Real Estate Marketing Adaptive Decision-Making Algorithm Based on Big Data Analysis

Sheng Lv

School of Wuhan University, Wuhan, Hubei 430072, China

Correspondence should be addressed to Sheng Lv; 2018101050111@whu.edu.cn

Received 11 October 2021; Revised 1 November 2021; Accepted 21 March 2022; Published 12 April 2022

Academic Editor: Jian Su

Copyright © 2022 Sheng Lv. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Aiming at the problems of low stability and efficiency of marketing decision-making and large complexity of marketing decision-making in the current marketing adaptive decision-making algorithm, a real estate marketing adaptive decision-making algorithm based on Big Data analysis is proposed. By analyzing the concept of Big Data, using the Big Data distributed computing architecture, researching the data mining-related algorithms. By constructing an association rule algorithm, mining the rules between real estate marketing and related factors. Based on the Spark-distributed computing platform, an optimization idea of association mining is designed. Decision tree algorithm is used to select discrete and continuous attribute features. According to the characteristics of real estate marketing data and the weight-based discrimination method, the decision tree pruning algorithm is optimized using the classification accuracy, stability, and complexity criteria, and the adaptive decision-making model of real estate marketing is constructed to realize the adaptive decision-making of real estate marketing. The experimental results show that the proposed algorithm has high stability and efficiency in real estate marketing adaptive decision-making and can effectively reduce the complexity of marketing decision-making.

1. Introduction

Marketing is produced in the process of commercial economic activities under the market economy. A series of production, information transmission, economic exchange, value transfer, and other activities will be carried out around the product [1–3]. The real estate industry is an old traditional industry with a huge system and complex industrial chain. At present, the organizational structure and marketing mode of most real estate enterprises still follow the tradition. In the face of huge market competition pressure, the traditional mode of real estate marketing needs to be broken through [4, 5]. Big Data touches the nerves of the real estate industry. By mining Big Data, real estate enterprises can accurately understand consumers’ purchase needs and make accurate marketing decisions [6, 7]. Therefore, the study of real estate marketing decision-making is of great significance for the real estate industry to reduce the waste of resources caused by blind marketing and quickly improve the accuracy and effect of marketing.

At present, scholars in related fields have studied marketing decision-making and achieved some theoretical results. Lessmann et al. [8] proposed an integrated learning framework to support marketing decision-making with customers as the profit goal. According to the principle of profit awareness integration, a framework is established for maximizing formal integration of statistical learning principles and business objectives and the extent to which the profit concise model increases the bottom line is evaluated. By studying the interaction between data-driven learning algorithm and its business value in real-world application environment, we can support marketing decision-making. The algorithm can effectively contribute to the emerging field of profit analysis and provide unique insights on how to implement profit analysis in marketing. Gáti and Bauer [9] put forward the marketing decisions of small- and medium-sized enterprises in Hungary. By defining the marketing activities of small and medium-sized enterprises in the scope different from large enterprises, using qualitative algorithms, this paper analyzes the marketing activities of small and
medium-sized enterprises from an exploratory perspective and studies 15 small- and medium-sized enterprises. In the context of Hungary, analyze the decisive role of SME leaders, strong customer attention, ability to adapt to the market, and characteristics of innovation and entrepreneurship. The algorithm can better explain how small- and medium-sized enterprises make marketing decisions. However, the aforementioned methods still have the problems of low stability, efficiency, and complexity of marketing decision-making. Lessmann et al. [8] proposed an integrated learning framework to support marketing decision-making with customers as the profit goal. If marketing information can contact the right customers, it is the most effective. Deciding which customers to contact is an important task in event planning. This paper mainly studies the empirical goal model and believes that the common practice of developing such a model cannot fully explain the business goal. In order to make up for this, the overall choice of profit consciousness is proposed, which is a model framework that integrates statistical learning principles and business objectives in the form of activity profit maximization. Studying the interaction between data-driven learning methods and their business value in real application environment is helpful to the emerging field of profit analysis and provides original insights on how to implement profit analysis in marketing. It also estimates the extent to which the profit awareness model increases the bottom line. The results of a comprehensive empirical study confirm the commercial value of the proposed integrated learning framework because it recommends more profitable target groups than several benchmarks. Pourmoayed and Relundnielsen [10] proposed the optimization of pig marketing decision under price fluctuation. In the production process of finishing pigs, pig marketing refers to a series of culling decisions before the production unit is emptied. The profit of the production unit largely depends on the pork price, feeding cost, and the cost of purchasing piglets. Therefore, the price fluctuation in the market will affect the profit. Under different price conditions, the optimal marketing decision may change. Most studies have considered pig marketing under the condition of constant price. However, as price fluctuation plays an important role in price and price optimization decision-making, it is necessary to consider the damage under price fluctuation. A hierarchical Markov decision process with two levels is established, which simulates the sequential marketing decision under price fluctuation in pigsty. The status of the system is based on the information of pork, piglets, and feed prices. In addition, Bayesian method is used to update the information and embed it into the hierarchical Markov decision process. The optimal strategies under different price fluctuation modes are analyzed, and the value of incorporating price information into the model is evaluated.

