symmetry

MDPI

Article

An Interactive Personalized Recommendation System Using the Hybrid Algorithm Model

Yan Guo, Minxi Wang * and Xin Li *

College of Management Science, Chengdu University of Technology, Chengdu 610059, China; guoyandoctor@sina.com

* Correspondence: wangminxi@mail.cdut.edu.cn (M.W.); lixin2012@cdut.cn (X.L.); Tel.: +86-28-84078945 (M.W.); +86-28-84078983 (X.L.)

Received: 6 August 2017; Accepted: 26 September 2017; Published: 6 October 2017

Abstract: With the rapid development of e-commerce, the contradiction between the disorder of business information and customer demand is increasingly prominent. This study aims to make e-commerce shopping more convenient, and avoid information overload, by an interactive personalized recommendation system using the hybrid algorithm model. The proposed model first uses various recommendation algorithms to get a list of original recommendation results. Combined with the customer's feedback in an interactive manner, it then establishes the weights of corresponding recommendation algorithms. Finally, the synthetic formula of evidence theory is used to fuse the original results to obtain the final recommendation products. The recommendation performance of the proposed method is compared with that of traditional methods. The results of the experimental study through a Taobao online dress shop clearly show that the proposed method increases the efficiency of data mining in the consumer coverage, the consumer discovery accuracy and the recommendation recall. The hybrid recommendation algorithm complements the advantages of the existing recommendation algorithms in data mining. The interactive assigned-weight method meets consumer demand better and solves the problem of information overload. Meanwhile, our study offers important implications for e-commerce platform providers regarding the design of product recommendation systems.

Keywords: e-commerce; recommendation system; interactive assign; hybrid algorithm; data mining

1. Introduction

The past few years have witnessed an astounding growth of electronic commerce, and online shopping has become a staple for many people. Meanwhile, mobile technology has promoted the development of online shopping [1]. In the virtual environment of e-commerce, enterprises can offer a great deal of product information. However, the new networked and pervasive information technology (IT) or computer-mediated environment produces an exploding volume of data, resulting in information overload [2]. If consumers want to find the product that they are interested in, they need to browse a lot of irrelevant information [3]. This process may lead to the continuous loss of consumers, and hinder the development of e-commerce. Therefore, consumers are in urgent need of a purchasing assistant to recommend products according to their interests. In this case, an e-commerce personalized recommendation system emerges, as the times require. It is a business intelligence platform based on massive data mining, which can help e-commerce websites provide personalized decision support and information service for customers. The recommendation system automatically completes the process of personalized selection, and recommends personalized products to meet customer demand [4,5].

However, one prerequisite for the current recommendation is that customers can clearly show their preferences. In fact, it is difficult for customers to express their preferences many times. Customers need to interact with the recommendation system and judge the recommendation results given by the system. Then, the system adjusts the recommended products, according to consumers' feedback. Multiple loops of interactive recommendations can ensure the accuracy of recommendations. Let us consider a scenario below. Lily is in need of a computer. If Lily knows nothing about computers, she cannot accurately describe her preferences by means of specific configuration indicators. In this case, the recommendation system needs to constantly identify her preferences by interacting with the customer.

Most of the current recommendation systems recommend products that have a high probability of being purchased [6,7]. They employ content-based filtering (CBF) [8,9], collaborative filtering (CF) [10,11] and data mining techniques [12]. We analyze the shortcomings of each algorithm in Section 2.1, and find that the existing algorithms have less direct interaction with customers [13]. The comparative results are shown in Table 1. In order to overcome the problems of each recommendation algorithm and exploit their respective advantages, many scholars studied the combined recommendation algorithms that fuse multiple recommendation techniques [14,15]. However, most of the research was based on two algorithms: CBF and CF. Few researchers studied the dynamic composition of multiple recommendation techniques with interactive designs.

Table 1. Comparative analysis of main recommendation algorithms.

Algorithms		Accuracy	Automaticity	Real-Time	Diversity	Scalability	Cold-Start Problem	Sparsity Problem
CBF		Inferior	Good	Good	Bad	Bad	New users	Not
CF	User-based CF	Better	Bad	Bad	Better	Bad	Serious	Serious
	Item-based CF	Better	Inferior	Inferior	Better	Bad	Serious	Serious
Association rules		General	Good	Good	Good	General	New projects	General

Therefore, this paper proposes a new model for a recommendation system of e-commerce using hybrid recommendation. The weights of different recommendation results were set according to the customer's real-time mutual information, so as to make the recommendation system more accurate. Firstly, generate the customer portrait based on consumer information, and use multiple recommendation methods to get a list of original recommendation results. Secondly, determine the corresponding weights according to the importance of different recommendation algorithms. In the process, the system continuously interacts with the consumer in real time, so that the personalized requirements of the consumer can be predicted accurately. Meanwhile, the weights of the corresponding algorithms are updated dynamically, and the optimal combination of the algorithms is automatically configured for the consumer. Finally, the final recommendation products are obtained by using the evidence synthesis formula to fuse results of the recommendation list.

The paper's organization is as follows: In Section 2, we review the literature of the recommendation system, hybrid algorithm and iterative design. Section 3 proposes the innovative methodology to improve the quality of recommendation. In Section 4, we design the experiment to illustrate the effectiveness of the proposed model. In the last section, we conclude the paper.

2. Literature Review

The literature review focuses on two perspectives: (1) the recommendation system and hybrid algorithm; (2) iterative design.

2.1. The Recommendation System and Hybrid Algorithm

The recommendation system is a kind of decision-support system based on the customer's preference [16]. In recent years, it has been widely used in e-commerce sites to provide consumers with product purchasing advice [17,18]. Since the development of the first recommendation system by Goldberg and his colleagues [19], various recommendation systems and related technologies have been introduced. Among these systems, user-based CF [20] is the most popular in e-commerce, such as on Amazon.com [21]. This system identifies a customer's preference to recommend the products most likely to be purchased by a similar customer group. However, the user-based CF has certain shortcomings. It is difficult to measure the similarities between customers, and there is the scalability issue [22]. As the number of customers and products increases, the computation time of algorithms increases exponentially [23]. To solve the scalability problem, there is the item-based CF method [24], which calculates item similarities offline. This method is based on customer's online history, and assumes that a customer will be more likely to purchase products that are similar or related to the products that he or she has already purchased.

The CBF method applies content analysis to target products that are described by their characteristics, such as size, color, shape and material. The recommendation system [25,26] guesses the consumer shopping preferences according to the characteristics of products that the consumers purchased in the past, and then recommends similar products to him or her [27]. However, a pure CBF system also has its limitations. One limitation is that customers can only receive recommendations similar to their earlier experiences. The other limitation is that certain items [28], such as music, photographs and multimedia, are difficult to analyze [29]. Based on CF and CBF [30], mining association rules methods has been introduced to help customers find products to purchase. At present, the Apriori algorithm is widely used in association rules mining. This algorithm has good automaticity, real-time applicability and diversity. However, it has a large amount of data in the process of mining frequent item sets, which increases the time and complexity of recommendation systems [31]. We compare the merits and demerits of the main recommendation algorithms, and obtain the result shown in Table 1.

