_j
11:**end for**
12:**return** Filled score matrix R

4. Three-way sustainable recommendation model

In order to reduce the recommendation cost of the traditional two-way recommendation model, a three-way sustainable recommendation model is established. In this section, the feasibility of the model will be analyzed theoretically from the three-way recommendation rules, algorithm ideas and time complexity.

4.1. Basic theory of three-way decisions

Given an information system $S = (U,A\cup D,\{V_a|a\in A\},\{I_a|a\in A\})$, where $U = \{x_1,x_2,\cdots,x_N\}$is a universe, A is an attribute set and $A = \{a_1,a_2,\cdots,a_n\}$. $D = \{C_1,C_2,\cdots,C_k\}$ denotes k disjoint decision classes. V_a is an attribute value set. $I_a : U\rightarrow V_a$ is an information function, which indicates that the attribute a in U is mapped to V_a, that is $I_a(x)\in V_a$.

Given an information system $S = (U,A\cup D,\{V_a|a\in A\},\{I_a|a\in A\})$, suppose $\Theta = \{C,\overline{C}\}$ denotes two states and $C\subseteqq U$, where C and $\overline{C}$ are complementary. The set $AC = \{a_p,a_N,a_B\}$ denotes three decision actions: acceptance, rejection and delay, that is $x\in POS_{(\alpha,\beta)}(C)$, $x\in NEG_{(\alpha,\beta)}(C)$

and $x \in BND_{(\alpha,\beta)}(C)$. The corresponding cost functions of different decision actions are shown in Table 1.

λ_{PP}, λ_{NP}, λ_{BP} respectively denote the cost of taking actions a_P, a_N, a_B when the object belongs to C. λ_{PN}, λ_{NN}, λ_{BN} respectively denote the cost of taking actions a_P, a_N, a_B when the object does not belong to C. $P(C|[x])$ denotes the conditional probability that $[x]$ belonging to the set C. For the objects in $[x]$, the expected cost of taking actions a_P, a_N, a_B respectively is

$$\begin{aligned} R(a_P|[x]) &= \lambda_{PP}P(C|[x]) + \lambda_{PN}P\left(\overline{C}\middle|[x]\right) \\ R(a_N|[x]) &= \lambda_{NP}P(C|[x]) + \lambda_{NN}P\left(\overline{C}\middle|[x]\right) \\ R(a_B|[x]) &= \lambda_{BP}P(C|[x]) + \lambda_{BN}P\left(\overline{C}\middle|[x]\right) \end{aligned} \tag{13}$$

According to the minimum risk decision process of Bayesian decision theory, the following minimum cost decision rules can be obtained (Zhang, Qiao, Wang, Liu, & Zhang, 2017).

(*P*) If $R(a_P|[x]) \leq R(a_N|[x])$ and $R(a_P|[x]) \leq R(a_B|[x])$, then $x \in POS_{(\alpha,\beta)}(C)$, accept decision.
(*N*) If $R(a_N|[x]) \leq R(a_P|[x])$ and $R(a_N|[x]) \leq R(a_B|[x])$, then $x \in NEG_{(\alpha,\beta)}(C)$, reject decision.
(*B*) If $R(a_B|[x]) \leq R(a_P|[x])$ and $R(a_B|[x]) \leq R(a_N|[x])$, then $x \in BND_{(\alpha,\beta)}(C)$, defer decision.

By $P(C|[x]) + P(\overline{C}\big|[x]) = 1$ and $\lambda_{PP} \leq \lambda_{BP} \leq \lambda_{NP}$, $\lambda_{NN} \leq \lambda_{BN} \leq \lambda_{PN}$, the minimum cost decision rules are simplified as

(*P*) If $P(C|[x]) \geq \alpha$ and $P(C|[x]) \geq \gamma$, then $x \in POS_{(\alpha,\beta)}(C)$, accept decision.
(*N*) If $P(C|[x]) \leq \beta$ and $P(C|[x]) \leq \gamma$, then $x \in NEG_{(\alpha,\beta)}(C)$, reject decision.
(*B*) If $P(C|[x]) \leq \alpha$ and $P(C|[x]) \geq \beta$, then $x \in BND_{(\alpha,\beta)}(C)$, defer decision.

Let

$$\alpha = \frac{(\lambda_{PN} - \lambda_{BN})}{(\lambda_{PN} - \lambda_{BN}) + (\lambda_{BP} - \lambda_{PP})}$$

$$\beta = \frac{(\lambda_{BN} - \lambda_{NN})}{(\lambda_{BN} - \lambda_{NN}) + (\lambda_{NP} - \lambda_{BP})} \tag{14}$$

$$\gamma = \frac{(\lambda_{PN} - \lambda_{NN})}{(\lambda_{PN} - \lambda_{NN}) + (\lambda_{NP} - \lambda_{PP})}$$

If we further assume that the cost functions satisfy $\frac{\lambda_{NP}-\lambda_{BP}}{\lambda_{BN}-\lambda_{NN}} > \frac{\lambda_{BP}-\lambda_{PP}}{\lambda_{PN}-\lambda_{BN}}$, we can prove $\alpha > \gamma > \beta$, the threshold γ is no longer needed, and the minimum cost decision rules are

(*P*) If $P(C|[x]) \geq \alpha$, then $x \in POS_{(\alpha,\beta)}(C)$, accept decision.
(*N*) If $P(C|[x]) \leq \beta$, then $x \in NEG_{(\alpha,\beta)}(C)$, reject decision.
(*B*) If $\beta < P(C|[x]) < \alpha$, then $x \in BND_{(\alpha,\beta)}(C)$, defer decision.

Where

$$\alpha = \frac{(\lambda_{PN} - \lambda_{BN})}{(\lambda_{PN} - \lambda_{BN}) + (\lambda_{BP} - \lambda_{PP})}$$

$$\beta = \frac{(\lambda_{BN} - \lambda_{NN})}{(\lambda_{BN} - \lambda_{NN}) + (\lambda_{NP} - \lambda_{BP})}$$

Table 1
Cost functions.