Aiming at the aforementioned problems, an adaptive decision-making algorithm for real estate marketing based on Big Data analysis is proposed. Using Big Data distributed computing architecture, research the data mining-related algorithms. By constructing an association rule algorithm, mining the rules between real estate marketing and related factors. Based on the Spark-distributed computing platform, an optimization idea of association mining is designed. Combine decision tree algorithm to select attribute features. According to the characteristics of real estate marketing data, based on the weight-based discrimination method, the decision tree pruning algorithm is optimized using triple criteria to realize the adaptive decision-making of real estate marketing. The marketing decision-making stability and efficiency of this method are relatively high, and the complexity is relatively small.

2. Big Data Technology

2.1. Concept and Advantages of Big Data. Big Data refers to a data set with a large scale that greatly exceeds the capability of traditional database software tools in terms of acquisition, storage, management, and analysis. It has four characteristics: massive data scale, rapid data flow, diverse data types, and low value density [11]. The main value of Big Data is to reflect the correlation between things, that is, summarize the laws and then predict the future. Big Data mining cannot adopt simple statistical sampling and other methods, and all data must be analyzed. Generally speaking, the more data, the higher value can be mined.

With the continuous development of Internet technology, Internet marketing in the real estate market has become a marketing behavior and trend. Facing the rapidly changing market, real estate enterprises should make rapid response and determine the project marketing decision, so they should timely and accurately understand the market situation and grasp the market dynamics. As the most popular word at present, Big Data has penetrated and radiated to all walks of life, subverting many operation thinking and marketing modes, especially in traditional industries. When using Big Data technology, you can know what data you need most through data mining, obtain more productivity through these data, improve production capacity, and bring more business value to enterprises. Big Data technology seeks the most appropriate enterprise marketing strategy in the analysis of massive data, and brings more wise strategies to enterprises through Big Data analysis. Big Data technology can improve the productivity of the industry, improve marketing decisions, and bring better development prospects to enterprises.

2.2. Big Data Distributed Computing Architecture

2.2.1. Hadoop Distributed Computing Framework. Hadoop is a Big Data-distributed file system. Users only need to directly use the provided transparent interface, which provides a lot of convenience for building distributed applications and improves throughput [12–14]. Hadoop can be managed by Zookeeper, confiG.multiple standby master nodes, and back up metadata. When a special situation such as a downtime occurs, the system will quickly restore the metadata, and the actual data on the child nodes is also greatly guaranteed, which strengthens the fault tolerance of the system platform. The Hadoop distributed platform architecture is as Figure 1.
According to Figure 1, we can see that from the initial HDFS in the underlying data management mode, the resource scheduling management module YARN, and the unified iterative computing model MapReduce, more functional modules have been derived. For example, HIVE can use SQL statement operation to simplify MapReduce programming model; HBase is about NoSQL distributed column storage database in Big Data ecology, and Zookeeper can act as the general service manager in Hadoop cluster, coordinate the working mode of each framework module in Hadoop ecology and manage the tasks of each node.

HDFS is one of the most basic components in Hadoop-distributed computing framework. HDFS is designed to store massive data. A single file stored in HDFS is often in GB or even TB. HDFS divides large files into smaller blocks and stores them on different data nodes. Processing and analyzing files in a centralized way will cause network congestion. Therefore, HDFS adopts the way of making computing close to data, which reduces the data movement overhead and improves the system throughput [15–17]. The architecture of HDFS adopts master-slave structure, and the architecture of HDFS system is as Figure 2.

As can be seen from the HDFS system architecture diagram, each HDFS cluster consists of a NameNode and multiple DataNodes. The file storage method of HDFS is to divide a file into multiple data blocks of the same size, and then store the data blocks in different DataNodes. The following will introduce the main components and functions in HDFS:

1. **NameNode**: The NameNode is the manager of the entire file system, coordinating and managing all the DataNodes of the entire system. It provides the directory information of the file system, the block information of each file, and the corresponding information of the DataNode storing the data block.