In order to overcome the problems of the above recommendation algorithms and exploit their respective advantages, many scholars studied hybrid recommendation algorithms that fuse multiple recommendation techniques [14,15]. The hybrid recommendation algorithms can be divided into the following categories, according to the fusion degree: (1) the recommendation results based on CBF and CF are implemented respectively. Then, fuse the results to get the final results; (2) some features of CF are integrated into CBF, the latter of which is the main method; (3) some features of CBF are integrated into CF, the latter of which is the main method; (4) a hybrid recommendation model based on CBF and CF is proposed. The model not only contains the characteristics of CBF, but also contains the characteristics of CF. Peng et al. [32] used a hybrid recommendation algorithm based on CBF and item-based CF to carry out a linear combination of item-property similarity and item-scoring similarity. However, its balance factor was static, and it only considered the number of the same attributes, ignoring the influence of other attributes. On that basis, Wu et al. [33] changed the equilibrium factor into dynamic, but the balance factor was not significant. Hussein et al. [34] provided a range of recommendation algorithms and strategies for producing group recommendations, as well as templates for combining different methods into hybrid recommenders. We find most research was based on two algorithms: CBF and CF. Few researchers studied dynamic composition of multiple recommendation techniques with interactive design.

2.2. Iterative Design

The recommendation system is not only based on customer history record, but also needs to be based on the current behavior data to give real-time feedback. The system continuously interacts with the customer and analyzes the customer feedback to correct and optimize the recommendation results. Interactivity realizes the dialogue between the customer and system. The goal of the recommendation system is to recommend products that best suit customers. Research shows that an effective recommendation system can inspire customers' trust in the system [35]. The recommendation system has transparent system logic and provides detailed information about recommended products. Swearing et al. [36] improved the recommendation method that allows customers to interact with the system more effectively, to make recommendations more effective. Felfernig et al. [37] put forward the idea of case-based reasoning, which used the products' characteristics to interact with the customer to determine the most appropriate products. Some researchers proposed interactive visualizations as a means to support interaction with recommendation systems at the ACM conference on Intelligent User Interfaces (IUI) in 2013 [38]. He et al. [39] presented an interactive visualization framework that combines recommendation with visualization techniques to support human–recommender interaction.

In the past 10 years, researchers studied recommendation systems from several perspectives: prediction accuracy, algorithm scalability, knowledge sources, types of recommended projects and tasks, and evaluation methods. There is not much research on the interaction between customers and recommendation systems from customers' perspective. The current black-box nature of recommendation systems prevents customers from providing input into the recommendation process in an interactive and iterative manner. Pleasant interactions of co-creation activities can provide personalized service well [40]. As the predication of the current task or interest of the customer is a challenging task, there is a need to develop a hybrid approach that enables the customer to interact with the system [41].

In this research, we propose an interactive personalized recommendation system based on a hybrid algorithm of multiple recommendation algorithms. The system can collect the customer's feedback information in real-time and dynamically adjust the weights of recommendation algorithms. Mufti-strategy fusion optimizes the diversity, novelty and timeliness of the recommendation, and gives customers surprising results. Details of our algorithm are discussed next.

3. Model Formulation

3.1. The Solution Framework

To construct an interactive recommendation system based on a hybrid algorithm of multiple recommendation algorithms, we followed the framework as outlined in Figure 1. The first phase is to preprocess the information of products and consumers, where retaining useful information and building relationships are the main tasks. The second phase obtains original recommendation results through various recommendation algorithms. The third phase calculates the weights for each result. A weight measures the importance of a recommended result. The fourth phase is to get the final recommendation results through fusing the original results of various recommendation algorithms. The last phase is to show the recommendation results to the customer through an interactive interface, and record the customer's feedback information to correct the recommendation weights. Details are described next.

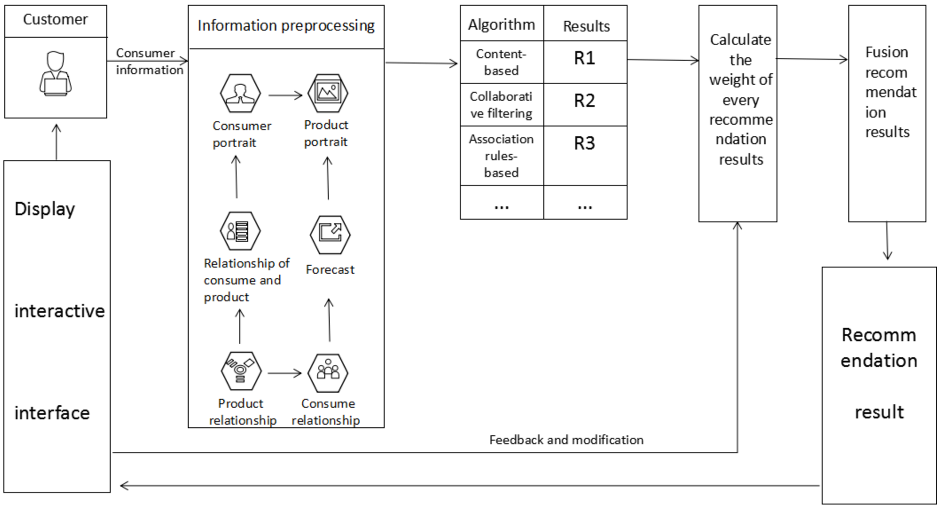

Figure 1. The proposed recommendation framework.

We propose a new method to enable customers to interact with recommendation systems and to create a feedback loop that incorporates customer feedback and input. Detailed description about the process of interactive interface and procedures with customers is shown in Figure 2. The customer-behavior node refers to the customer's current browsing/search products, click, evaluation and so on, which is used as a basis for calculating personalized recommendations. The information such as time, location and weather is denoted by the context node. The recommendation engine node gets the information from the customer behavior node and the context node to calculate the data for the medium node. The medium node represents data inferred from customer behavior and context data by the recommendation engine: a list of customers that are similar to the active customer is a typical example of such data.