Decision action	C(Positive sample)	$\overline{C}$(Negative sample)
a_P	λ_{PP}	λ_{PN}
a_N	λ_{NP}	λ_{NN}
a_B	λ_{BP}	λ_{BN}

4.2. The rule of three-way recommendation

The traditional two-way recommendation model only contained two simple decisions: recommend or not recommend. However, when the information is insufficient, uncertain and inaccurate, it is difficult to make an accurate recommendation with two recommendations. In order to reduce the recommendation cost of the traditional two-way recommendation model, three-way decisions are introduced on the basis of collaborative filtering algorithm to make the uncertain results sustainable, and a three-way sustainable recommendation model is constructed. The three-way recommendation systems mainly include the following four parts.

1) Define two user preferences $\Theta = \{X, \overline{X}\}$:$X$ means like and$\overline{X}$means dislike.
2) Define a set of decision actions AC: a_P, a_N, a_B respectively denote three decision actions of recommendation, non-recommendation and delayed recommendation.
3) Predict the probability that users like items $P(X|v_j)$.
4) According to the three-way recommendation decision rules, choose to recommendation, non-recommendation, or delay recommendation.

Compared with the recommendation or non-recommendation in the two-way recommendation model, the three-way recommendation considers the cost of misclassification in the recommendation process, and obtains three-way decisions of recommendation, non-recommendation or delayed recommendation. The details of the recommendation cost matrix are shown in Table 2.

λ_{PP}, λ_{NP} and λ_{BP} respectively denote the cost of recommendation, non-recommendation, and delay recommendation items that users like, λ_{PN}, λ_{NN} and λ_{BN} respectively denote the cost of recommendation, non-recommendation, and delay recommendation items that users don't like. The expected cost of adopting recommendation, non-recommendation and delayed recommendation are

$$\begin{aligned} Cost_P &= \lambda_{PP} \times N_{PP} + \lambda_{PN} \times N_{PN} \\ Cost_N &= \lambda_{NP} \times N_{NP} + \lambda_{NN} \times N_{NN} \\ Cost_B &= \lambda_{BP} \times N_{BP} + \lambda_{BN} \times N_{BN} \end{aligned} \tag{15}$$

Total recommendation cost $T - Cost = Cost_P + Cost_N + Cost_B$. $Cost_P$, $Cost_N$and$Cost_B$respectively denote the expected cost of recommending, not recommending and delaying recommendation actions. According to the Bayesian decision criterion, the following minimum cost decision rules can be obtained.

If $Cost_P \leq Cost_N$ and $Cost_P \leq Cost_B$, then $v_j \in POS_{(\alpha,\beta)}(X)$.
If $Cost_N \leq Cost_P$ and $Cost_N \leq Cost_B$, then $v_j \in NEG_{(\alpha,\beta)}(X)$.
If $Cost_B \leq Cost_P$ and $Cost_B \leq Cost_N$,then $v_j \in BND_{(\alpha,\beta)}(X)$.

In formula (15), N_{PP}, N_{NP} and N_{BP} respectively denote the number of a_P, a_N and a_B actions taken by the user on the items they like. N_{PN}, N_{NN} and N_{BN} respectively denote the number of a_P, a_N and a_B actions taken by the user on the items they dislike. The calculation formula is shown in formula (16).

Table 2
Three-way recommendation cost matrix.

Decision rules	User's preferences Like(X)	Dislike($\overline{X}$)
Recommendation (a_P)	λ_{PP}	λ_{PN}
Non-recommendation (a_N)	λ_{NP}	λ_{NN}
Delayed recommendation (a_B)	λ_{BP}	λ_{BN}

$$\begin{aligned}
N_{PP} &= \left|\left\{\langle i,j\rangle \middle| r_{ij} \geq \varphi, P(X|v_j) \geq \alpha\right\}\right| \\
N_{PN} &= \left|\left\{\langle i,j\rangle \middle| 0 < r_{ij} < \varphi, P(X|v_j) \geq \alpha\right\}\right| \\
N_{NP} &= \left|\left\{\langle i,j\rangle \middle| r_{ij} \geq \varphi, P(X|v_j) \leq \beta\right\}\right| \\
N_{NN} &= \left|\left\{\langle i,j\rangle \middle| 0 < r_{ij} < \varphi, P(X|v_j) \leq \beta\right\}\right| \\
N_{BP} &= \left|\left\{\langle i,j\rangle \middle| r_{ij} \geq \varphi, \beta < P(X|v_j) < \alpha\right\}\right| \\
N_{BN} &= \left|\left\{\langle i,j\rangle \middle| 0 < r_{ij} < \varphi, \beta < P(X|v_j) < \alpha\right\}\right|
\end{aligned} \tag{16}$$

In the formula, $P(X|v_j)$ denotes the probability that u_i likes v_j, r_{ij} represents the actual score of u_i to v_j, φ denotes the threshold of scoring level.

According to the Bayesian decision criterion, *POS* domain, *NEG* domain and *BND* domain are obtained.

$$\begin{aligned}
POS_{(\alpha,\beta)}(X) &= \left\{v_j \middle| P(X|v_j) \geq \alpha\right\} \\
NEG_{(\alpha,\beta)}(X) &= \left\{v_j \middle| P(X|v_j) \leq \beta\right\} \\
BND_{(\alpha,\beta)}(X) &= \left\{v_j \middle| \beta < P(X|v_j) < \alpha\right\}
\end{aligned} \tag{17}$$

Where

$$\begin{aligned}
\alpha &= \frac{(\lambda_{PN} - \lambda_{BN})}{(\lambda_{PN} - \lambda_{BN}) + (\lambda_{BP} - \lambda_{PP})} \\
\beta &= \frac{(\lambda_{BN} - \lambda_{NN})}{(\lambda_{BN} - \lambda_{NN}) + (\lambda_{NP} - \lambda_{BP})}
\end{aligned} \tag{18}$$

In general, the higher the score, the greater the user's preference for the item.However, the user's scoring of an item is not only related to the user's degree of preference, but also depends on the user's scoring preference. User's scoring benchmarks are different. In order to avoid the influence of user scoring preferences, the Min-Max standardization method is used to process the predicted scores and the probability that u_i likes v_j is obtained. The probability formula is shown in formula (19).

$$P(X|v_j) = \begin{cases} 1 r'_{ij} > \mathrm{Max} r_i \\ \dfrac{r'_{ij} - \mathrm{Min} r_i}{\mathrm{Max} r_i - \mathrm{Min} r_i} \mathrm{Min} r_i \leq r'_{ij} \leq \mathrm{Max} r_i \\ 0 r'_{ij} \leq \mathrm{Min} r_i \end{cases} \tag{19}$$

In the formula, r'_{ij} denotes the predicted score of u_i to v_j, $\mathrm{Min}r_i$ and $\mathrm{Max}r_i$ denote the minimum and maximum scores of u_i respectively.