2. **Secondary NameNode**: The NameNode is the most critical computing node in the entire HDFS. Once the NameNode fails, the entire system will be paralyzed. The Secondary NameNode is the backup node of the NameNode and periodically reads the running log of the NameNode.

3. **DataNode**: After the file is divided into data blocks of the same size, it will be stored in the DataNode. DataNodes are mostly inexpensive machines, which are prone to failure. In order to achieve high fault tolerance of the system and ensure that data can be accessed, one data block will be stored in multiple DataNodes.

4. **Client**: When the Client needs to read and write files in the system, it needs to first access the NameNode to read the metadata of the file system, and after obtaining the relevant information, access the corresponding DataNode to read and write files.

MapReduce is another core component of the Hadoop-distributed computing framework. MapReduce is a parallel programming model, which can process a large amount of distributed unstructured data, and can generate summary results, providing scalability across cluster nodes [18, 19]. It is mainly divided into two steps: Map and Reduce. Map is a key-value pair that generates a set of key/value intermediate values for unstructured data as input. Then these key-value pairs are used as input to the Reduce operation, and Reduce sorts these values and merges them into a smaller set of values or a single value. Typical applications of MapReduce include word counting, distributed sorting, and pattern-based search [20]. The MapReduce running framework is as Figure 3.

The specific operation of MapReduce is mainly to separate the calculation program and then merge the results. In the MapReduce task, the input data will be divided into a set of many unrelated key/value pairs, and many Map tasks process the key/value together. MapReduce arranges the output order of the Map, combines a key value combination at the same time, inputs the Reduce task after the combination, and outputs the final result through the calculation of the Reduce task.

### 2.2.2. Spark-Distributed Computing Framework

The Spark-distributed computing framework is an emerging computing framework that can process massive amounts of data quickly and conveniently. Spark was originally designed based on the MapReduce parallel computing model. Not only inherits the excellent distributed computing characteristics of MapReduce, but more importantly, introduces a new type of core data structure resilient distributed data sets (RDDs). Spark provides the RDD functional programming model interface, that is, method-oriented programming, memory-based cluster computing, and RDD to achieve parallel operations, which effectively improves the efficiency of data processing [21–23]. Spark contains a lot of modules, and the core modules inside it are as Figure 4.

1. **Spark Core**: Also known as the Spark core component module, designed an abstract directed acyclic graph memory-based computing framework for the entire cluster ecological environment. In such a computing framework, the core module must be RDD, which has the characteristics of strong scalability and high fault tolerance. In the cluster, each worker child node can be unified into the basic RDD
Figure 2: HDFS system architecture.

Figure 3: MapReduce running framework.

Figure 4: Spark internal core modules.
At runtime, two types of operators are supported for each RDD operation, namely transformation and action.

(2) Spark SQL: This module provides an SQL-like query engine module for R&D engineers who are proficient in traditional databases and SQL statements [24]. Structured data have the characteristics of fast processing. Data frame and data set are the core basic programming abstraction. It can quickly build various data sources, such as HDFS-distributed data files, hive table data, associated relational source databases, and RDDS.

(3) Spark Streaming: This module was developed for real-time processing of dynamic data tasks, some mainstream data streams or instant log communication, such as Kafka, Flume, and ZeroMQ. Not only can handle such a simple socket data source but also provides a visual master UI for these operations.

(4) GraphX: The main function of this module is a distributed graph computing framework, and this module contains two modes: graph storage and graph calculation. The storage method has evolved from edge segmentation to point segmentation at the beginning, and performance has been significantly improved. Graph computing is based on bulk synchronous parallel (BSP), and regardless of the vertical serial mode or the horizontal parallel mode, the key node is to set the synchronization point to realize the super step mode.

(5) MLlib: This module contains some common libraries used in machine-learning algorithms. It not only includes clustering, classification, recommendation, feature engineering, regression, statistics, and other learning libraries, but also provides a unified API interface for these learning libraries to realize low-cost research on machine-learning or data mining algorithms.