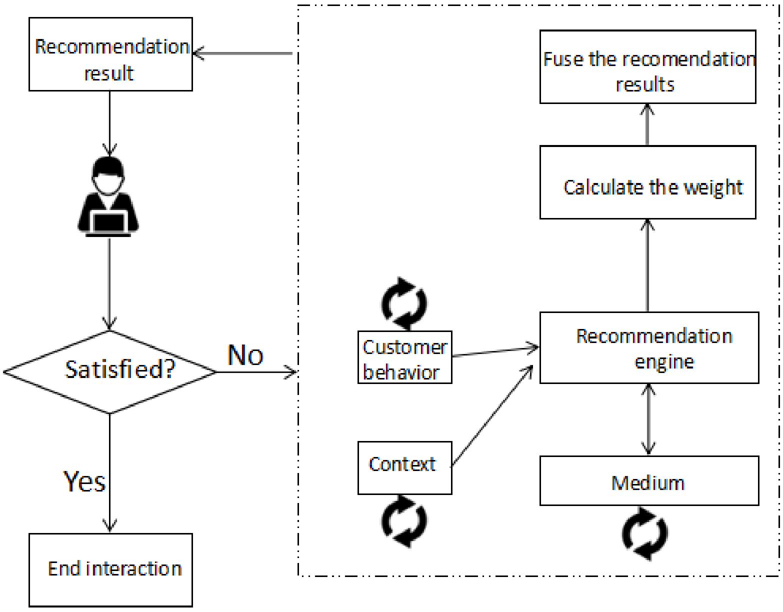

Figure 2. The framework of the interactive process.

The interactive process is illustrated by the arrows in Figure 2. The straight arrows indicate the data flow, while the revolving arrows refer to customer interactions with data elements of the different nodes. For instance, the revolving arrow of the customer behavior and context nodes represents interaction of customers with an interactive interface that represents customer behavior and context data, respectively. The revolving arrow of the medium node represents interaction of customers with an interactive interface of medium data, such as a list of like-minded customers. Customer feedback through the three nodes is transmitted to the recommendation engine through the straight arrows pointing towards the recommendation engine node. Then, the engine recalculates the weight and transmits the revised data to the medium and the recommendation result to the interactive interface. If the consumer is satisfied with the recommendation result, the interaction is ended. Otherwise, the interactive loop continues.

3.2. Obtain the Recommendation List

Obtaining the recommendation list requires three elements: products, customers and recommendation algorithms. The system mines customers' interest portraits from multiple dimensions, and precisely pulses customers' preference. Meanwhile, the system processes the products' information and gets the relation between the products. According to the needs of each recommendation algorithm, the system handles related information as the basis for subsequent analysis. C is a collection of all customers, and P is a collection of all the products. The utility function $u()$ is used to calculate the recommendation degree of the product p to the customer c. The goal of the recommendation algorithm is to find the product $p*$ that has the maximum recommendation degree. The formula is as follows:

$$\forall c \in C, p* = \text{argmax}_{p \in P} u(c, p). \tag{1}$$

3.2.1. CBF Algorithm

The CBF algorithm takes the feature as the basic unit to describe the product. It extracts features from customers' data and builds the customer preference model $ContentBasedProfile(c)$. Then, according to the location and frequency of different features in the product, it determines the match extent between the predicted product and the customer preference. The utility function can be calculated as follows:

$$u(c, p) = sim(ContentBasedProfile(c), Content(p)), \tag{2}$$

where $sim()$ is the similarity calculation, which can be calculated by Euclidean distance, Pearson similarity, vector cosine angle and so on. The object of the algorithm is mostly described by the text of the product, and the recommendation process does not require customer intervention. Therefore, it is a real-time process.

3.2.2. CF Algorithm

The CF algorithm, one of the most widely used techniques in recommendation systems, is based on a group of customers with the same interest [42]. CF uses the matrix of customers and products shown in Figure 3 to find the similarity between customers and products. Then, it combines the customer's historical information to obtain the recommendation degree of the product, and produces the recommendation result according to the recommendation value. CF can be divided into two types: user-based CF and item-based CF.

$$\begin{array}{c|cccc|}
 & \text{Product}_1 & \text{Product}_2 & \dots & \text{Product}_n \\
\text{Customer}_1 & R11 & R12 & \dots & R1n \\
\text{Customer}_2 & R21 & R22 & \dots & R2n \\
\vdots & \dots & \dots & \dots & \dots \\
\text{Customer}_m & Rm1 & Rm2 & \dots & Rmn
\end{array}$$

Figure 3. The matrix of customers and products.

The specific idea of user-based CF is as follows. (1) Calculate the similarity $sim(c, c')$ between the customer c and the other customers by using the customer's score for the product as a vector to determine the customer c preference similarity c'; (2) obtain the utility value of the customer c to the product p by the weighted average of the similarity c' and the similarity of the customer c. Therefore, the utility function is

$$u(c, p) = ave(\sum u(c', s) \times sim(c, c'))(c_i \in C). \tag{3}$$

Item-based CF is based on the following assumptions: Products that attract customers' interest must be similar to products that have been rated high previously. According to the hypothesis, calculate the similarity $sim(p, p')$ between the product p' evaluated by the customer c, and the predicted product p. Then, obtain the utility value of the customer c to the product p by the weighted average of the similarity $sim(p, p')$ and the score of the customer c to the product p. Therefore, the utility function is

$$u(c,p) = ave(\sum u(c,p') \times sim(p,p'))(c_i \in C). \tag{4}$$

The difference between the CBF algorithm and the CF algorithm is as follows: When mining the relationships among projects, the CBF algorithm needs no customer information except for the characteristics of the products. The item-based CF algorithm determines the relationship among the products using the customers' score.

3.2.3. Association Rules-Based Algorithm

The essence of recommendation based on association rules is to extract the relationship among products in a data set [43]. The recommendation algorithm has two key elements: $support(p \Rightarrow p') = S(p' \cup p)$ and $confidence(p \Rightarrow p') = S(p'|p)$, where p is a product to be predicted. Support is an important basis for data pruning, while confidence reflects the link between the product p and customer-interested product p', that is, the measure of utility value.

This article takes the above four common recommendation algorithms as an example to form factors of a hybrid recommendation strategy. Through the above four algorithms, the corresponding results are obtained, which are the preparations for the following fusion recommendation results.

3.3. Measure the Weights of Each Recommendation Result

After recommendation results of four algorithms are generated, the next step is to determine the weights of the corresponding results. In decision analysis, different results have different importance, and they play different supporting roles in our decision making. The weight of the original recommendation result is closely related to the accuracy of the final recommendation result. If the result of a recommendation algorithm has higher accuracy, then the algorithm should be more reliable and have greater weight in the decision process. For recommended data sources I, $I =< U, A \cup P >$, the importance of a subset of non-null conditional attributes $SA(\phi \subset SA \subseteq A)$ is:

$$\sigma(SA,P) = \frac{\gamma_A(P) - \gamma_{A-SA}(P)}{\gamma_A(P)} = 1 - \frac{\gamma_{A-SA}(P)}{\gamma_A(P)}, \tag{5}$$

where $\gamma_A(P)$ is the classification accuracy of attributes set A relative to class attributes P, and $\sigma(SA,D)$ describes the impact of deleting attributes subset SA on the classification of conditional attributes. The greater the impact, the more important the attribute subset.