4.3. Description of the algorithm

Given a user set $U = \{u_1, u_2, \cdots, u_n\}$, an item set $V = \{v_1, v_2, \cdots, v_m\}$, an item's attribute set $A = \{a_1, a_2, \cdots, a_h\}$ and each item $v_j = \{g_1^{(j)}, g_2^{(j)}, \cdots, g_h^{(j)}\}$, $g_i^{(j)}$ denotes the value of item v_j in attribute a_i. Given the rating threshold $\varphi = 3$, the items rated by the user are divided into two categories: likes (X) and dislike $(\overline{X})$. Recommend a list of items for users according to their preferences.

Fig. 1 is the overall process of 3NBCFR algorithm. Fig. 1(a) shows the process of filling the score matrix by the Naive Bayesian classifier. Aiming at the problem of user scoring sparsity, each user uses the existing scoring information and item attributes to construct the Naive Bayesian classifier to predict the missing data in the score matrix, as in Algorithm 1. Fig. 1(b) shows the use of three-way recommendation score matrix to calculate the threshold (α, β). Fig. 1(c) and Fig. 1(d) show the use of predicted scores r'_{ij} to obtain the probability of the target user t_i likes the item v_j, and three-way recommendations are made for the unrated item v_j: recommendation, non-recommendation and delayed recommendation. Finally, Fig. 1(e) shows that items are selected from recommendation and delayed recommendation according to the probability to form a recommendation list.

The details of the three-way recommendation algorithms are shown in Algorithm 2.

Step1~5: Calculate $sim(i,j)$ between u_i and u_j.
Step 7 and Step 8: Calculate t_i's neighbor user set $G(t_i)$ and unrated item set $V(t_i)$.
Step 9~12: First, calculate the prediction score r'_{ij} of t_i to v_j, and then predict the probability $P(X|v_j)$ that t_i likes v_j according to the obtained r'_{ij}.
Step 13: Calculate the threshold value (α, β) according to the cost matrix.
Step 14~20: Compare $P(X|v_j)$ and (α, β) according to the three-way recommendation rules and get three-way recommendation results.

Finally, the recommendation list is obtained according to the three-

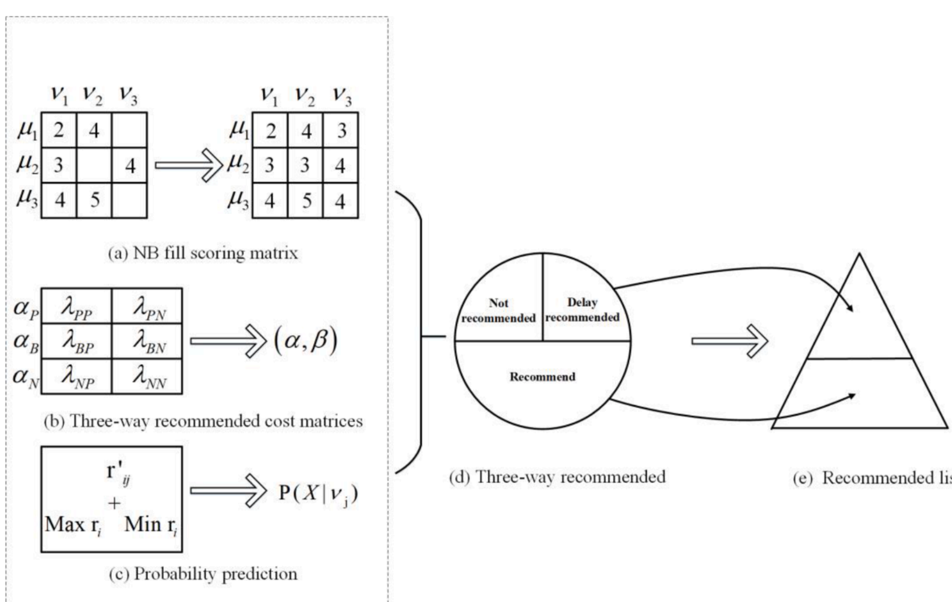

Fig. 1. Three-way Naive Bayesian recommendation model.

way recommendation results.

Algorithm 2:Three-way sustainable recommendation model
Input: Score matrix R, User set $U = \{u_1, u_2, \cdots, u_n\}$, Item set $V = \{v_1, v_2, \cdots, v_m\}$, Target user set $T = \{t_1, t_2, \cdots, t_P\}$
Output: Three-way recommendation results and Recommendation list
for $u_i \in U$ **do**
for $u_j \in U$ **do**
Obtain the similarity $sim(i, j)$ between u_i and u_j according to formula (4)
end for
end for
for $t_i \in T$ **do**
Calculate the l neighbors user set $G(t_i)$ of t_i
Calculate the set of unrated items $V(t_i)$ for t_i
for $v_j \in V(t_i)$ **do**
Calculate the predicted score r'_{ij} of t_i to v_j according to formula (5)
Predict the probability $P(X|v_j)$ of t_i like v_j according to formula (19)
end for
Calculate the threshold (α, β) according to formula (18)
if $P(X|v_j) \geq \alpha$ **then**
Divide v_j into $POS_{(\alpha,\beta)}(X)$
else if $P(X|v_j) \leq \beta$ **then**
Divide v_j into $NEG_{(\alpha,\beta)}(X)$
else if $\beta < P(X|v_j) < \alpha$ **then**
Divide v_j into $BND_{(\alpha,\beta)}(X)$
end if
end for

4.4. Analysis of algorithm

The time complexity of Algorithm 1 mainly comes from the establishment of Naive Bayesian classifiers for n users, that is, the corresponding relationship between item-item attributes is established for each user. The total number of items is m, and the total number of item attributes is h. Therefore, the time complexity of establishing each Naive Bayesian classifier is $O(hm)$, then the total time complexity is $nO(hm)$.