2.3. Data Mining-Related Algorithms

2.3.1. Decision Tree Algorithm. Decision tree algorithm is a method for in-depth analysis of classification problems. The decision tree is a directed acyclic tree. The general framework of the decision tree algorithm is the same. Greedy non-backtracking algorithm is adopted to construct the decision tree in a top-down manner [25–27]. If the dataset is $C$ and the category set is $C = \{C_1, C_2, ..., C_k\}$, where the attribute $V$ divides $C$ into multiple subsets, the nonoverlapping value of $V$ is $V = \{v_1, v_2, ..., v_n\}$, $C$ is divided into $C = \{N_1, N_2, ..., N_n\}$, and the mean in $N_n$ is $v_i$. If $|N_n|$ is defined as the number of examples of $V = v_i$ in the data set $C$, $|C_j|$ is the number of examples of $V = v_i$ and $C = C_j$, that is, $V = v_i$ and the category is $C_j$. Then, there are the following definitions:

(1) The probability of $C_j$ occurrence:

$$P(C_j) = \frac{|C_j|}{C} = freq(C_j, C).$$

(2) The probability of $V = v_i$ occurrence:

$$P(v_i) = \frac{|N_n|}{C}.$$

(3) The probability that the attribute $V = v_i$ example has the category condition $C_j$:

$$P(C_j|v_i) = \frac{|C_j|}{|N_n|}.$$

(4) Category information entropy:

$$H(C) = -\sum_j P(C_j) \log P(C_j) = -\sum_j \frac{|N_j|}{|C|} \log \frac{|N_j|}{|C|} = Info(C).$$

(5) Category conditional entropy calculation:

$$H(C|V) = -\sum_i P(V_j) \sum_j P(C_j|V_j) \log P(C_j|V_j) = -\sum_i \frac{|N_j|}{|V_i|} Info(N_n) = Info_v(C).$$

(6) Information gain:

$$I(C, V) = H(C) - H(C|V) = Info(C) - Info_v(C) = gain(v).$$

(7) Information entropy of attribute $v$: 

$$H(V) = -\sum_i P(V_i) \log P(V_i) = -\sum_i \frac{|V_i|}{|V|} \log \frac{|V_i|}{|V|} = Info_v(C).$$
6 Security and Communication Networks

\[ \text{Info}(V) = - \sum_{i} P(v_i) \log P(v_i) = - \sum_{i=1}^{k} \frac{|N_i|}{|C|} \log \frac{|N_i|}{|C|} = \text{Info}(C) - \text{Info}_c(C) = \text{split}_\text{Info}(v). \] (7)

(8) Information gain rate:

\[ \text{gain} \_\text{ration}(v) = \frac{I(C, V) - H(V)}{\text{split}_\text{Info}(v)} \] (8)

2.3.2. Association Rule Algorithm. In the object database, there are hidden potential rules between objects. In order to explore whether there is some correlation information between these rules that are not easy to be found, the concept of association rules is proposed \[28–30\]. The meaning of association rules is proposed \[28–30\].

(1) Transaction set and data items: replace the diversified original data with excavability, that is, discretization and dataization. At the same time, a new data item is generated, which is defined as \( T = \{t_1, t_2, ..., t_n\} \), where \( n \) is a non-negative integer, and each subelement is independent of each other. Suppose the transaction set of all data items is defined as \( T = \{t_1, t_2, ..., t_n\} \), where each transaction \( t_n \) includes some subitem sets

\[ \text{Sup}(X \rightarrow Y) = \text{Sup}(X \cup Y) = \frac{P(X \cup Y)}{P(I)} = \frac{\text{Count}(X \cup Y)}{|m|} \] (10)

In formula (10), the degree of support determines the frequency rule applicable to a given data set, and retains its implicit rules.

(2) Support number and degree of support: An important attribute of the data item set is its support number. In the mathematical definition, it means the number of transactions that include the specified data item. The expression of the support number \( \sigma(X) \) of the data item \( X \) is as follows:

\[ \sigma(X) = | \{t_n | X \subseteq t_n, t_n \in T \} |. \] (9)

By calculating the number of support for each data item, for the established item set \( T \). Assuming that the total transaction database length is \( m \), by calculating the proportion of \( \{X, Y\} \) in the entire transaction item set, the data item support is obtained, which shows the deductible probability, and the expression is as follows:

(3) Confidence degree: For the obtained intermediate related data, the credibility degree is calculated to obtain the confidence degree of the \( \{X, Y\} \) association rule, which can also be called the credibility degree \[31\]. It is determined that the frequency of occurrence of \( Y \) is contained in event \( X \), and the expression is as follows:

\[ \text{Conf}(X \rightarrow Y) = \frac{P(Y | X)}{P(X)} = \frac{\text{Sup}(X \cup Y)}{\text{Sup}[X]} \] (11)

In formula (11), assuming that the higher the confidence of the rule of \( X \rightarrow Y \), the greater the probability of occurrence of the \( Y \) transaction contained in \( X \), and the strong association rule can be effectively calculated.