The recommendation capabilities of the recommendation algorithms are different under different situations. Therefore, the weights should be different in the personalized recommendation process. A recommendation algorithm R is given the basis of the distribution of credibility, that is, the rule condition consists of a subset of conditional attributes. The weight of the result is

$$impa_i = \sigma(SA_i,P) = \frac{\gamma_A(P) - \gamma_{A-SA_i}(P)}{\gamma_A(P)} = 1 - \frac{\gamma_{A-SA_i}(P)}{\gamma_A(P)}. \tag{6}$$

Key steps of the method for determining weights are outlined below.

Step 1. Let $A = \left\{a_1, \ldots, a_i, \ldots a_{|A|}\right\}$ be the set of all input attributes in recommended data sources I, and let RC be the real recommended products. $|CES|$ is the number of recommendation algorithms.

Step 2. Train the neural network N to maximize the network accuracy with A as input and RC as output.

Step 3. For $j = 1, 2, \ldots, |CES|$, let N_j be a network whose connection weights are as follows:

(a) For all the inputs except $\left\{a_{j,1}, a_{j,2}, \ldots a_{j,|SAj|}\right\}$, assign the connection weights of N_j equal to the weights of N.

(b) Set the connection weights of $\left\{a_{j,1}, a_{j,2}, \ldots a_{j,|SAj|}\right\}$ to zero.

Compute the output of network N_j.

Step 4. Compute the influence of SA_j to the network accuracy.
Step 5. If $j \geq |CES|$, go to Step 6, otherwise, set $j = j + 1$ and go to Step 3.
Step 6. The derived $\left\{w_1, \ldots, w_j, \ldots w_{|CES|}\right\}$ are the weights of the recommendation algorithms.

In the above step, we first find a trained neural network N with the entire set of attributes A, $A = \left\{a_1, \ldots, a_i, \ldots a_{|A|}\right\}$, as its input. For each recommendation algorithm, suppose the condition attributes consists of SA_j, $SA_j = \left\{a_{j,1}, a_{j,2}, \ldots a_{j,|SAj|}\right\}$. We calculate the prediction result of the neural network after deleting $\left\{a_{j,1}, a_{j,2}, \ldots a_{j,|SAj|}\right\}$, as described in Step 3. That is, the connection weights of the $\left\{a_{j,1}, a_{j,2}, \ldots a_{j,|SAj|}\right\}$ are set to 0, and the connection weights of other attributes are the same as those of the trained neural network N. Then, compute the output of network N_j, whose connection weights are stated as above. Last, as Step 4 says, compare the difference of accuracy between network N_j and N. The difference of accuracy can be regarded as the influence of the algorithm on the result. The greater the influence of the attribute set on the result, the bigger is its weight.

3.4. Fuse the Results

Through the acquisition of recommendation results and their importance, we obtain the representation of the recommendation scheme as follows:

$$< o, R_i, w_i > \rightarrow < \{p_{i_1}, conf_{i_1}\}, \ldots, \left\{p_{i_j}, conf_{i_j}\right\}, \ldots, \{p_{i_m}, conf_{i_m}\} >, \tag{7}$$

where $i = 1, 2, \ldots, t$, o is the target consumer, R_i is the one recommendation algorithm, w_i is the weight of R_i, p_{i_j} is the recommended target product, and $conf_{i_j}$ is the basic credibility that R_i thinks o buys p_{i_j}. If every recommendation algorithm gives different recommendation results, how do you recommend products to consumers at this time? In this paper, we use the synthetic formula of evidence theory to synthesize the recommendation results to form the final recommendation scheme. It realizes the product recommendation by using a variety of algorithms.

In the process of information fusion, evidence theory first defines an identifiable framework $\Theta = \{\theta_1, \theta_2, \ldots, \theta_n\}$ to describe n basic assumptions about decision problems. Based on the identifiable framework, we can obtain the dense of underlying assumption, denoted 2^Θ. The basic idea of evidence theory is that, for a decision object, the basic credibility distribution is given by the conclusion whose concentration is 2^Θ. The basic credibility assignment is any function as follows: $m : 2^\Theta \rightarrow [0, 1]$, and that has the nature: $m(\phi) = 0$, $\sum\limits_{\Delta \in 2^\Theta} m(\Delta) = 1$. Given the basic credibility assignment of m, we can obtain the confidence function $bel : 2^\Theta \rightarrow [0, 1]$ and the likelihood function $pl : 2^\Theta \rightarrow [0, 1]$:

$$bel(\theta) = \sum_{\Delta \subseteq \theta} m(\Delta), \tag{8}$$

$$pl(\theta) = \sum_{\Delta \cap \theta \neq \phi} m(\Delta) = 1 - bel(\Theta - \theta). \tag{9}$$

In the framework of traditional evidence theory, the synthesis rule is the most classical and commonly used synthesis method that has the characteristics of commutativity and associativity. The formula is as follows:

$$m(\theta) = \frac{\sum\limits_{\theta_i \cap \theta_j = \theta} m_1(\theta_i) \times m_2(\theta_j)}{1 - \sum\limits_{\theta_i \cap \theta_j = \phi} m_1(\theta_i) \times m_2(\theta_j)}, \tag{10}$$

where m_1 and m_2 are the basic credibility assignments given by any two evidence sources, θ_i and θ_j are hypotheses about the evidence sources and $\sum\limits_{\theta_i \cap \theta_j = \phi} m_1(\theta_i) \times m_2(\theta_j)$ stands for the inconsistent information of two evidence sources.

Let us take an example to present the fusion mechanism. Companies are making decisions about which computers are recommended to Lily. We use three algorithms mentioned in Section 3.2 as the recommendation methods. The three recommended computers are defined as A, B, C. The reliability given by the algorithms is shown in the Table 2.

Table 2. Belief degrees of customer purchase.

Product	Algorithm	Recommend	Don't Recommend	Uncertain
A	CBF	65%	30%	5%
	CF	67%	30%	3%
	Association rules	63%	34%	3%
B	CBF	66%	32%	2%
	CF	65%	30%	5%
	Association rules	67%	29%	4%
C	CBF	56%	40%	4%
	CF	58%	30%	12%
	Association rules	62%	33%	5%

After getting a list of recommended products from each recommendation algorithm, we use the method mentioned in Section 3.3 to obtain the recommendation weights. Table 3 shows the results of each recommendation algorithm and their corresponding weights.

Table 3. Recommendation results and the corresponding weights.

Product	Recommend	Don't Recommend	Uncertain	Algorithm	Weights
A	0.65	0.30	0.05	CBF	0.37
B	0.66	0.32	0.02		
C	0.56	0.40	0.04		
A	0.67	0.30	0.03	CF	0.51
B	0.65	0.30	0.05		
C	0.58	0.30	0.12		
A	0.63	0.34	0.03	Association rules	0.12
B	0.67	0.29	0.04		
C	0.62	0.33	0.05		

Evidence theory is used to synthesize the basic credibility assignment. The results are shown in Table 4. We can get personalized recommendation results from different recommendation algorithms whose results are occasionally conflicting. For example, the recommendation result of algorithm R_1 and algorithm R_3 is B, but the recommendation result of algorithm R_2 is A. Through the fusion of the initial results from each recommendation algorithm, the final result is A (0.658 > 0.656 > 0.577).