Algorithm 2 is mainly divided into 3 parts. The first part is to calculate the similarity between users. The second part is to predict the score and probability. The third part is three-way recommendation. The main cost of the algorithm comes from the first part, which needs to calculate the similarity of n users, so the overall time complexity is $O(n^2)$.

In summary, the total time complexity of the 3NBCFR algorithm is $nO(hm)+O(n^2)$. But in actual applications, the number of item attributes h will not be too large, so the time complexity of 3NBCFR is $O(n^2)$. The time complexity of the traditional collaborative filtering algorithm is $O(n^2)$, so the 3NBCFR algorithm does not produce higher consumption.

5. Experiment and result analysis

In this section, firstly, we introduce the experimental environment and the score matrix of the data set. Then we analyze the performance and advantages of the algorithm from the two aspects of recommendation quality and recommendation cost.

5.1. Experimental environment

We select Movielens (100 K) (Harper & Konstan, 2015) as the experimental data set. The Movielens data set contains ratings data of multiple users on multiple movies, from which 30 users' ratings on 14 movies are randomly selected as the data set. The composed score matrix is shown in Table 3, where 0 represents the missing value of the matrix. The experiment is implemented in PyCharm 2018.3.4 version under the WIN7 system environment. The system processor is Inter Core i7–6700, and the RAM size is 8 GB.

5.2. Experimental process and result analysis

1) Quality of recommendation

To visually express the experimental results, the experimental data need to be trimmed. Given that the data set itself is randomly ordered, and the data constituting the score matrix is also randomly selected in this paper, we directly used the last six user scoring data. In order to ensure the rigor of the experiment and more comprehensively verify the recommended effect of the proposed model, the number of neighbor users 2, 3 and 4 were selected to carry out the experiment, and the probability prediction value of the unscored movie for six users in three cases was calculated, as shown in Tables 4–6. And according to the probability value, three movies were recommended to six users, as shown in Fig. 2.

From Table 4 to 6 and Fig. 2, for user u_{25}, when the number of neighbor users is 2, the recommended movie order is v_1,v_3,v_{10}, when the number of neighbor users is 3 or 4, the recommended movie order is v_1,v_{10},v_3. There are some sequence changes for user u_{25}. For user u_{26}, when the number of neighbor users is 2, the recommended movie order is v_9,v_{14},v_{11}, and when the number of neighbor users is 3 or 4, the recommended movie order is v_{14},v_{11},v_9. The order of movies v_9 changes from first to third. For user u_{27}, when the number of neighbor users is 2, 3, 4, the recommended movie order is v_{10},v_7,v_4. There is no change. For users u_{28}, when the number of neighbor users is 2, 3, 4, the order of recommended movies is v_6,v_8,v_1. For users u_{29}, when the number of neighbor users is 2, 3, 4, the order of recommended movies is v_7,v_{11},v_{12}. For user u_{30}, when the number of neighbor users is 2, the recommended movie order is v_6,v_9,v_{12}, and when the number of neighbor users is 3 or 4, the recommended movie order is v_6,v_4,v_{12}. The movie v_9 is no longer recommended to the user and the movie v_4 is recommended to the user.

2) Recommended cost

Firstly, the decision recommendation is made according to the predicted score and predicted probability generated by the training set, and then the total recommendation cost $T-Cost$ is calculated according to the actual score, score level threshold φ, prediction probability $P(X|v_j)$ and threshold (α, β) of the users to be recommended. In order to verify that the recommendation cost of the 3NBCFR algorithm is improved compared with the traditional collaborative filtering recommendation algorithm, the recommendation cost of the two algorithms when the number of neighbor users is 2 is compared, as shown in Tables 7 and 8.

When calculating the recommendation cost, we mainly consider the misclassification cost caused by the wrong recommendation and the learning cost caused by the delayed recommendation, so the cost of the correct recommendation is not considered, that is $\lambda_{PP} = \lambda_{NN} = 0$. In Table 7, when $\lambda_{PN} = 0.8$ is fixed, λ_{NP} varies from 1.2 to 0.4, and $(\lambda_{BP}, \lambda_{BN})$ varies from $(0.35, 0.35)$ to $(0.15, 0.15)$, the recommendation cost comparison results of the two algorithms are displayed. Table 8 shows the recommendation cost comparison results of the two algorithms when $\lambda_{NP} = 0.8$ is fixed and λ_{PN} varies from 1.2 to 0.4, and $(\lambda_{BP}, \lambda_{BN})$ varies from $(0.35, 0.35)$ to $(0.15, 0.15)$. Then calculate the threshold (α, β) and the total recommendation cost $T-Cost$ according to the method described in Section 4.2. The results are shown in Tables 9 and 10 and the results are displayed in two rows, the upper side is the threshold (α, β), and the lower side is the $T-Cost$ value.

From Tables 7 and 8, when the cost conditions are the same, the recommendation cost of the three-way Naive Bayesian recommendation algorithms is generally lower than that of the collaborative filtering algorithm. According to the score matrix filled by the construction of the Naive Bayesian classifier and the number of misclassifications reduced by the delay recommendation, the recommendation cost in the recommendation process is reduced. It can be seen from Tables 9 and 10 that when λ_{PN} and λ_{NP} are fixed, the smaller the value of λ_{BP} and λ_{BN} (that is the larger the threshold α and the smaller the β value), the lower the recommendation cost. When λ_{PN}, λ_{BP} and λ_{BN} are fixed and λ_{NP} is smaller, the recommendation cost generally shows a downward trend. Similarly, when λ_{NP}, λ_{BP} and λ_{BN} are fixed and λ_{PN} is smaller, the recommendation cost generally shows a downward trend.

