Peer-reviewed version available at *Information* **2020**, *11*, 171; doi:10.3390/info11030171

Article

Customer loyalty improves the effectiveness of recommender systems based on complex network

Yun Bai [1,*], Suling Jia[1] , Shuangzhe Wang[1] and Binkai Tan[1]

[1] School of Economics and Management, Beihang University, Beijing 100191, China ; baiyun12138@buaa.edu.cn;

* Correspondence: baiyun12138@buaa.edu.cn

Abstract: A good recommender system can infer customers' preferences based on their historical purchase records, and recommend products that the customers may be interested in, saving them a lot of time and energy. For enterprises, it is difficult to recommend accurately to each customer, and the bad recommendation may be counterproductive. Customer loyalty is an indicator that measures the preference relationship between customers and products in the field of marketing. A hypothesis is proposed in this study: if companies can divide customers into different groups based on customer loyalty, the recommendation effect on certain groups is better than that on overall customers.

In this study, customer loyalty is measured by four features of the RFML model. All customers are viewed as points on a four-dimensional space, which are clustered by the k-means model. Two recommendation algorithms based on complex networks are tested: recommendation algorithm based on bipartite graph and PersonalRank (BGPR), and recommendation algorithm based on a single vertex set network and DeepWalk (SVDW). The experimental results show that customer loyalty has improved the effectiveness of the two algorithms over 14%, and the recommendation effect is the best on customer groups with a loyalty level of 4 (the highest level is 5). The recommendation algorithms with customer loyalty are better than using them alone.

Keywords: Recommender Systems; Customer Loyalty; Complex Networks

1. Introduction

Advances in Internet technology have increased the way companies communicate with customers. The marketing strategy of modern enterprises has gradually changed from traditional profit-oriented to customer-oriented. Customer loyalty means that customers are willing to purchase products or services of the same brand continuously in the future, according to [1], customers' preferences will not change due to changes in the market or advertising of other products. If familiar with the distribution of customer loyalty, companies will improve the quality of customer service, meet the personalized needs of high-loyal customers, and increase customers' satisfaction.

The huge amount of shopping information in various e-commerce platforms expands the customer's choice greatly, but it also brings the "information overload" problem [2]. Customers spend too much time reading and retrieving information, and can not find the products they want quickly and accurately. The recommender systems solve this problem. It analyzes the past purchase records of customers, and predicts customers' preferable products. By presenting these predictions in front of customers, the e-commerce platform has implemented the product recommendation function.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 February 2020 doi:10.20944/preprints202002.0367.v1

Peer-reviewed version available at Information 2020, 11, 171; doi:10.3390/info11030171

The model and theory of complex networks appeared at the end of the 20th century. Its basic idea is to abstract entities of a complex system into nodes and relationships between entities into edges. It can represent complex systems in real life as mathematical models through network structures. In recent years, many scholars have studied recommender systems based on complex networks and achieved good results [3-6]. A product recommendation model based on a complex network establishes a network structure to describe the relationship between customers and products and calculates the similarity between products to recommend suitable ones to target customers based on their historical purchase records. Currently, few studies have combined customer loyalty with complex network recommendations. Customer loyalty is an important indicator of the relationship between customers and products. Therefore, it's a meaningful study to consider the impact of customer loyalty when building a recommender system based on a complex networks. In this study, the exogenous variable of customer loyalty is added into the recommender system based on complex network, and the recommendation effect is improved. Specifically, the main contents of this study are as follows:

1. The RFML model is used for feature extraction of customer loyalty, and the k-means algorithm is used to cluster the features of customer loyalty, thereby distinguishing customers with different loyalty levels.
2. A recommender system based on the bipartite graph is constructed, and the PageRank method is used for random walks. After adding the exogenous variable of customer loyalty, the system compares the recommendation accuracy rate of customers with different loyalty levels and of the overall customers.
3. A recommender system based on a single vertex set network and DeepWalk is constructed and the accuracy of adding the exogenous variable of customer loyalty is analyzed.

The remaining of this paper is organized as follows: In Section 2, related works are introduced. Section 3 explains the proposed methods. In Section 4, the experimental results and analytical discussions are described. Finally, Section 5 concludes all the work.

2. Related works

2.1. Customer loyalty

Customer loyalty is a key factor for companies to enhance their competitive strength. [7] proposed that loyal customers would have a greater willingness to purchase a larger number of products, pay less attention to the price of products, and recommend the brand to his family and friends. An important issue for customer loyalty is how to quantify and calculate it. There are three methods used commonly: expert experience scoring, long-term forecasting, and short-term forecasting.

1. Expert experience scoring

Each expert scores the loyalty degree of customers based on their purchase records. The advantage of this method is that it's comprehensive and the results have application significance. But for different indicators, expert ratings may be subjective, and data processing methods are cumbersome. In [8], a customer segmentation method based on the customer life cycle is proposed. The characteristics of data and the experience of business experts were taken into account when constructing the judgment matrix. When designing the brand relationship questionnaire, 13 experts (all with a Ph.D. in marketing) were invited to evaluate the items of the dimension "love and passion" in [9].

2. Long-term forecasting

A common long-term forecasting method used to calculate customer loyalty is Customer Lifetime Value (CLV). It has been described in [10] that Lifetime Value is the total revenue from customers minus the cost of attracting, selling, and servicing them, considering time. In [11] , customer loyalty depends on the ability of customers to increase enterprises' income and reduce enterprises' costs at different stages in their lifetime. Experimental results based on different data sets in the literature [12-14] suggest that customer loyalty is a driver of CLV and a predictor of long-term profitability to a firm.

The concept of net present value in financial management is introduced into the calculation of CLV [15], which causes difficulties in estimating the parameters of the model [16]. CLV also ignores some non-money costs in the company and does not consider the randomness of the amount the customers spend, which makes CLV less accurate when simulating the real world. According to [17], many companies sell products to customers through retailers, lacking a process of direct communication with customers; due to customers' privacy, insufficient data processing capabilities, and the high cost of collecting and analyzing data, CLV analysis is difficult to implement.

3. Short-term forecasting

The RFM model is often used to analyze customer loyalty. It utilizes the customers' historical purchasing records to predict their likely consumption behavior for a short period in the future. The RFM model has three key indicators. R represents the time of last consumption, F is the frequency of consumption, and M is the total consumption. Because the three indicators of the RFM model are easy to obtain from customers' consumption records, they are widely used in marketing [18-21]. The RFM model can categorize customers while analyzing customer loyalty. Its main idea is to classify customers by calculating the RFM value of the customers and comparing it with a threshold standard. [22] calculate RFM values of customers in duty-free stores, and then subdivide customers into different groups through the Self-Organizing Map (SOM) method. The customer quintiles method has been introduced in [19], which assigns a score of 1 to 5 to each indicator. The higher the score of an indicator, the better the customer's performance on this indicator, so this method divides customers into 125 categories (5 * 5 * 5). [23] proposed the RFML model based on RFM. A new indicator, the total length of time from the customer's initial purchase to the customer's final purchase, was introduced to the RMFL model. [23] also used the k-means algorithm to cluster customers and computed the distance from the cluster's center (R, F, M, L) to the coordinate origin (0, 0, 0, 0) to represent the loyalty of each type of customer. Experimental results show that the model has a strong ability to evaluate customer loyalty.