Table 4. The fusion results of each recommendation algorithm.

Product	Recommend	Don't Recommend	Uncertain
A	0.658	0.305	0.037
B	0.656	0.306	0.038
C	0.577	0.341	0.082

By using the proposed method, we get the products that are suitable for the customer. In our example, only three product's recommendation results are obtained. In real-life transactions, there are thousands of goods in the database. Space for the recommended products is limited. The system can choose the recommended products according to their recommendation probabilities. The recommendation results of the hybrid algorithms are shown to the consumer through the interactive interface. Then, according to the feedback behavior of the consumer, the weight of each result in Section 3.3 is changed dynamically. Furthermore, the original results are re-fused until the final results are optimal.

4. Experimental Results

4.1. Data Preparation

The database of the women's clothing online shop is chosen as a research example, which is from Taobao.com. Taobao.com is a large network retail business circle under Alibaba Group. It is the most-visited online retail website in China. Women's clothing online shops have a large scale in Taobao.com. Their products are often in large quantities, and the quality is difficult to identify. Research evidence has suggested that an e-commerce recommendation system is critical for consumer purchase decisions and product sales [44]. An e-commerce recommendation system recommends products that the customer will find most valuable among the available products. Thus, an e-commerce recommendation system is essential to help consumers find valuable products and avoid information overload. We obtained data from the online shop selling fashion apparel. The number of successful transactions has been about 800,000 since the beginning of the store in 2009. In this paper, we select the data from the shop's transaction database in a period of time to test the effectiveness of the proposed algorithm.

The subjects of the experiment are 200 consumers. Of the customers, 120 bought clothes online for the first time in the online store. 50 consumers purchased two times during the experiment. The remaining 30 customers shopped three times or more at the store during the experiment. There are four kinds of recommendation algorithms: CBF, item-based CF, user-based CF and association rules. In the test experiments, we compared the above four algorithms and the hybrid algorithm proposed in this paper. Comparison items include the consumer coverage, the consumer discovery accuracy, the recommendation recall and the recommendation speed.

4.2. The Consumer Coverage

The consumer coverage refers to:

$$Coverage(t) = \sum (RS(t) \cap RR(t)) / \sum RS(t), \tag{11}$$

where $t \geq 0$, $RS(t)$ is the recommendation resource set at time t equals t in the recommendation system, and $RR(t)$ is the resource set of the consumer selection. The higher the consumer coverage, the stronger the coverage ability of the recommendation system to the consumer's needs and features. The comparison of experiment results on consumer coverage is shown in Table 5 and Figure 4.

Table 5. The consumer coverage of each recommendation algorithm.

Algorithm	**Data Statistics Time (24 H × 7)**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
proposed algorithm	18%	28%	36%	43%	51%	60%	67%	73%	80%	79%
user-based CF	16%	22%	29%	37%	44%	52%	59%	61%	70%	65%
association rules	17%	27%	34%	41%	50%	58%	65%	70%	78%	75%
CBF	17%	25%	32%	39%	48%	57%	64%	68%	77%	72%
item-based CF	16%	24%	31%	38%	47%	55%	61%	65%	73%	68%

Because of the diversity of consumer preferences, there are often small probability categories in the personalized recommendation process. The coverage is an important index of the rule recommendation ability. The recommendation rules should contain concise recommendation products and provide personalized recommendation for consumers' different needs. As can be seen from Figure 4, with the continuous increase of the customer information in the recommendation system, the customer coverage increases continuously until the peak value. In addition, the proposed algorithm is better than other algorithms, and user-based CF is the worst. This is because the core of user-based CF is to identify and explore specific consumer groups. The algorithm forms total customer preference forecasts through the analysis of characteristics of consumers' interest and the fusion of similar consumers' evaluation of products. However, the cost of system computing and storage increases dramatically following the increase in the number of products and consumers. Furthermore, the consumer coverage declines. The proposed method can reduce the weight of the user-based CF to improve the consumer coverage. From the point of view of the coverage speed, the advantages of the proposed algorithm are more obvious, particularly with the rapid changes of consumer characteristics under corporate advertising, promotions and other activities. Therefore, the proposed method can accurately recommend products that consumers are interested in, to avoid the loss of consumers and enhance corporate profits.

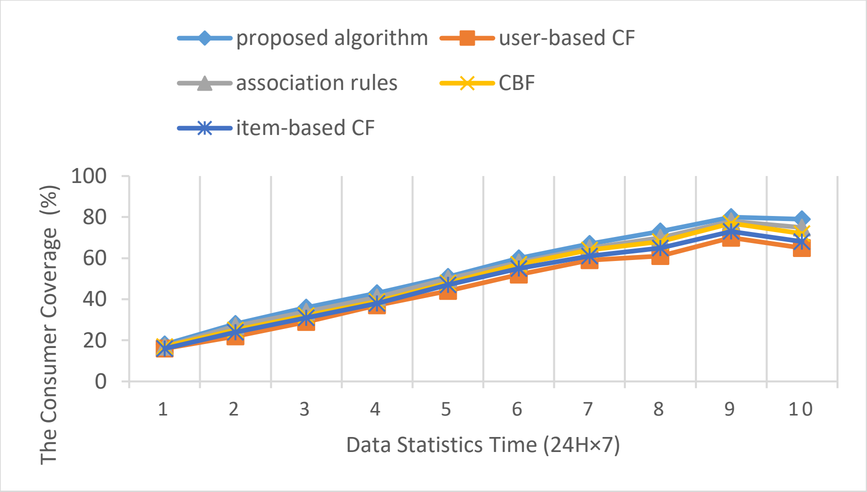

Figure 4. Comparison experiment of the consumer coverage.

4.3. The Consumer Discovery Accuracy

The consumer discovery accuracy is defined as the proportion of selected resources in recommended resources. It becomes very important for the recommendation system. If the system recommends a product that is not required for consumer demand, the recommendation system will help consumers make decisions and improve the purchase rate. In this comparative experiment, we first carried out an experimental analysis of 200 subjects. However, due to limitations of the small sample size, the differences of experimental results were not obvious. Thus, we chose 300 other customers from the shop's transaction database as the experimental subjects. The total number of customers increased from 200 to 500. Among them, the proportion of all types of consumers remains unchanged. A comparison of five algorithms is shown in Table 6 and Figure 5.

Table 6. The consumer discovery accuracy of each recommendation algorithm.