Comprehensive experimental results show that the 3NBCFR

Table 3
Score matrix.

	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	v_{13}	v_{14}
u_1	5	3	4	3	3	4	1	5	3	2	5	5	5	5
u_2	3	3	0	5	1	2	4	3	0	1	5	5	4	0
u_3	5	0	2	4	4	4	4	4	0	5	5	0	2	5
u_4	2	1	3	2	0	4	3	0	3	3	4	3	0	0
u_5	2	0	4	4	0	4	0	4	4	5	5	5	5	5
u_6	2	0	3	4	0	4	5	4	0	0	4	4	4	2
u_7	4	3	0	3	0	4	4	4	5	4	5	0	0	0
u_8	4	3	0	4	4	4	5	4	0	4	5	4	0	3
u_9	4	0	0	2	3	4	3	3	0	4	4	0	0	4
u_{10}	3	0	3	0	3	5	0	2	0	5	5	5	0	2
u_{11}	5	0	0	5	0	4	0	4	5	0	5	3	5	4
u_{12}	5	0	0	5	0	3	5	3	5	0	0	4	5	2
u_{13}	3	2	0	4	0	3	3	3	3	4	4	0	0	3
u_{14}	0	3	2	4	3	0	3	4	0	3	3	3	4	4
u_{15}	4	3	0	3	0	5	1	5	0	5	5	0	0	3
u_{16}	3	2	0	4	3	2	5	3	3	2	1	3	3	3
u_{17}	5	5	0	5	5	4	0	4	0	5	5	0	0	5
u_{18}	3	2	1	4	0	4	0	0	4	4	4	3	0	0
u_{19}	5	4	3	4	3	5	4	5	0	5	5	0	4	0
u_{20}	4	3	2	3	4	4	5	5	0	5	5	5	0	0
u_{21}	5	0	3	4	5	5	4	5	0	4	5	0	0	5
u_{22}	2	3	2	4	3	3	3	0	0	3	4	0	3	3
u_{23}	4	2	2	4	0	4	4	3	0	4	4	0	0	4
u_{24}	5	3	3	4	2	4	5	5	0	4	4	4	0	3
u_{25}	0	2	0	4	0	3	4	0	4	0	3	2	4	0
u_{26}	3	0	0	4	0	4	0	5	0	3	0	3	5	0
u_{27}	4	2	0	0	3	4	0	5	3	0	5	3	0	4
u_{28}	0	0	3	2	4	0	4	0	3	0	4	2	0	4
u_{29}	5	4	0	4	0	4	0	5	3	4	0	0	4	0
u_{30}	5	0	3	0	2	0	3	4	0	5	5	0	0	3

Table 4
Probability prediction value when number of neighbor users is 2.

User	Probability						
u_{25}	**0.88**(v_1)	**0.88**(v_3)	0.47(v_5)	0.17(v_8)	**0.88**(v_{10})	0.00(v_{14})	
u_{26}	0.04(v_2)	0.00(v_3)	0.43(v_5)	0.24(v_7)	**0.74**(v_9)	**0.43**(v_{11})	**0.74**(v_{14})
u_{27}	0.23(v_3)	**0.56**(v_4)	**0.71**(v_7)	**0.74**(v_{10})	0.41(v_{13})		
u_{28}	**0.60**(v_1)	0.10(v_2)	**1.00**(v_6)	**0.83**(v_8)	0.56(v_{10})	0.56(v_{13})	
u_{29}	0.24(v_3)	0.22(v_5)	**1.00**(v_7)	**0.74**(v_{11})	**0.74**(v_{12})	0.00(v_{14})	
u_{30}	0.68(v_2)	0.68(v_4)	**0.88**(v_6)	**0.68**(v_9)	**0.68**(v_{12})	0.68(v_{13})	

Table 5
Probability prediction value when number of neighbor users is 3.

User	Probability						
u_{25}	**0.77**(v_1)	**0.64**(v_3)	0.46(v_5)	0.37(v_8)	**0.77**(v_{10})	0.22(v_{14})	
u_{26}	0.04(v_2)	0.00(v_3)	0.32(v_5)	0.00(v_7)	**0.55**(v_9)	**0.60**(v_{11})	**0.82**(v_{14})
u_{27}	0.19(v_3)	**0.52**(v_4)	**0.63**(v_7)	**0.65**(v_{10})	0.42(v_{13})		
u_{28}	**0.61**(v_1)	0.11(v_2)	**0.95**(v_6)	**0.78**(v_8)	0.58(v_{10})	0.58(v_{13})	
u_{29}	0.07(v_3)	0.34(v_5)	**1.00**(v_7)	**0.85**(v_{11})	**0.85**(v_{12})	0.04(v_{14})	
u_{30}	0.67(v_2)	**0.67**(v_4)	**0.81**(v_6)	0.49(v_9)	**0.67**(v_{12})	0.49(v_{13})	

algorithm has better recommendation results than the traditional collaborative filtering algorithm, which proves that the construction of a Naive Bayesian classifier can effectively solve the problem of data sparsity and delayed recommendation can solve the unsustainable problem of two-way decisions. 3NBCFR has reduced the recommendation cost, and promoted the construction of smart city.

6. Conclusion and future scope

This paper combines three-way decisions, Naive Bayesian and collaborative filtering algorithm, a three-way Naive Bayesian collaborative filtering recommendation model (3NBCFR) is proposed, which effectively mitigates the data sparseness in the recommendation system,

Table 6
Probability prediction value when number of neighbor users is 4.