2.2. Recommender system

In the field of e-commerce, recommender systems are defined as implementing traditional marketing strategies via the Internet [24]. As a very important part of the e-commerce website, the recommender system can predict the potential consumption behavior of customers by learning and analyzing their previous behavior. Since the development of the recommender system in the 1990s, many different methods have been proposed, which can be roughly classified into content-based filtering, collaborative filtering, and hybrid filtering [25].

1. Content-based filtering recommender systems

According to the items the customer has selected in the past, the content-based filtering recommender systems show customers products that they may be interested in. It does not require the user ratings or opinions on items but just calculates the similarity between the products to be recommended and the user has purchased. The content-based filtering recommender systems have long been used in recommendations in media areas such as news, websites, and television [26-29].

2. Collaborative filtering recommender systems

Peer-reviewed version available at *Information* **2020**, *11*, 171; doi:10.3390/info11030171

In the field of recommender systems, collaborative filtering is a mature algorithm, first proposed by [30]. There are two main types of collaborative filtering: one is based on customers and another is based on products. The basic idea of the one based on customers is that customers who have approximate scores may have similar interests, so they are likely to choose identical products. The collaborative filtering based on products is similar in principle. After calculating the similarity of various products, the system can recommend similar products to those who have purchased a certain product. The collaborative filtering recommender systems do not rely heavily on customers' purchase records. As long as the customer has purchased the product, or commented the point he is interested in, the system can recommend related products and content to him [31-34].

Since the use of collaborative filtering algorithms, some shortcomings have been exposed, such as cold start and lack of scoring data. [35] proposed a method based on clustering, which can solve the problem of recommending products for minority users in the absence of scoring data. A decision tree method was proposed in [36], which can deal with the problem of missing data in cold start, to implement the function of recommending products to new customers.

3. Hybrid recommender systems

A hybrid recommender system is a way to combine different technologies for the recommendation. In practical applications, considering the data sources and characteristics, hybrid recommender systems that meet specific needs will be constructed. WebBot, a hybrid recommender system containing content filtering and collaborative filtering, was proposed in [37]. Content-based filtering is used to record the relationship between the web page content and users' preferences, and collaborative filtering is used to compare browsing records among different users. WebBot's list will change as the page that the user is currently viewing, so its recommendations are flexible. In [38], a hybrid recommender system based on enhanced fuzzy multi-criteria collaborative filtering is applied to movie recommendation. The method considers demographic information and item-based ontology semantic filtering methods to improve the accuracy of recommendation. [39] use sequential pattern analysis to mine the customers' purchase rules over time, and apply these rules to the recommender system. Therefore, the collaborative can still make recommendations without the customers' explicit evaluation information and the quality of recommendations is improved concurrently.

4. Complex network recommender systems

In recent years, complex networks have attracted interest from researchers in computer science, physics, mathematics, and management science. As a modeling method, a complex network can abstract the multiple entities and relationships into a network structure and describe them by nodes and edges of the network. Since the data sets in the field of recommender systems usually present a "customer-products" structure, the recommendation technology and complex networks have something in common. In [40], the bipartite graph model is used to describe the relationship between customers and products, which improves the recommendation quality of the spreading activation algorithm to a certain extent. A Complex network is used to implement digital tourism assistance, which can recommend optimal routes and provide mobile guidance to the users in [41]. [42] use the standard complex network technology in the sample of social musician network and developed a derived complex network by confirming attribute similarity. The content-based recommendation algorithm is then used to calculate the similarity between two related music network databases to complete the recommendation.

3. Proposed method

In this section, a method for calculating customer loyalty and two recommending algorithms based on complex network are introduced.

3.1. K-means clustering method based on the RFML model: designed for customer loyalty

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 February 2020 doi:10.20944/preprints202002.0367.v1

Peer-reviewed version available at *Information* **2020**, *11*, 171; doi:10.3390/info11030171

3.1.1. The RFML model

The RFML model measures customer loyalty through indicators, namely R: recent purchase, F: purchase frequency, M: total purchase amount, and L: lifetime. The specific calculation methods of the four indicators are as follows: R is calculated by counting the number of days between the time when the customer's last purchase occurred and a specific date (used to separate the training set and the test set); F is obtained by counting the number of customer purchase records; M can be obtained by accumulating the customer's consumption amount; L is obtained by calculating the number of days between the customer's earliest purchase and the last purchase. When R is smaller, F is larger, M is larger, L is larger, and the customer is more loyal.

The four indicators have the same effect on customer loyalty, according to [43]. Using the same weighting method in this study, the RFML model is constructed as follows:

$$\mathrm{RFML} = \mathrm{R} + \mathrm{F} + \mathrm{M} + \mathrm{L} \tag{1}$$

To eliminate the influence of dimension on the final result, this study standardized the indicators. For positive indicators (F, M, L), formula (2) is used for standardization. The negative indicator, R, is standardized by formula (3).

$$x_{ij} = \frac{x_j - x_{min}}{x_{max} - x_{min}} \tag{2}$$

$$x_{ij} = \frac{x_{max} - x_j}{x_{max} - x_{min}} \tag{3}$$

x_j is the j_{th} value of x. x_{min} is the minimum value of x. x_{max} is the maximum value of x. x_{ij} is the standardized value of x_j, and all x_{ij} is between 0 and 1.

3.1.2. K-means clustering based on the RFML model

Clustering is the process of dividing similar data into clusters. A cluster is a collection of data that is similar to each other within the cluster and different from those in other clusters [44]. K-means is a classical method to solve the clustering problem. The basic idea of this method is: first, k data from the set containing n data are selected randomly as the initial cluster center. Then the distance from the remaining $(n - k)$ data to the previously centers (usually Euclidean distance) is calculated. Each remaining data will be assigned to the cluster to which the nearest cluster center belongs, according to the nearest neighbor principle. The mean of all data in each cluster is calculated as the new cluster center generated. If the difference between the new cluster center and that obtained in the previous iteration is less than the threshold, the algorithm converges and the iteration stops. Otherwise, the iterative process continues.