Algorithm	Number of Customers									
	50	**100**	**150**	**200**	**250**	**300**	**350**	**400**	**450**	**500**
proposed algorithm	20%	35%	50%	70%	75%	81%	78%	75%	70%	67%
CBF	17%	27%	41%	60%	64%	71%	61%	52%	48%	41%
item-based CF	18%	33%	48%	67%	71%	79%	71%	65%	59%	55%
user-based CF	18%	32%	47%	66%	68%	77%	69%	61%	54%	49%
association rules	17%	30%	44%	63%	67%	75%	65%	58%	51%	44%

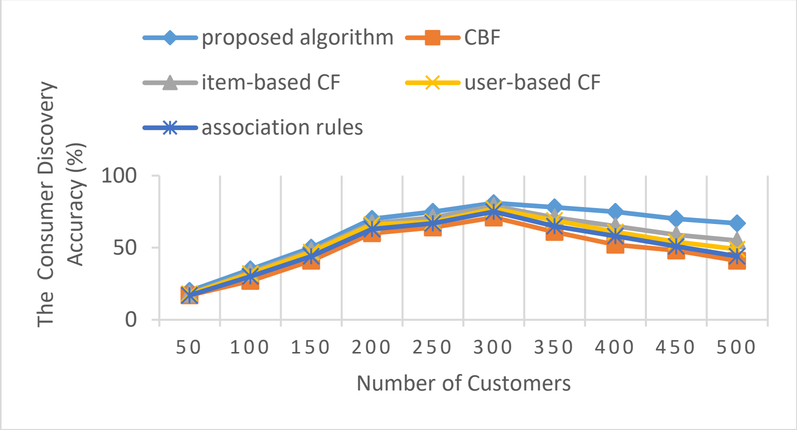

Figure 5. Comparison experiment of the consumer discovery accuracy.

The above experiment compares the consumer discovery accuracy of four algorithms and the proposed algorithm by using different sizes of data. When the number of customers is 300, the accuracy for the proposed algorithm is 81%, as shown in Table 6. From Figure 5, we can see that the proposed method in terms of accuracy is higher than that of the other four algorithms. In the four algorithms, the consumer discovery accuracies become relatively stable with the continuous enrichment of customer information, and then begin to decrease after reaching their peak values. However, the proposed method allows consumers to interact with the system dynamically to change weights. Consumer feedback can adjust the recommended products in a timely manner, so the accuracy of the peak was higher and the accuracy decreased more slowly. This means that the strategy using the hybrid algorithm to analyze consumer behavior is more accurate. It has a positive impact on the quality of the recommendation rules. This advantage is more pronounced after the number of consumers increases to 300, as shown in Figure 5. On the one hand, the proposed algorithm can explore products similar to the customer's previous interests, according to his/her browsing record. On the other hand, it can tap new requirements and interest points for the customer by fusing similar customers. In addition, we find that the consumer discovery accuracies of the five algorithms to old customers are not very different. However, with new customers, the different consumer discovery accuracies have a greater effect. Therefore, the recommendation system should be combined with a customer relationship management system in the future research.

4.4. The Recommendation Recall

The recommendation recall is defined as the proportion of products selected by the customer to the total number of products recommended to the customer. In this comparative experiment, the situation is the same as that of Section 4.3. The differences of the experimental results are not obvious due to the limit of the number of subjects. This problem may reduce the experimental validity, so we further expanded the number of experimental subjects. The total number of customers increased from 200 to

500. Among them, the proportion of all types of consumers remained unchanged. The experimental results are shown in Table 7 and Figure 6.

Table 7. The recommendation recall of each algorithm.

Algorithm	Number of Customers									
	50	100	150	200	250	300	350	400	450	500
proposed algorithm	18%	26%	35%	41%	45%	48%	45%	43%	41%	39%
association rules	17%	24%	33%	39%	41%	45%	42%	38%	34%	29%
item-based CF	17%	22%	30%	35%	39%	42%	40%	35%	31%	25%
CBF	17%	22%	29%	35%	38%	42%	39%	35%	31%	25%
user-based CF	16%	21%	29%	34%	38%	42%	39%	34%	30%	24%

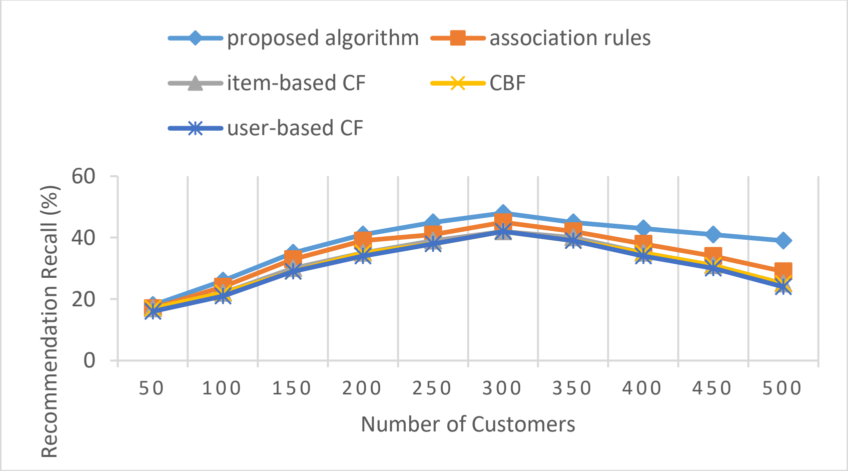

Figure 6. Comparison experiment of the recommendation recall.

As Figure 6 shows, the recommendation recall rate of the five algorithms tends to be stable with the richness of customers and resources. Moreover, due to the stability of customers' needs, the recommendation recall begins to decline. However, the proposed algorithm has more advantages in the recommendation recall criterion, and maintains a high recall for a long time. This is due to the fact that consumer demand is always changing, while with traditional recommendation systems it is hard to recommend something new [9,10]. The proposed method combines a hybrid algorithm and interactive assignment together, achieves high performance and accuracy, and alleviates the cold-start problems in recommendation systems to some extent. The results demonstrate that consumers can enjoy the convenient and high-quality service produced by the recommendation system using the proposed algorithm. A good recommendation system not only helps consumers to find useful information, but also helps retailers to show the products of interest to consumers, providing both consumers and retailers with a win-win situation.

4.5. The Recommendation Speed

We define the recommendation speed as the time from data input to the recommended products' output. Experimental comparison of the proposed method and the other four methods follows. The experiment shows that the time consumed in the proposed method is more than the other four methods, and the difference increases more and more as the number of consumers increases. The recommendation speeds of the five algorithms are shown in Table 8 and Figure 7.

Table 8. Comparison experiment of the recommendation speed according to the number of customers.