User	Probability						
u_{25}	**0.76**(v_1)	**0.56** (v_3)	0.51 (v_5)	0.44(v_8)	**0.66**(v_{10})	0.32(v_{14})	
u_{26}	0.08(v_2)	0.00(v_3)	0.40 (v_5)	0.00(v_7)	**0.49**(v_9)	**0.62**(v_{11})	**0.81**(v_{14})
u_{27}	0.20(v_3)	**0.61**(v_4)	**0.62**(v_7)	**0.63**(v_{10})	0.45(v_{13})		
u_{28}	**0.64**(v_1)	0.24(v_2)	**0.83**(v_6)	**0.69**(v_8)	0.62(v_{10})	0.62(v_{13})	
u_{29}	0.09(v_3)	0.41(v_5)	**1.00**(v_7)	**0.92**(v_{11})	**0.81**(v_{12})	0.07(v_{14})	
u_{30}	0.64(v_2)	**0.64**(v_4)	**0.75**(v_6)	0.49(v_9)	**0.71**(v_{12})	0.56(v_{13})	

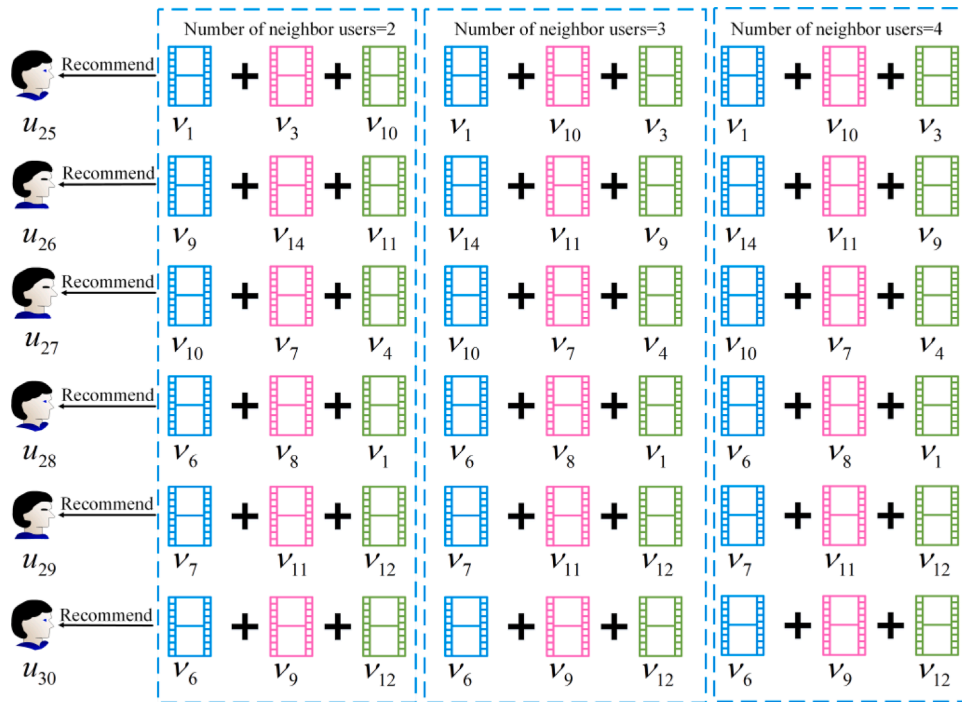

Fig. 2. Results of recommendation.

Table 7
Recommendation cost when λ_{PN} is fixed.

$(\lambda_{PN}, \lambda_{NP})$	$(\lambda_{BP}, \lambda_{BN})$									
	(0.35,0.35)		(0.30,0.30)		(0.25,0.25)		(0.20,0.20)		(0.15,0.15)	
	CF	**3NBCFR**	CF	**3NBCFR**	CF	**3NBCFR**	CF	**3NBCFR**	CF	**3NBCFR**
(0.8,1.2)	15.61	**13.40**	15.61	**14.42**	15.61	**11.45**	15.61	**11.80**	15.61	**9.60**
(0.8,1.0)	13.56	**11.40**	13.56	**12.42**	13.56	**13.25**	13.56	**10.60**	13.56	**8.60**
(0.8,0.8)	11.50	**10.75**	11.50	**10.45**	11.50	**11.25**	11.50	**9.80**	11.50	**8.25**
(0.8,0.6)	–	–	10.72	**9.60**	10.72	**9.60**	10.72	**7.80**	10.72	**7.35**
(0.8,0.4)	–	–	–	–	–	–	9.56	**8.60**	9.56	**6.85**

Table 8
Recommendation cost when λ_{NP} is fixed.

$(\lambda_{PN}, \lambda_{NP})$	$(\lambda_{BP}, \lambda_{BN})$									
	(0.35,0.35)		(0.30,0.30)		(0.25,0.25)		(0.20,0.20)		(0.15,0.15)	
	CF	**3NBCFR**	CF	**3NBCFR**	CF	**3NBCFR**	CF	**3NBCFR**	CF	**3NBCFR**
(1.2,0.8)	15.23	**13.90**	15.23	**13.70**	15.23	**12.75**	15.23	**11.80**	15.23	**8.40**
(1.0,0.8)	12.85	**12.15**	12.85	**11.90**	12.85	**12.75**	12.85	**11.80**	12.85	**8.40**
(0.8,0.8)	11.86	**10.75**	11.86	**10.40**	11.86	**11.25**	11.86	**10.60**	11.86	**8.25**
(0.6,0.8)	–	–	10.23	**9.20**	10.23	**9.75**	10.23	**9.60**	10.23	**8.25**
(0.4,0.8)	–	–	–	–	–	–	9.65	**8.80**	9.65	**6.60**

Table 9
Value of (a, b) and $T - Cost$ when λ_{PN} is fixed.

$(\lambda_{PN}, \lambda_{NP})$	$(\lambda_{BP}, \lambda_{BN})$				
	(0.35,0.35)	(0.30,0.30)	(0.25,0.25)	(0.20,0.20)	(0.15,0.15)
(0.8,1.2)	(0.563,0.292) 13.4	(0.625,0.25) 14.4	(0.688,0.208) 11.45	(0.75,0.167) 11.8	(0.813,0.125) 9.6
(0.8,1.0)	(0.563,0.35) 11.4	(0.625,0.3) 12.4	(0.688,0.25) 13.25	(0.75,0.2) 10.6	(0.813,0.15) 8.6
(0.8,0.8)	(0.563,0.438) 10.75	(0.625,0.375) 10.4	(0.688,0.313) 11.25	(0.75,0.25) 9.8	(0.813,0.188) 8.25
(0.8,0.6)	(0.563,0.583) 0	(0.625,0.5) 9.6	(0.688,0.417) 9.6	(0.75,0.333) 7.8	(0.813,0.25) 7.35
(0.8,0.4)	(0.563,0.875) 0	(0.625,0.75) 0	(0.688,0.625) 0	(0.75,0.5) 8.6	(0.813,0.375) 6.85

Table 10
Value of (a, b) and $T - Cost$ when λ_{NP} is fixed.