The K-means method has the advantages of fastness and high scalability, but its clustering result is greatly affected by the number of initial class centers k. Given this problem, this study uses the Davies-Bouldin Index (DB Index) [45] to measure the clustering effect under different k, and selects a better clustering result. DB Index is an internal scheme that evaluates the effect of clustering based on the scale and characteristics of the data set. The basic idea is to measure the effect of clustering by calculating the ratio of within-cluster similarity and among-cluster similarity. Because the idea of the clustering method is to make the within-cluster similarity as large as possible and otherwise the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 February 2020 doi:10.20944/preprints202002.0367.v1

Peer-reviewed version available at Information 2020, 11, 171; doi:10.3390/info11030171

among-cluster similarity, it's obvious that the smaller the DB Index, the better the clustering result. The specific calculation method is as follows:

1. S_i is defined to calculate the degree of dispersion of data points in the i_{th} class, as in formula (4) :

$$S_i = \{\frac{1}{T_i}\sum_{j=1}^{T_i} |X_j - A_j|^q\}^{\frac{1}{q}} \tag{4}$$

T_i is the number of data points in i_{th} cluster. X_j is the j_{th} data point in i_{th} cluster, and A_j is the center of i_{th} cluster. When q is 1, S_i calculates the average absolute distance from each data point in the i_{th} cluster to the center of cluster i . When q is 2, S_i calculates the average geometric distance from each data point in the i_{th} cluster to the center of cluster j. Both can be used to measure the degree of dispersion of data points in the cluster.

2. M_{ij} is defined to calculate the distance between the i_{th} cluster and the j_{th} cluster, as shown in formula (5).

$$M_{ij} = \{\sum_{\lambda=1}^{m} |a_{\lambda j} - a_{\lambda j}|^q\}^{\frac{1}{q}} \tag{5}$$

$a_{\lambda i}$ is the λ_{th} attribute value of the i_{th} cluster. $a_{\lambda j}$ is the λ_{th} attribute value of the j_{th} cluster. Similarly, when q = 1, M_{ij} calculates the absolute distance between the two points. When q = 2, M_{ij} calculates the geometric distance between the two points.

3. R_{ij} is defined to calculate the similarity between the i_{th} cluster and the j_{th} cluster.

$$R_{ij} = \frac{S_i + S_j}{M_{ij}} \tag{6}$$

4. Calculate the similarity R_{ij} between cluster I and all the other clusters, select the largest $R_i = \max(R_{ij})$ and record.

5. All R_i are calculated and DB Index is the mean of R_i.

$$\mathrm{DBI} = \frac{1}{N}\sum_{i=1}^{N} R_i \tag{7}$$

3.2. *Recommender system based on bipartite graph and PersonalRank (BGPR)*

A recommender system based on a complex network utilizes nodes to describe customers and products and edges to describe purchase relationships of customers and products. There are two main network structures to describe a recommender system: the "customer-product" bipartite graph and the "product-product" single vertex set. Since the purpose of this study is to recommend shops to customers, in this section, a "customer-shop" bipartite graph is constructed, and the PersonalRank algorithm is then used to recommend shops.

3.2.1 "Customer-shop" bipartite graph network

A bipartite graph is a special model in graph theory. Its nodes can be divided into two

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 February 2020 doi:10.20944/preprints202002.0367.v1

Peer-reviewed version available at Information 2020, 11, 171; doi:10.3390/info11030171

independent and disjoint sets U and V so that the two nodes connected by each edge belong to these two independent sets. The bipartite graph G is usually expressed mathematically as $G = (U, V, E)$, where U and V represent two independent sets of nodes, and E represents a set of edges [46]. A bipartite graph is shown below.

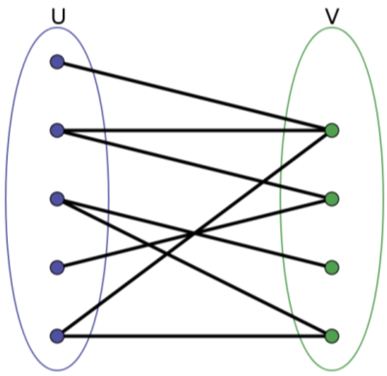

Figure 1. An example of bipartite graph

It should be noted that the degree of any node i in the bipartite graph is the number of all edges connected to the node, or the number of all neighbor nodes of i.

In the data of this study, each purchase record of a customer can be represented by a number pair (u, m) to indicate that user u has purchased at shop m. The data of this structure can be easily represented by a bipartite graph. Let $G = (V_U, V_M, E)$ denote a "customer-shop" bipartite graph, where V_U represents the set of nodes for customers, V_M represents the set of nodes for shops, and E is the set of edges connecting the customers and shops. Each number pair (u, m) has an edge $e(v_u, v_m) \in E$ in the graph G. $v_u \in V_U$, and $v_m \in V_M$. There are no edges between nodes in the same set, according to the nature of the bipartite graph. The following figure is an example of "customer-shop" bipartite graph.

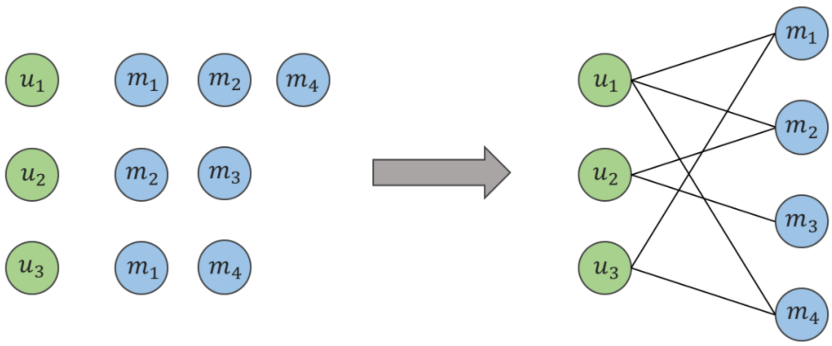

Figure 2. This is an example of a "customer-shop" bipartite graph. The green nodes represent customers, the blue ones are shops, and the edges between the two nodes are the purchasing relationship between customers and shops. For instance, u_1 is connected to m_1, m_2, and m_4, which means that customer 1 has purchased products in shops 1, 2, and 4.

3.2.2 PersonalRank for recommender system

When a bipartite graph structure is used for recommendation, the task of recommending a shop to customer u can be transformed into search of a shop node that is most likely to be connected with customer node v_u. One solution is to measure the correlation between customer v_u and the shops

Peer-reviewed version available at Information 2020, 11, 171; doi:10.3390/info11030171

that have no edges connection with v_u, and generate a recommendation list for customer u according to the level of correlation. The graph ranking algorithm can be used to calculate the node correlation problem. PageRank is a classic algorithm for solving graph ranking problems, and the PersonalRank algorithm is the application of the PageRank algorithm to bipartite graphs [47]. PageRank is an algorithm designed by [48] based on citation analysis to solve the problem of graph ranking and is later applied in the search engine, Google.

When recommending an online shop to customer u, the PersonalRank is similar to PageRank. It is assumed that the customer starts to walk from node v_u and continuously to other nodes through the edges. The customer will choose to continue to walk with probability α in each node or return to the last node with probability $(1-\alpha)$. This is a random process. After multiple rounds of iterations, the probability of each node being accessed will tend to converge. A node with a higher probability means that the customer u is more likely to purchase in this shop. The formula of PersonalRank is as follows:

$$\mathrm{PR}(\mathrm{v}) = (1-\alpha)\gamma + \alpha \sum_{v' \in in(v)} \frac{PR(v')}{|out(v')|}$$
$$\gamma = \begin{cases} 1, v = v_u \\ 0, v \neq v_u \end{cases} \tag{8}$$

$\mathrm{PR}(\mathrm{v})$ is the probability that node v is visited. α is the probability that customer keep walking. $\mathrm{in}(\mathrm{v})$ is the nodes set connected with node v. $\mathrm{out}(\mathrm{v}')$ is the nodes set connected with node v'. v_u is the original node.