Customers	Proposed Algorithm (s)	CBF (s)	Association Rules (s)	Item-Based CF (s)	User-Based CF (s)
50	0.33	0.28	0.29	0.29	0.31
100	0.81	0.49	0.49	0.51	0.54
150	1.23	0.71	0.78	0.79	0.81
200	1.78	0.99	1.02	1.04	1.18
300	2.45	1.48	1.51	1.54	1.64
350	2.79	1.76	1.81	1.85	1.96
400	3.21	2.01	2.04	2.16	2.23
450	3.78	2.21	2.47	2.56	2.67
500	4.44	2.52	2.81	2.94	2.99

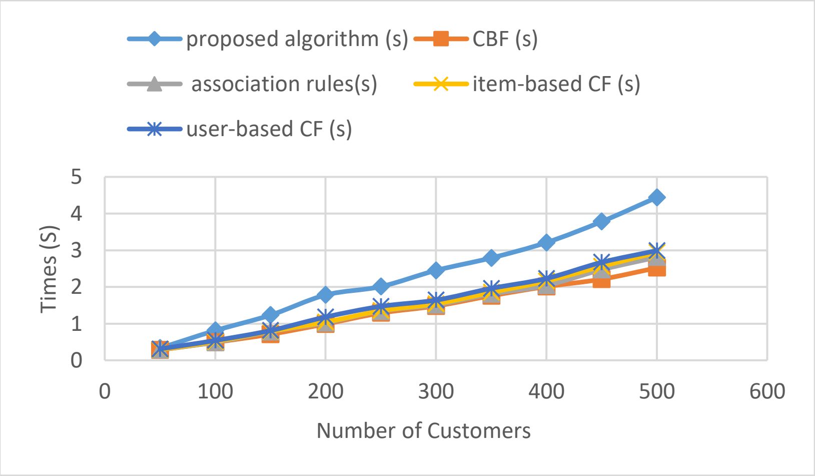

Figure 7. A comparison of the recommendation speed according to the number of customers.

The above experiment compares the time consumed by the proposed algorithm and the other four algorithms by expanding the consumer number. Experiments show that the time consumed in the proposed algorithm is more than that in other algorithms, and the difference increases more and more as the number of customers increases. Thus, the proposed algorithm cannot improve the speed of recommendation. This is because the proposed algorithm needs to constantly interact with consumers to change the weight of each recommendation algorithm, taking more time.

5. Conclusions and Discussion

The recommendation system is an important tool to offer personalized service. Current recommendation systems rarely use interactive methods to dynamically change the weights of recommendation algorithms, so as to achieve accurate recommendation. We argue that a truly successful recommendation system should be based on a personalized recommendation algorithm, to form a long-term stable relationship with consumers. The main contributions of this research are twofold. Firstly, we should pay attention to the fact that it is sometimes hard for consumers to express their preferences. Consumers need to interact with the system and evaluate the recommendation products given by the system. Then, the system runs repeated cycles of interactive recommendations to ensure the accuracy of recommendations in data mining. Accuracy is very important for the recommended products. Accurate recommendations can not only improve customer satisfaction, but also improve corporate sales. Thus, online retailers are advised to provide these recommendations to increase sales. Retailers should pay attention to customer feedback behavior through the interactive interface, so as to guide and cultivate consumption.

Secondly, to achieve accurate recommendations, we propose a more effective model using a combination of recommendation algorithms in data mining. The hybrid recommendation method complements the advantages of the existing recommendation algorithms, obtains original results with each algorithm, and then establishes the weights according to the interactive result with consumers. Finally, the synthetic formula of evidence theory is used to fuse the original results to obtain the final recommendations. Meanwhile, our study also offers important implications for e-commerce platform providers regarding the design of product-recommendation systems. E-commerce platform operators could redesign the recommendation algorithm of recommendation systems based on the above insights, to select more-appropriate products to recommend.

Although this study has highlighted several notable results and contributions, certain limitations also need to be recognized. On the one hand, based on the experimental results of Section 4.5, we find that the proposed algorithm has no advantage in the recommendation speed, being more time-consuming than the other ones. Therefore, we intend to improve the speed of recommendation in future research. On the other hand, we chose only four algorithms as the basis of the hybrid algorithm, and we can add more recommendation algorithms in future research.

Acknowledgments: This study was supported by funding from Cultivating program of excellent innovation team of Chengdu University of Technology (KYTD201406); New type management think tanks of Chengdu University of Technology (2017).

Author Contributions: Minxi Wang conceived the overall idea of the article and analyzed the data. Xin Li designed the experiments and contributed analysis tools. Yan Guo performed the experiments and wrote the paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Voelker, T.A.; Steel, D.; Shervin, E. Internet presence as a small business capability: The case of mobile optimization. *J. Small Bus. Strategy* **2017**, *27*, 90–103.
2. Orlandi, L.B. Organizational capabilities in the digital era: Reframing strategic orientation. *J. Innov. Knowl.* **2016**, *1*, 156–161. [CrossRef]
3. Marung, U.; Theera-Umpon, N.; Auephanwiriyakul, S. Top-N recommender systems using genetic algorithm-based visual-clustering Methods. *Symmetry* **2016**, *8*, 54. [CrossRef]
4. Ricci, F.; Rokach, L.; Shapira, B. *Introduction to Recommender Systems Handbook*; Springer: Berlin, Germany, 2011; pp. 1–35.
5. Ullah, F.; Sarwar, G.; Lee, S. N-screen aware multicriteria hybrid recommender system using weight based subspace clustering. *Sci. World J.* **2014**, *2014*, 679–849. [CrossRef] [PubMed]
6. Garfinkel, R.; Gopal, R.; Pathak, B.; Yin, F. Design of a shopbot and recommender system for bundle purchases. *Decis. Support Syst.* **2006**, *42*, 1974–1986. [CrossRef]
7. Lee, J.; Lee, D.; Lee, Y.C.; Hwang, W.S.; Kim, S.W. Improving the accuracy of top-N recommendation using a preference model. *Inf. Sci.* **2016**, *348*, 290–304. [CrossRef]
8. Zenebe, A.; Norcio, A.F. Representation, similarity measures and aggregation methods using fuzzy sets for content-based recommender systems. *Fuzzy Sets Syst.* **2009**, *160*, 76–94. [CrossRef]
9. Lops, P.; Gemmis, M.; de Semeraro, G. Content-based recommender systems: State of the art and trends. In *Recommender Systems Handbook*; Ricci, F., Rokach, L., Shapira, B., Kantor, P.B., Eds.; Springer: New York, NY, USA, 2011.
10. Herlocker, J.L.; Konstan, J.A.; Loren, J.; Terveen, G.; Riedl, T. Collaborative filtering recommender systems. *ACM Trans. Inf. Syst.* **2004**, *22*, 5–53. [CrossRef]
11. Melville, P.; Mooney, R.J.; Nagarajan, R. Content-boosted collaborative filtering for improved recommendations. In Proceedings of the Eighteenth National Conference on Artificial Intelligence, Edmonton, AB, Canada, 28 July–1 August 2002.
12. Liang, T.P.; Yang, Y.F.; Chen, D.N.; Ku, Y.C. A semantic-expansion approach to personalized knowledge recommendation. *Decis. Support Syst.* **2008**, *45*, 401–412. [CrossRef]
13. Adomavicius, G.; Tuzhilin, A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Trans. Knowl. Data Eng.* **2005**, *17*, 734–749. [CrossRef]