$(\lambda_{PN}, \lambda_{NP})$	$(\lambda_{BP}, \lambda_{BN})$				
	(0.35,0.35)	(0.30,0.30)	(0.25,0.25)	(0.20,0.20)	(0.15,0.15)
(1.2,0.8)	(0.708,0.438) 13.9	(0.75,0.375) 13.7	(0.792,0.313) 12.75	(0.833,0.25) 11.8	(0.875,0.188) 8.4
(1.0,0.8)	(0.65,0.438) 12.15	(0.7,0.375) 11.9	(0.75,0.313) 12.75	(0.8,0.25) 11.8	(0.85,0.188) 8.4
(0.8,0.8)	(0.563,0.438) 10.75	(0.625,0.375) 10.4	(0.688,0.313) 11.25	(0.75,0.25) 9.8	(0.813,0.188) 8.25
(0.6,0.8)	(0.417,0.438) 0	(0.5,0.375) 9.2	(0.583,0.313) 9.75	(0.667,0.25) 9.6	(0.75,0.188) 8.25
(0.4,0.8)	(0.125,0.438) 0	(0.25,0.375) 0	(0.375,0.313) 0	(0.5,0.25) 8.8	(0.625,0.188) 6.6

and solves the problem that the recommendation cost of two-way decisions is high but the efficiency is low to the detriment of sustainable recommendation. Using the existing score information and item attributes to construct a Naive Bayesian classifier to predict the score of unrated items, improves the problem of data sparseness. Secondly, in order to solve the uncertainty of two-way decisions which is difficult to deal with, and further reduce the cost of recommendation, three-way decisions are introduced, which realizes three-way recommendations according to the users' real score, grade threshold, recommendation cost and prediction probability. Finally, the experiment shows that 3NBCFR is better than the traditional collaborative filtering, suitable for movie recommendation, and the recommendation decision-making becomes more intelligent, which helps to build a smart city.

In the next research work, since the attribute condition independence assumption of Naive Bayesian affects the classification effect, it has a certain influence on the filling of the score matrix in the recommendation system, and further research will be conducted on the filling of the score matrix with Semi-Naive Bayesian, Bayesian network and other methods to obtain better recommendation effects and contribute to the construction of a smart city.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Breese, J., Hecherman, D., & Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. In *The 14th Conference on Uncertainty in Artificial Intelligence (UAI-98)* (pp. 43–52).
Choi, K., & Suh, Y. (2013). A new similarity function for selecting neighbors for each target item in collaborative filtering. *Knowledge-Based Systems, 37*, 146–153.
Filip, Š., Dominika, B., & Jozef, G. (2021). City Logistics as an Imperative Smart City Mechanism: Scrutiny of Clustered EU27 Capitals. *Sustainability, 13*(7), 1–16.
Fong, C. S., Aghamohammadi, N., Ramakreshnan, L., Sulaiman, N. M., & Mohammadi, P. (2019). Holistic recommendations for future outdoor thermal comfort assessment in tropical Southeast Asia: A critical appraisal. *Sustainable Cities and Society, 46*.
Gao, K. Y., Yang, X., Wu, C. X., Qiao, T. T., Chen, X. Y., Yang, M., et al. (2020). Exploiting location-based context for POI recommendation when traveling to a new region. *IEEE Access : Practical Innovations, Open Solutions, 8*, 52404–52412.
Gao, Y., Qi, H., Liu, J., & Liu, D. Y. (2008). A collaborative filtering recommendation algorithm combining probabilistic relational models and user grade. *Journal of Computer Research and Development, 45*(9), 1463–1469.
Goldberg, D., Nichols, D., Oki, B. M., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM, 35*(12), 61–70.
Harper, F. M., & Konstan, J. A. (2015). The MovieLens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems, 5*(4), 1–19.
Hu, F., Zhang, M., & Yu, H. (2019). Active learning method based on three-way decision. *Control and Decision, 34*(4), 718–726.
Huynh, H. X., Phan, N. Q., Pham, N. M., Pham, V. H., & Ismail, M. (2020). Context-similarity collaborative filtering recommendation. *IEEE Access : Practical Innovations, Open Solutions, 8*, 33342–33351.
Jackowski, K. (2018). New diversity measure for data stream classification ensembles. *Engineering Applications of Artificial Intelligence, 74*, 23–34.
Li, B., Tian, L. Y., Chen, D. Q., & Liang, S. Y. (2020a). An adaptive dwell time scheduling model for phased array radar based on three-way decision. *Journal of Systems Engineering and Electronics, 31*(3), 500–509.
Li, D. C., & Gong, Z. G. (2020). A Deep neural network for crossing-city POI recommendations. *IEEE Transactions on Knowledge and Data Engineering, 1*.
Li, M. X., Chen, K. B., Liu, C. Y., & Liu, B. X. (2020b). The interval parameter optimization model based on three-way decision space and its application on "green products recommendation". *Discrete Dynamics in Nature & Society, 2020*, 1–12.
Liu, D., & Ye, X. Q. (2020). A matrix factorization based dynamic granularity recommendation with three-way decisions. *Knowledge-Based Systems, 191*.
Liu, L. L., & Zhang, Y. (2021). Smart environment design planning for smart city based on deep learning. *Sustainable Energy Technologies and Assessments, 47*, Article 101425.
Nathan, T., & Owusu, A. (2020). Sustainable adoption of smart homes from the Sub-Saharan African perspective. *Sustainable Cities and Society, 63*, Article 102434.
Pan, Y. H., Wang, Y. H., Zhou, P., Yan, Y., & Guo, D. M. (2020). Activation functions selection for Bp neural network model of ground surface roughness. *Journal of Intelligent Manufacturing, 31*(8), 1825–1836.
Qin, Q., & Zhang, H. R. (2020). Three-way recommendation based on trust transfer mechanism. *Pattern Recognition and Artificial Intelligence, 33*(7), 600–609.
Ren, Y. G., Zhang, Y. P., & Zhang, Z. P. (2020). Collaborative filtering recommendation algorithm based on rough set rule extraction. *Journal on Community, 41*(1), 80–87.
Romain, R., Siddharth, V., & Peter, G. (2020). An observation of the impact of CoViD-19 recommendation measures monitored through urban noise levels in central Stockholm,Sweden. *Sustainable Cities and Society, 63*, Article 102469.
Soheil, S., Haneen, K., & Dominique, L. (2020). Impacts of autonomous vehicles on public health: A conceptual model and policy recommendations. *Sustainable Cities and Society, 63*(0), Article 102457.