3.3. *Recommender system based on single vertex set and DeepWalk (SVDW)*

3.3.1. "Shop-shop" single vertex set network

The recommender systems based on the bipartite graph study the relationship between two node sets. Another approach is to study the relationships within the single vertex set. According to the requirements of the recommender system, the bipartite graph model needs to be mapped into a single vertex set network. One mapping method is: if two nodes in the same set are connected to at least one node in another set by an edge, then there is an edge in the single vertex set connecting the two nodes [49]. In the "customer-shop" bipartite graph $\mathrm{G} = (V_U, V_M, \mathrm{E})$, there is an edge connection between two shop nodes if they have common neighbor nodes in the set of customers. The weight of the edge is the number of common neighbor nodes. After the above steps, the "customer-shop" bipartite graph can be mapped to a directionless weighted network $\mathrm{G}' = (V_M, \mathrm{E}')$ of shops, where V_M is the shop node set and E' is the edge set.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 February 2020 doi:10.20944/preprints202002.0367.v1

Peer-reviewed version available at *Information* **2020**, *11*, 171; doi:10.3390/info11030171

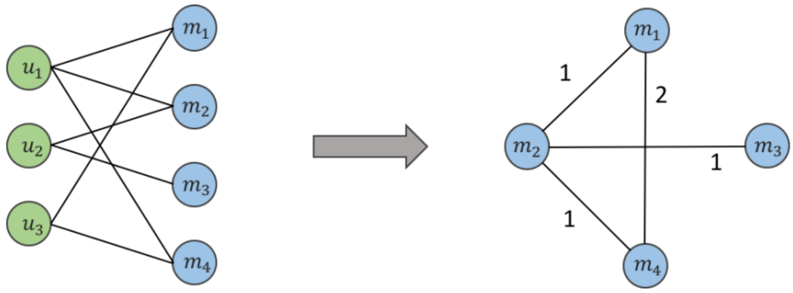

Figure 3. An example of "shop-shop" single vertex set network

3.3.2. DeepWalk for recommender system

DeepWalk is a method to learn the implicit representation of nodes in a complex network. Implicit representation refers to projecting the relationships between nodes in a continuous vector space to analyze the network. Deepwalk is a combination of RandomWalk and Word2Vec. RandomWalk has been used for similarity measurement in many machine learning problems such as content recommendation [50] and community discovery [51]. Word2vec can map words to a unified coordinate system based on text content to get a vector representation of each word. DeepWalk first utilizes weighted RandomWalk to randomly walk through the graph to collect node access sequences, then treats the node access sequences as a sentence and inputs them into the Word2Vec model, and vector representations of each node are obtained. Finally, the similarity between the nodes can be measured by calculating the distance between the corresponding vectors of the nodes to recommend shops for customers.

1. RandomWalk

For node $v_m \in V_M$, W_M is defined as a sequence that starts with v_m. W_M is a random process composed of variables $W_m^1, W_m^2, \dots, W_m^k, \dots, W_m^t$. W_m^{k+1} is equal to v_m with probability α. A node is selected from the neighbor nodes of W_m^k, and the probability that W_m^{k+1} is equal to this node is $(1-$ α). The probability of selecting nodes from the neighbor nodes of W_m^{k+1} is as follows:

$$\mathrm{prob}\left(W_m^{k+1} = v_j \middle| W_m^k = v_i\right) = \begin{cases} \dfrac{weight(v_j)}{\sum_{d=1}^{p} weight(v_d)}, & if\ (v_i, v_j) \in E' \\ 0, & otherwise \end{cases} \tag{9}$$

v_j is one of the neighbor nodes of W_m^k. $\mathrm{weight}(v_j)$ is the weight of the edge connecting W_m^k and v_j. p is the number of neighbor nodes of W_m^k. t is the length of each node sequence.

2. Word2Vec

Word2Vec is a widely used word embedding model with the advantage of fast training speed. Word2Vec uses cross-entropy as a training target to replace the traditional maximum likelihood function, and at the same time introduces negative sampling and hierarchical softmax normalization function creatively, so that the model can be trained in a few hours. The Word2Vec model has two training methods, Continuous Bag-Of-Word (CBOW) and Skip-Gram. The difference is that the CBOW method uses context words to predict the central word, while Skip-Gram does the opposite. It uses the central word to predict the context words [52]. The purpose of this study is to

 doi:10.20944/preprints202002.0367.v1

Peer-reviewed version available at Information 2020, 11, 171; doi:10.3390/info11030171

recommend similar shops to customers based on their historical purchase records, which is more in line with Skip-Gram's training hypothesis, so we will focus on the Skip-Gram model.

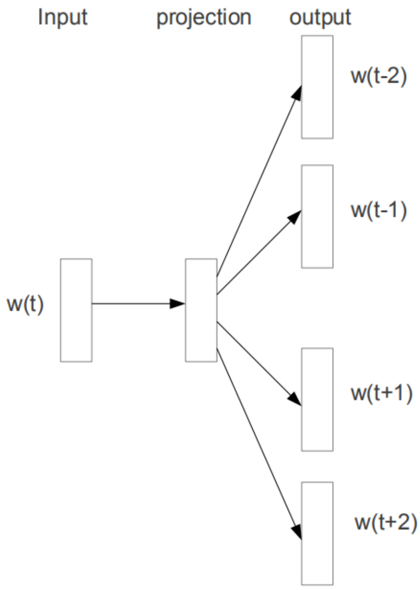

Figure 4. Model structure of Skip-Gram

As shown in Figure 4, Skip-Gram has a three-layer structure: an input layer, a projection layer, and an output layer. Its forward calculation process is:

$$\mathrm{p}(w_o|w_i) = \frac{e^{U_o V_i}}{\sum_j e^{U_j V_i}} \tag{10}$$

V_i is the column vector of the matrix in input layer and is also the input vector of w(t). U_j is the row vector of the matrix in the output layer. The essence of Skip-Gram is to calculate the cosine similarity between the input vector and the output vector of a target word and to perform softmax normalization. Word2Vec utilizes a hierarchical softmax method. The basic idea is to decompose the complex normalized probability into a product of multiple conditional probability:

$$\mathrm{p}(v|context) = \prod_{i=1}^{m} p(b_i(v)|b_1(v), \dots, b_{i-1}(v), context) \tag{11}$$

Hierarchical softmax reduces the computational complexity of probability by constructing a binary tree. In practical applications, binary trees based on Huffman coding can meet the needs of most application scenarios.

3.4. *Evaluation index*

Considering the balance between the accuracy and scale of recommendation, F1-Score is used in this study to evaluate the recommendation results.