14. Lampropoulos, A.S.; Sotiropoulos, D.N.; Tsihrintzis, G.A. Evaluation of a cascade hybrid recommendation as a combination of one-class classification and collaborative filtering. In Proceedings of the International Conference on Tools with Artificial Intelligence, Athens, Greece, 7–9 November 2012; Volume 1, pp. 674–681.
15. Kardan, A.A.; Ebrahimi, M. A novel approach to hybrid recommendation systems based on association rules mining for content recommendation in asynchronous discussion groups. *Inform. Sci.* **2013**, *219*, 93–110. [CrossRef]
16. Liu, H.F.; Hu, Z.; Mian, A.; Tian, H.; Zhu, X.Z. A new user similarity model to improve the accuracy of collaborative filtering. *Knowl. Based Syst.* **2014**, *56*, 156–166. [CrossRef]
17. Mishra, R.; Kumar, P.; Bhasker, B. A web recommendation system considering sequential information. *Decis. Support Syst.* **2015**, *75*, 1–10. [CrossRef]
18. Lee, Y.H.; Hu, P.J.H.; Cheng, T.H.; Hsieh, Y.F. A cost-sensitive technique for positive-example learning supporting content-based product recommendations in B-to-C e-commerce. *Decis. Support Syst.* **2012**, *53*, 245–256. [CrossRef]
19. Goldberg, D.; Nichols, D.; Oki, B.M.; Terry, D. Using collaborative filtering to weave an information tapestry. *Commun. ACM* **1992**, *35*, 61–70. [CrossRef]
20. Nilashi, M.; Ibrahim, O.B.; Ithnin, N.; Sarmin, N.H. A multi-criteria collaborative filtering recommender system for the tourism domain using Expectation Maximization (EM) and PCA–ANFIS. *Electron. Commer. Res. Appl.* **2015**, *14*, 542–562. [CrossRef]
21. Thabtah, F. A review of associative classification mining. *Knowl. Eng. Rev.* **2007**, *22*, 37–65. [CrossRef]
22. Benlian, A.; Titah, R.; Hess, T. Differential effects of provider recommendations and consumer reviews in e-commerce transactions: An experimental study. *J. Manag. Inf. Syst.* **2012**, *29*, 237–272. [CrossRef]
23. Hung, L.P. A personalized recommendation system based on product taxonomy for one-to-one marketing online. *Expert Syst. Appl.* **2005**, *29*, 383–392. [CrossRef]
24. Deshpande, M.; Karypis, G. Item-based top-N recommendation algorithms. *ACM Trans. Inf. Syst.* **2004**, *22*, 143–177. [CrossRef]
25. Ullah, F.; Lee, S.C. Social content recommendation based on spatial-temporal aware diffusion modeling in social networks. *Symmetry* **2016**, *8*, 89. [CrossRef]
26. Fu, X.; Han, G.H. Trust-embedded information sharing among one agent and two retailers in an order recommendation system. *Sustainability* **2017**, *9*, 710. [CrossRef]
27. Scholz, M.; Dorner, V.; Franz, M.; Hinz, O. Measuring consumers' willingness to pay with utility-based recommendation systems. *Decis. Support Syst.* **2015**, *72*, 60–71. [CrossRef]
28. Chen, M.H.; Teng, C.H.; Chang, P.C. Applying artificial immune systems to collaborative filtering for movie recommendation. *Adv. Eng. Inform.* **2015**, *29*, 830–839. [CrossRef]
29. Cheung, K.W.; Kwok, J.T.; Law, M.H.; Tsui, K.C. Mining customer product ratings for personalized marketing. *Decis. Support Syst.* **2003**, *35*, 231–243. [CrossRef]
30. Puglisi, S.; Parra-Arnau, j.; Forné, j.; Rebollo-Monedero, D. On content-based recommendation and user privacy in social-tagging systems. *Comput. Stand. Inter.* **2015**, *41*, 17–27. [CrossRef]
31. Guo, Y.; Wang, M.X.; Li, X. Application of an improved Apriori algorithm in a mobile e-commerce recommendation system. *Ind. Manag. Data Syst.* **2017**, *117*, 287–303. [CrossRef]
32. Peng, Y.; Cheng, X.P. Item-based collaborative filtering algorithm using attribute similarity. *Comput. Eng. Appl.* **2007**, *43*, 144–147.
33. Wu, Y.P.; Zheng, J.G. Collaborative filtering recommendation algorithm on improved similarity measure method. *Comput. Appl. Softw.* **2011**, *28*, 7–8.
34. Hussein, T.; Linder, T.; Gaulke, W.; Ziegler, J. Hybreed: A software framework for developing context-aware hybrid recommender systems. *User Model. User Adapt. Interact.* **2014**, *24*, 121–174. [CrossRef]
35. Zheng, L.; Zhu, F.X.; Huang, S.; Xie, J. Utility-Based link recommendation for online social networks. *Manag. Sci.* **2017**, *414*, 1–18.
36. Swearing, K.; Sinha, R. Beyond algorithms: An HCI perspective on recommender systems. In Proceedings of the ACM SIGIR Workshop on Recommender Systems, New York, NY, USA, September 2001; pp. 1–11.
37. Felfernig, A.; Teppan, E.; Gula, B. Knowledge-based recommender technologies for marketing and sales. *Int. J. Pattern Recogn.* **2007**, *21*, 333–354. [CrossRef]

38. Verbert, K.; Parra, D.; Brusilovsky, P.; Duval, E. Visualizing recommenda-tions to support exploration, transparency and controllability. In Proceedings of the International Conference on Intelligent User Interfaces (IUI '13), Santa Monica, CA, USA, 19–22 March 2013; p. 351.
39. He, C.; Parra, D.; Verbert, K. Interactive recommender systems: A survey of the state of the art and future research challenges and opportunities. *Expert Syst. Appl.* **2016**, *56*, 9–27. [CrossRef]
40. Alves, H.; Ferreira, J.J.; Fernandes, C.I. Customer's operant resources effects on co-creation activities. *J. Innov. Knowl.* **2016**, *1*, 69–80. [CrossRef]
41. Adomavicius, G.; Tuzhilin, A. Context-aware recommender systems. In *Recommender Systems Handbook*; Ricci, F., Rokach, L., Shapira, B., Kantor, P.B., Eds.; Springer: Berlin, Germany, 2012; pp. 217–253.
42. Salter, J.; Antonopoulos, A. Cinema screen recommender agent: Combining collaborative and content-based filtering. *IEEE Intell. Syst.* **2006**, *21*, 35–41. [CrossRef]
43. Wang, H.M.; Li, J. An Improved Apriori Algorithm Based On the Boolean Matrix and Hadoop. *Procedia Eng.* **2011**, *15*, 1827–1831.
44. Chen, Y.; Xie, J. Online consumer review: Word-of-mouth as a new element of marketing communication mix. *Manag. Sci.* **2008**, *54*, 477–491. [CrossRef]