Tang, X., Tan, Z. Y., Hu, S. Y., & Geng, H. (2018). Evaluating spatial service and layout efficiency of municipal Wi-Fi facilities for Smart City planning: A case study of Wuhan city, *China. Socio-Economic Planning Sciences, 65*, 101–110.

Wang, E. K., Wang, F., Kumari, S., Yeh, J. H., & Chen, C. M. (2020). Intelligent monitor for typhoon in IoT system of smart city. *The Journal of Supercomputing, 77*(3), 3024–3043.

Wang, H., Li, W., Zhao, Z. Z., Wang, Z. F., Li, M. H., & Li, D. F. (2021). Intelligent distribution of fresh agricultural products in smart city. *IEEE Transactions on Industrial Informatics*, 1.

Wang, L. Y., Zhang, C. Y., & Liu, B. X. (2016). Dynamic strategy regulation model of three-way decisions based on interval concept lattice and its application. *Computer Engineering and Applications, 52*(24), 80-84+101.

Wang, Y. G., & Liu, K. Q. (2020). Collaborative filtering recommendation algorithm for clustering optimization. *Computer Engineering and Applications, 56*(15), 66–73.

Xu, C., Guan, Z. Y., Zhao, W., Wu, Q. Z., Yan, M., Chen, L., et al. (2020). Recommendation by users' multi-modal preferences for smart city applications. *IEEE Transactions on Industrial Informatics, 1*.

Xue, Z. A., Wang, P. H., Liu, J., Zhu, T. L., & Xue, T. Y. (2016). Three-way decision theory based on intuitionistic fuzzy sets. *Computer Science, 43*(1), 30–34.

Yan, S. H., Yang, A. M., Kong, S. S., Bai, B., & Li, X. Y. (2021). Predictive intelligence powered attentional stacking matrix factorization algorithm for the computational drug repositioning. *Applied Soft Computing Journal, 110*, Article 107633.

Yang, A. M., Zhuansun, Y. X., Shi, Y., Liu, H. X., & Li, R. S. (2021). Iot system for pellet proportioning based on bas intelligent recommendation model. *IEEE Transactions On Industrial Informatics, 17*(2), 937–942.

Yang, B.& ., & Li, J. H. (2020). Complex network analysis of three-way decision researches. *International Journal of Machine Learning and Cybernetics, 11*(5), 973–987.

Yao, K. M., Wang, H. Y., Li, Y. L., & Rodrigues, J. J. P. C. (2021). Albuquerque VHCD (2021) A Group Discovery Method Based on Collaborative Filtering and Knowledge Graph for IoT Scenarios. *IEEE Transactions on Computational Social Systems*, 1–12.

Yao, Y. Y. (2009). Three-way decision: An interpretation of rules in rough set theory. In *Proceedings of the 4th International Conference on Rough Sets and Knowledge Technology* (pp. 642–649).

Yao, Y. Y. (2010). Three-way decisions with probabilistic rough sets. *Information Sciences, 180*, 341–353.

Yao, Y. Y. (2011). The superiority of three way decisions in probabilistic rough set models. *Information Sciences, 181*, 1080–1096.

Yao, Y. Y. (2012). An outline of a theory of three-way decisions. In *Proceedings of the 8th International Conference on Rough Sets and Current Trends in Computing (RSCTC 2012)* (pp. 1–17). Chengdu, China: Springer.

Yao, Y. Y., & Wong, S. K. M. (1992). A decision theoretic framework for approximating concepts. *International Journal of Man-machine Studies, 37*(6), 793–809.

Zhang, C. Y., Qiao, P., Wang, L. Y., Liu, L., & Zhang, J. S. (2017a). Dynamic three-way decision and its application based on probabilistic PS-rough set. *Journal of Nanjing University (Natural Science), 53*(5), 937–946.

Zhang, C. Y., Wang, L. Y., Li, M. X., & Liu, B. X. (2016). Model of three-way decision based on the space of set pair information granule and its application. *Journal on Communications, 37*(S1), 15–24.

Zhang, F., Lee, V. E., Jin, R. M., Garg, S., Choo, K. K. R., Michele, M., & Cheng, C. (2019). Privacy-aware smart city: A case study in collaborative filtering recommender systems. *Journal of Parallel and Distributed Computing, 127*, 145–159.

Zhang, H. R., Min, F., & Shi, B. (2017b). Regression-based three-way recommendation. *Information Sciences, 378*, 444–461.

Zhang, Q. H., Cheng, Y. L., Zhao, F., Wang, G. Y., & Xia, S. Y. (2021). Optimal scale combination selection integrating three-way decision with hasse diagram. *IEEE Transactions on Neural Networks and Learning Systems*, 1–15.

Zhang, Y. W., Yin, C. H., Wu, Q. L., He, Q., & Zhu, H. B. (2020). Location-aware deep collaborative filtering for service recommendation. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 1–12.

Zheng, L. P., Hu, M. J., Yang, H. H., & YJ, L. I. N. (2019). Research on collaborative filtering algorithm based on rough set. *Journal of Shandong University (Natural Science), 54*(2), 45–54.