1. Accuracy

The percentage of shops that customer actually purchased to all the shops recommended by the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 February 2020 doi:10.20944/preprints202002.0367.v1

Peer-reviewed version available at Information 2020, 11, 171; doi:10.3390/info11030171

system to the customer in the test set.

$$\text{accuracy} = \frac{|R \cap T|}{|R|} \tag{12}$$

R is the set of shops recommended to the target customer set. T is the set of new shops that the target customer set visit in the test data.

2. Recall

The proportion of system-recommended shops in new shops visited by customers.

$$\text{recall} = \frac{|R \cap T|}{|T|} \tag{13}$$

3. F1-Score

F1-Score is the harmonized average of accuracy and recall.

$$\text{F1} = \frac{2 * accuracy * recall}{accuracy + recall} \tag{14}$$

4. Experiment and analysis

4.1. Experimental environment

1. Hardware

CPU: 2.5 GHz Intel Core i7

Memory: 16G

2. Software

Operating system: mac OS (High Sierra)

Database: MySQL 8.0

Programming language: Python 2.7/ R 3.3.0

4.2. Data set

Group B is the largest real estate group in Malaysia. It operates more than 150 subsidiaries including tourism, resorts, hotels, shopping malls and so on. Group B launched its customer loyalty program in 2010, aiming to create a platform using general points as a medium, bringing together shops in different fields to provide customers with rich points redemption options, thereby enhancing customer loyalty.

Table 1. The distribution of data and the division of training and test sets

	Training set	Test set
Time span	2010.07.26 - 2015.06.20	2015.06.21 - 2016.09.20
Number of customers	21,187	18,986
Number of online shops	2,418	1,753
Number of purchasing records	694,176	459,962
Proportion of data	3/5	2/5

4.3. K-means clustering based on the RFML model

The RFML loyalty score of each customer is first calculated, and each customer is regarded as a point in a four-dimensional space. The four dimensions are the four indicators of the RFML. K-means method is used to cluster the customer set. The initial k value range is 5 to 50, multiple clustering is performed, and the DB Index is recorded during each clustering process. The k value corresponding to the minimum DB Index is used as the optimal clustering result.

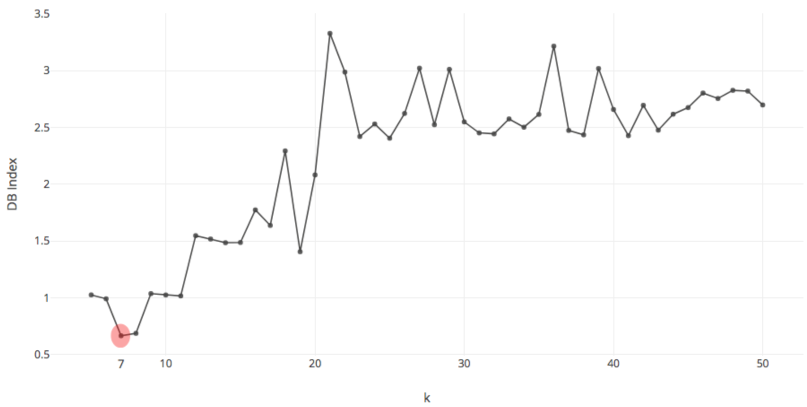

Figure 5. DB Index at different k values. The horizontal axis represents the k value, and the vertical axis represents the DB Index. When k = 7, the DB Index value is minimum.

The RFML value of each cluster center and the value of each indicator are calculated separately. The results are as follows:

Table 2. Results of clustering

Cluster number	**Cluster scale**	**Cluster center**				**RFML**
		R	**F**	**M**	**L**	**Value**
7	1	0.99776	0.00470	1.00000	0.72909	1.61819

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 February 2020 doi:10.20944/preprints202002.0367.v1

Peer-reviewed version available at Information 2020, 11, 171; doi:10.3390/info11030171

6	3	0.97668	0.71829	0.04344	0.41103	1.35833
5	3749	0.95915	0.00241	0.00066	0.65454	1.01921
4	7133	0.92724	0.00235	0.00035	0.38307	1.00368
3	8875	0.95269	0.00029	0.00005	0.04897	0.68357
2	5645	0.62550	0.00036	0.00005	0.1053	0.48267
1	3992	0.32104	0.00014	0.00004	0.03593	0.22910

According to section 3.1.1, a higher RFML value means higher customer loyalty. Further, the data points are shown in the R-L and F-M cross coordinate systems, respectively, as shown in Figure 6.

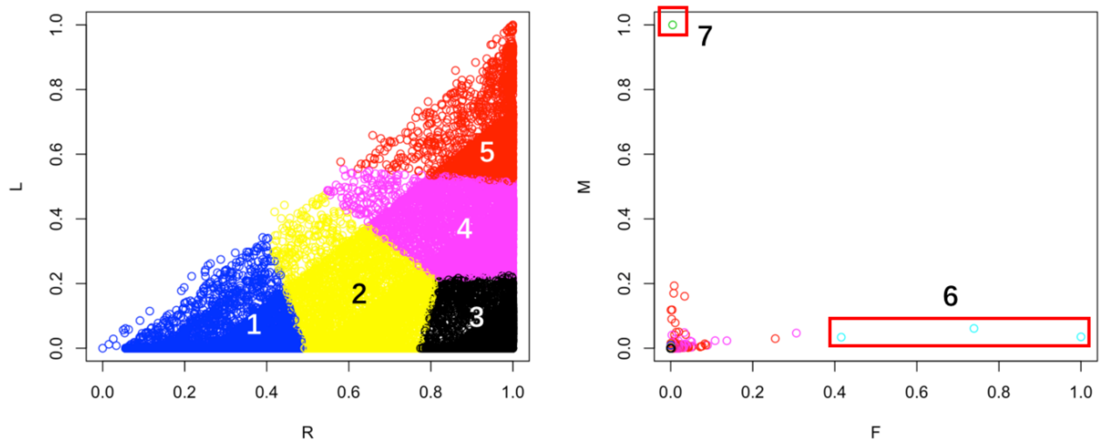

Figure 6. Visualization of clustering results

According to the left part of figure 6, because there are few customers in the 6th and 7th clusters, the data points are covered by the other points and not visible. The rest customers are evenly divided into five clusters in the R-L dimension. The R-L cross-distribution figure shows obvious boundaries because the R and L have a certain correlation. Customers in the red cluster have high R values (recent purchases) and L values (lifetimes), showing that they are active recently and have long lifetimes. In the right part of figure 6, the 6th and 7th clusters are outliers with high R and L values. To simplify the analysis, these 4 customers in the two clusters are classified into the 5th cluster.

4.4. Impact of customer loyalty on BGPR

The purpose of this section is to explore whether customer loyalty affects the accuracy of a bipartite graph-based recommender system. The target customers are first selected from the training set and the test set. The method is: from 21,187 customers in the training set and 18,986 customers in the test set, we select customers who appear in both sets at and have purchased in new shops in the test set. 8,318 target customers are obtained finally.

Peer-reviewed version available at *Information* **2020**, *11*, 171; doi:10.3390/info11030171

The "customer-shop" bipartite graph of the target customers is constructed from the training set data. Then PersonalRank algorithm is used to make recommendations for target customers. F1-Score of the test set is calculated during this process. The pseudocode of the experimental process is as follows:

Algorithm 1: BGPR

- Construction of "customer-shop" bipartite graph G;
- Initialization of parameter N
- For each target customer u:

    Calculate the similarity RD among the target customer node v_u and all other nodes according to formula (8);

    Filter out the shop nodes that are not connected to the customer node v_u in the bipartite graph and add them to the recommended candidate list CR;

    Arrange the shop nodes in the candidate list CR in the order of decreasing similarity, and select the first N shops;

- Calculate the F1-Score of the recommendation result according to the formula (14);

The iterative approach is used to estimate N (the number of shops recommended to each customer) in the experiment. The results of the iteration are shown in Figure 7.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 February 2020 doi:10.20944/preprints202002.0367.v1

Peer-reviewed version available at *Information* **2020**, *11*, 171; doi:10.3390/info11030171

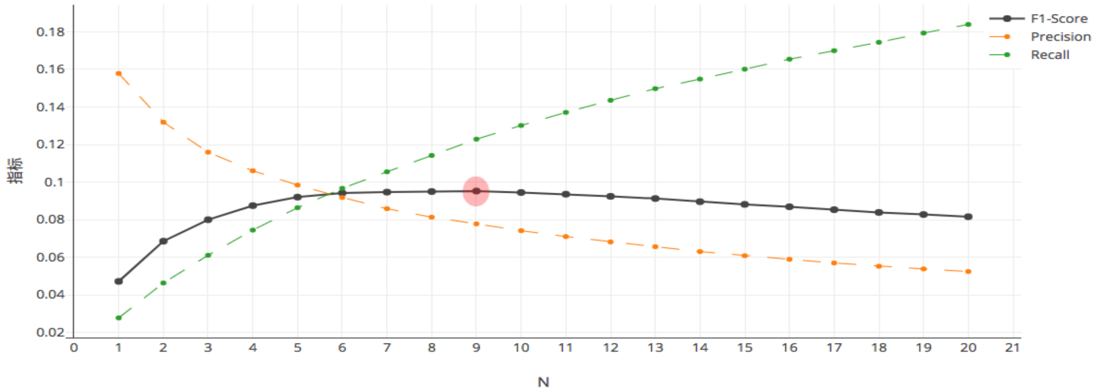

Figure 7. Iteration results of BGPR algorithm.

As N increases, precision(accuracy) gradually decreases, recall gradually increases, and F1-Score increases first and then decreases. When N is 9, F1-Score is 0.095, reaching the peak.

Considering customer loyalty, 8,318 target customers are grouped according to different customer loyalty levels, and each group of customers is recommended. According to the above experimental results, let N equal 9. The performance of the PersonalRank in different customer groups is shown in Table 3.

Table 3. Recommendation effect of PersonalRank under different customer loyalty levels

Customer loyalty levels	Number of customer	Precision	Recall	F1-Score
1	0	0.000	0.000	0.000
2	24	0.032	0.233	0.057
3	360	0.056	0.160	0.083
4	4769	0.086	0.148	0.109
5	3165	0.067	0.091	0.077

From Table 3, PersonalRank performs differently on customer groups with different loyalty levels. The recommendation effect is best when the loyalty level is 4. Compared with the experimental results without customer loyalty (see Table 4), the three evaluation indexes have improved, and the F1-Score improvement rate is 14.74%. Therefore, rationally adding the customer loyalty factor can improve the accuracy of the recommender system based on the bipartite graph.

Table 4. Recommendation effect of PersonalRank with and without customer loyalty

	Precision	Recall	F1-Score
Without customer loyalty	0.078	0.123	0.095
With customer loyalty	0.086	0.148	0.109

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 February 2020 doi:10.20944/preprints202002.0367.v1

Peer-reviewed version available at *Information* **2020**, *11*, 171; doi:10.3390/info11030171

4.5. Impact of customer loyalty on SVDW

According to the bipartite graph G in 4.4, a single vertex set network G' of shops is constructed. The number of nodes in G' is 1,753, which means that there are 1,753 shops. The average number of customers the per shop is 60.79. The visualization of the single vertex set network of shops is shown in figure 8, where the larger the number of customers, the larger the shape of shop node.

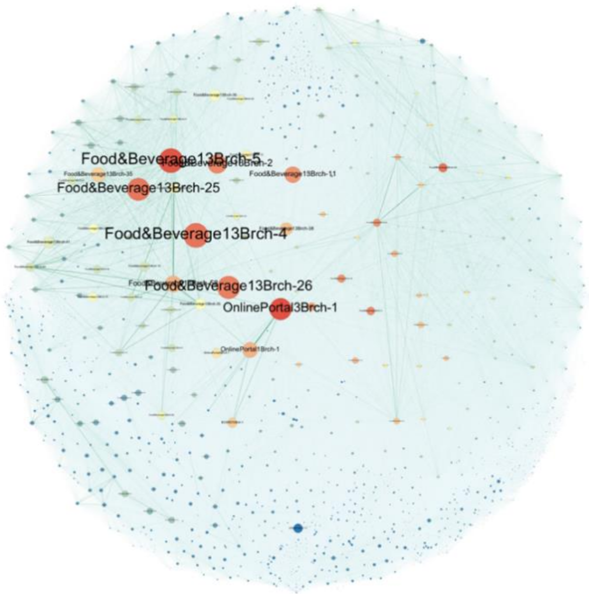

Figure 8. The visualization of the single vertex set network of shops

Based on the single vertex set network of shops, DeepWalk is used to calculate the similarity between the shops that the target customer has historically purchased and not. Shops, where the target customer has purchased in the training data set, are excluded. The remaining shops are sorted in the order of decreasing similarity, and the first N shops are selected to recommend to the target customers. After that, the F1-Score of the recommendation result is calculated using the test data set. The pseudocode of the experimental process is as follows:

Peer-reviewed version available at *Information* **2020**, *11*, 171; doi:10.3390/info11030171

Algorithm 2: SVDW

- Construction of "shop-shop" single vertex set network G;
- Initialization of parameter N;
- For each shop $v_m \in V_M$:

  For θ (Maximum number of walks per shop node):

  $W_m = RandomWalk(v_m, t)$;

  Model = Word2Vec(W_m);
- For each target customer u:

  Calculate the average vector $\overline{Vec_u}$ of all the shops purchased by customer u;

  Calculate the cosine similarity RD of $\overline{Vec_u}$ to all other shop nodes;

  Filter out the shop nodes that customer u has not purchased as the recommendation candidate list CR;

  Arrange the shop nodes in the candidate list CR in the descending order of similarity, and select the first N shops;
- Calculate the F1-Score of the recommendation result according to the formula (14);

In the experiment, the parameters are set as: θ = 70, t = 50, α = 0.3. The experiment utilizes an iterative approach to estimate N (the number of shops recommended to each customer), and the results of the iteration are shown below.

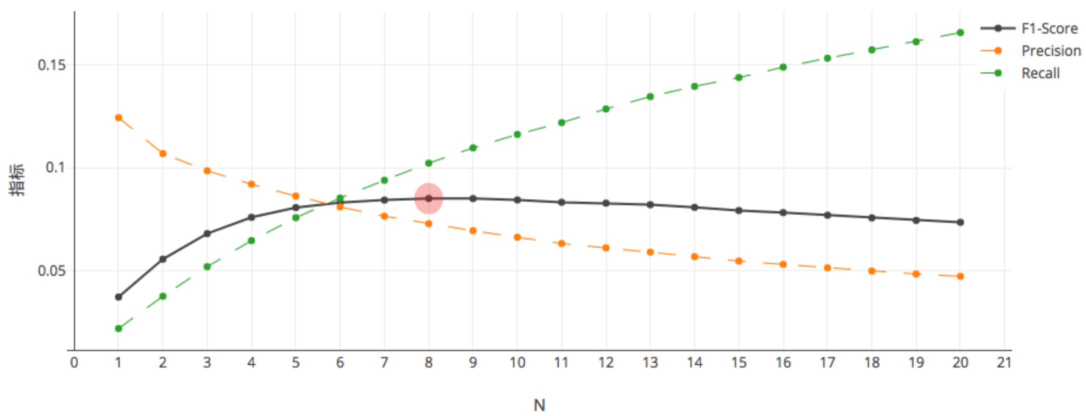

Figure 9. Iteration results of DeepWalk algorithm.

As N increases, precision(accuracy) gradually decreases, recall gradually increases, and F1-Score increases first and then decreases. When N is 8, F1-Score is 0.085, reaching the peak.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 February 2020 doi:10.20944/preprints202002.0367.v1

Peer-reviewed version available at Information 2020, 11, 171; doi:10.3390/info11030171

Considering customer loyalty, a method similar to section 4.4 was adopted. The results of DeepWalk recommendation in different customer loyalty groups are shown in the table below：

Table 5. Recommendation effect of DeepWalk under different customer loyalty levels

Customer loyalty levels	Number of customer	Precision	Recall	F1-Score
1	0	0.000	0.000	0.000
2	24	0.042	0.267	0.072
3	360	0.054	0.138	0.078
4	4769	0.082	0.126	0.099
5	3165	0.061	0.073	0.066

From Table 5, DeepWalk performs differently on customer groups with different loyalty levels. The recommendation effect is best when the loyalty level is 4. Compared with the experimental results without customer loyalty (see Table 6), the three evaluation indexes have improved, and the F1-Score improvement rate is 16.47%. Therefore, rationally adding customer loyalty factors can improve the accuracy of the recommender system based on the single vertex set network.

Table 6. Recommendation effect of DeepWalk with and without customer loyalty

	Precision	Recall	F1-Score
Without customer loyalty	0.072	0.102	0.085
With customer loyalty	0.082	0.126	0.099

This section may be divided by subheadings. It should provide a concise and precise description of the experimental results, their interpretation as well as the experimental conclusions that can be drawn.

5. Conclusion

With the development of complex networks and recommender systems, the intersection of the two fields, the recommender system based on a complex network, has gradually become a popular research direction. The main idea is to establish a network to describe the relationship between customers and products (or online shops). The mutual recommendation ability based on similarity is calculated, and related products (or online shops) are recommended to target customers.

In the field of marketing, customer loyalty is an important indicator to measure the strength of

Peer-reviewed version available at *Information* **2020**, *11*, 171; doi:10.3390/info11030171

the preference relationship between customers and products. Therefore, introducing customer loyalty into recommender system is a meaningful research direction. In this study, we consider the impact of customer loyalty on the accuracy of recommendation algorithms based on complex networks.

This study uses customer purchase data of B Group in Malaysian. The RFML model is introduced to extract four features of customer loyalty. These four features are then passed into the k-means model to cluster customers, and 5 customer clusters with different loyalty levels are obtained. Two recommendation algorithms: BGPR and SVDW are used for experimental verification. The experimental results prove that: (1) The two recommendation algorithms have different effects on customer clusters with different loyalty levels, and both perform best on level 4. (2) Customer loyalty can optimize the recommendation algorithm based on complex networks. Moreover, we are inspired that if recommending products or services to those customers with higher loyalty, it may bring more benefits when the e-commerce platform uses recommender systems.

References

1. Oliver, R.L., *Whence consumer loyalty?* Journal of marketing, 1999. **63**(4_suppl1): p. 33-44.
2. Butcher, H. *Information overload in management and business*. in *IEE Colloquium on information overload*. 1995. IET.
3. Cano, P., et al., *Topology of music recommendation networks.* Chaos: An interdisciplinary journal of nonlinear science, 2006. **16**(1): p. 013107.
4. Naruchitparames, J., M.H. Güneş, and S.J. Louis. *Friend recommendations in social networks using genetic algorithms and network topology*. in *2011 IEEE Congress of Evolutionary Computation (CEC)*. 2011. IEEE.
5. Li, J., S. Ma, and S. Hong. *Recommendation on social network based on graph model*. in *proceedings of the 31st Chinese Control Conference*. 2012. IEEE.
6. Wang, Q., et al. *Academic paper recommendation based on community detection in citation-collaboration networks*. in *Asia-Pacific web conference*. 2016. Springer.
7. Reichheld, F.F., R. Markey, and C. Hopton, *The loyalty effect.* EUROPEAN BUSINESS JOURNAL, 2000. **12**(3): p. 134-139.
8. Han, S.H., S.X. Lu, and S.C. Leung, *Segmentation of telecom customers based on customer value by decision tree model.* Expert Systems with Applications, 2012. **39**(4): p. 3964-3973.
9. Kressmann, F., et al., *Direct and indirect effects of self-image congruence on brand loyalty.* Journal of Business research, 2006. **59**(9): p. 955-964.
10. Hwang, H., T. Jung, and E. Suh, *An LTV model and customer segmentation based on customer value: a case study on the wireless telecommunication industry.* Expert systems with applications, 2004. **26**(2): p. 181-188.

Peer-reviewed version available at *Information* **2020**, *11*, 171; doi:10.3390/info11030171

11. Jackson, B.B., *Build customer relationships that last*. Vol. 11. 1985: Harvard Business Review.

12. Qi, J.-Y., et al., *Are customer satisfaction and customer loyalty drivers of customer lifetime value in mobile data services: a comparative cross-country study*. Information Technology and Management, 2012. **13**(4): p. 281-296.

13. Zhang, J.Q., A. Dixit, and R. Friedmann, *Customer loyalty and lifetime value: an empirical investigation of consumer packaged goods.* Journal of marketing theory and practice, 2010. **18**(2): p. 127-140.

14. SAFARI, K.M. and K.Z. SAFARI, *An empirical analysis to design enhanced customer lifetime value based on customer loyalty: Evidences from Iranian banking sector.* 2012.

15. Andon, P., J. Baxter, and G. Bradley, *Calculating customer lifetime value (CLV): theory and practice*, in *Kundenwert*. 2003, Springer. p. 299-315.

16. Fader, P.S., B.G. Hardie, and K. Jerath, *Estimating CLV using aggregated data: the Tuscan lifestyles case revisited.* Journal of Interactive Marketing, 2007. **21**(3): p. 55-71.

17. Karvanen, J., A. Rantanen, and L. Luoma, *Survey data and Bayesian analysis: a cost-efficient way to estimate customer equity*. Quantitative Marketing and Economics, 2014. **12**(3): p. 305-329.

18. Yeh, I.-C., K.-J. Yang, and T.-M. Ting, *Knowledge discovery on RFM model using Bernoulli sequence.* Expert Systems with Applications, 2009. **36**(3): p. 5866-5871.

19. Miglautsch, J.R., *Thoughts on RFM scoring.* Journal of Database Marketing & Customer Strategy Management, 2000. **8**(1): p. 67-72.

20. Khajvand, M., et al., *Estimating customer lifetime value based on RFM analysis of customer purchase behavior: Case study*. Procedia Computer Science, 2011. **3**: p. 57-63.

21. Safari, F., N. Safari, and G.A. Montazer, *Customer lifetime value determination based on RFM model.* Marketing Intelligence & Planning, 2016.

22. Ha, S.H. and S.C. Park, *Application of data mining tools to hotel data mart on the Intranet for database marketing*. Expert Systems with Applications, 1998. **15**(1): p. 1-31.

23. Hosseini, S.M.S., A. Maleki, and M.R. Gholamian, *Cluster analysis using data mining approach to develop CRM methodology to assess the customer loyalty.* Expert Systems with Applications, 2010. **37**(7): p. 5259-5264.

24. Schafer, J.B., J.A. Konstan, and J. Riedl, *E-commerce recommendation applications.* Data mining and knowledge discovery, 2001. **5**(1-2): p. 115-153.

25. Riyahi, M. and M.K. Sohrabi, *Providing effective recommendations in discussion groups using a new hybrid recommender system based on implicit ratings and semantic similarity*. Electronic Commerce Research and Applications, 2020: p. 100938.

26. Yeung, K.F. and Y. Yang. *A proactive personalized mobile news recommendation system*. in *2010 Developments in E-systems Engineering*. 2010. IEEE.

Peer-reviewed version available at *Information* **2020**, *11*, 171; doi:10.3390/info11030171

27. Van Meteren, R. and M. Van Someren. *Using content-based filtering for recommendation*. in *Proceedings of the Machine Learning in the New Information Age: MLnet/ECML2000 Workshop*. 2000.

28. Blanco-Fernandez, Y., et al., *Providing entertainment by content-based filtering and semantic reasoning in intelligent recommender systems.* IEEE Transactions on Consumer Electronics, 2008. **54**(2): p. 727-735.

29. Barragáns-Martínez, A.B., et al., *A hybrid content-based and item-based collaborative filtering approach to recommend TV programs enhanced with singular value decomposition.* Information Sciences, 2010. **180**(22): p. 4290-4311.

30. Goldberg, D., et al., *Using collaborative filtering to weave an information tapestry.* Communications of the ACM, 1992. **35**(12): p. 61-70.

31. Linden, G., B. Smith, and J. York, *Amazon. com recommendations: Item-to-item collaborative filtering.* IEEE Internet computing, 2003. **7**(1): p. 76-80.

32. Das, A.S., et al. *Google news personalization: scalable online collaborative filtering*. in *Proceedings of the 16th international conference on World Wide Web*. 2007.

33. Chen, J., et al. *Attentive collaborative filtering: Multimedia recommendation with item-and component-level attention*. in *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*. 2017.

34. Subramaniyaswamy, V., et al., *A personalised movie recommendation system based on collaborative filtering.* International Journal of High Performance Computing and Networking, 2017. **10**(1-2): p. 54-63.

35. Ghazanfar, M.A. and A. Prügel-Bennett, *Leveraging clustering approaches to solve the gray-sheep users problem in recommender systems.* Expert Systems with Applications, 2014. **41**(7): p. 3261-3275.

36. Golbandi, N., Y. Koren, and R. Lempel. *Adaptive bootstrapping of recommender systems using decision trees*. in *Proceedings of the fourth ACM international conference on Web search and data mining*. 2011.

37. Burke, R., *Hybrid recommender systems: Survey and experiments.* User modeling and user-adapted interaction, 2002. **12**(4): p. 331-370.

38. Kermany, N.R. and S.H. Alizadeh, *A hybrid multi-criteria recommender system using ontology and neuro-fuzzy techniques.* Electronic Commerce Research and Applications, 2017. **21**: p. 50-64.

39. Choi, K., et al., *A hybrid online-product recommendation system: Combining implicit rating-based collaborative filtering and sequential pattern analysis.* electronic commerce research and applications, 2012. **11**(4): p. 309-317.

40. Huang, Z., H. Chen, and D. Zeng, *Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering.* ACM Transactions on Information Systems (TOIS), 2004. **22**(1): p. 116-142.

41. Schering, A.-C., et al. *A mobile tourist assistance and recommendation system based on complex networks*. in *Proceedings of the 1st ACM international workshop on Complex networks meet information & knowledge management*. 2009.

42. Fields, B., et al., *Analysis and exploitation of musician social networks for recommendation and discovery.* IEEE Transactions on Multimedia, 2011. **13**(4): p. 674-686.

43. Hughes, A.M., *Strategic database marketing: the masterplan for starting and managing a profitable, customer-based marketing program*. Vol. 12. 2000: McGraw-Hill New York.

44. Han, J., J. Pei, and M. Kamber, *Data mining: concepts and techniques*. 2011: Elsevier.

45. Davies, D.L. and D.W. Bouldin, *A cluster separation measure.* IEEE transactions on pattern analysis and machine intelligence, 1979(2): p. 224-227.

46. Chartrand, G. and P. Zhang, *Chromatic graph theory*. 2008: Chapman and Hall/CRC.

47. Haveliwala, T.H., *Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search.* IEEE transactions on knowledge and data engineering, 2003. **15**(4): p. 784-796.

48. Page, L., et al., *The pagerank citation ranking: Bringing order to the web*. 1999, Stanford InfoLab.

49. Guillaume, J.-L. and M. Latapy, *Bipartite graphs as models of complex networks.* Physica A: Statistical Mechanics and its Applications, 2006. **371**(2): p. 795-813.

50. Fouss, F., et al., *Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation.* IEEE Transactions on knowledge and data engineering, 2007. **19**(3): p. 355-369.

51. Andersen, R., F. Chung, and K. Lang. *Local graph partitioning using pagerank vectors*. in *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*. 2006. IEEE.

52. Mikolov, T., et al. *Distributed representations of words and phrases and their compositionality*. in *Advances in neural information processing systems*. 2013.