(4) Frequent itemsets: a rule standard belonging to a kind of itemsets. Suppose \( A = \{a_1, a_2, a_3, ..., a_m\} \) is a collection of items, \( D = \{d_1, d_2, d_3, ..., d_n\} \) represents a collection of database transactions, and if any \( d_n \in D \) is set, there is \( d_n \subseteq A \). Let any subset \( A_j \) of \( A \) be the itemset, suppose \( A_j \) contains \( K \) elements, and call \( A_j \) the \( K \) itemset. If the itemset \( A_j \) satisfies the minimum support degree \( \min \_\text{support} \), then \( A_j \) is called the frequent itemset.

(5) Lift: Generally speaking, the validity of association rules is evaluated according to the calculated ratio of \( X \rightarrow Y \)'s confidence and support. If the result is greater than 1, and both the minimum support threshold and the minimum confidence threshold are met, it can be described as an effective strong association rule \[32\]. On the contrary, it shows that \( X \) and \( Y \) are independent of each other and have no influence on each other, and the correlation result calculated by instantaneously meeting the threshold is not valid. The expression is as follows:

\[ \text{lift}(X \rightarrow Y) = \frac{P(Y | X)}{P(Y)} = \frac{\text{Conf}(X \rightarrow Y)}{\text{Sup}(Y)} \] (12)

(6) Association rules: The result data information that is excavated is a basic implicit rule shaped like \( X \rightarrow Y \), and the standard to measure this rule is calculated based on support, confidence, and promotion.
3. Adaptive Decision Algorithm for Real Estate Marketing

Aiming at the problems of low stability, efficiency, and complexity of marketing decision-making algorithm, an adaptive decision-making algorithm for real estate marketing based on Big Data analysis is proposed. By constructing association rule algorithm, the rules between real estate marketing and related factors are mined. In Spark-distributed computing platform, the optimization idea of association mining is designed. The decision tree algorithm is used to select the attribute characteristics of real estate marketing. In the pruning process of the decision tree, the triple standards of classification accuracy, stability, and complexity are used to further optimize the pruning process, so as to build an adaptive decision-making model of real estate marketing based on the aforementioned contents. In this process, we need to pay attention to the association relationship among association algorithm, attribute characteristics, and adaptive decision model.

3.1. Building an Algorithm for Association Rules

(1) Define FP_TREE: Its data structure is a kind of tree structure of grid structure, including root nodes and some other nodes. These nodes form a subgrid structure as the child nodes of the root or descendants [33]. Each node structure generated is mainly composed of three attributes: transaction value $TV$, global support number $Global_Count$, and link attribute $Link$. The transaction value records the corresponding transaction data item represented by the node. The global support count is when the current $TV$ is inserted, the attribute value is set to 1, and if its $TV$ is equal to the currently inserted $TV$, it will be accumulated by 1. The global count attribute records the node support presented to its $TV$. In the process of building the tree, all the support of frequent and infrequent patterns is recorded in the global count attribute [34]. Link attributes are used to generate the associated subnet structure or descendants of the root. The entire node can be expressed as follows:

$$Node_i = \{TV, Global_Count, Link\}. \quad (13)$$

The $TV$ of the root node is assumed to be NULL and can be divisible by all $TV$. First, check the separability between the first two $TV$. If the first $TV$ divides the second one, or is divided evenly, then the dividend will be the parent node, and the divisor will be its descendant node. If they are not divisible by each other, the two $TV$ are inserted into the tree as the only independent point.

(2) Construct FP_TREE: After the $TV$ data set is created, the transaction value $TV$ is converted into nodes one by one and inserted into the FP-tree data structure.

When constructing FP-tree, the following rules should be followed:

Rule 1: If the $TV$ of node $n_i$ is not the divisor or dividend of any existing node in the FP-tree structure, just insert it as a new node, and its local count will be 1.

Rule 2: If the $TV$ of node $n_r$ and $n_e$ are equal, then $r = s$. If the current $TV$ read is equal to the $TV$ of $n_r$, the insertion process accumulates the local count attribute of node $n_r$ by 1.

Rule 3: If the $TV$ of the node $n_r$ is the divisor of several other $TV$, then the global count attribute value of the node will be equal to the number of dividends of $n_i$ in the conversion data set, or the number of parent nodes from another perspective.

According to the construction rules, the node $N_j$ in the tree can be defined, and the node set is set to $N_k$, and in a given $k$-item set, the corresponding $TV$ data set can be calculated. The node set of the tree can be expressed as follows:

$$N_j = \{D_{TV}, D_{TV} = D_{Link}(N_j, N_k)\}. \quad (14)$$

In formula (14), $D_{TV}$’s $D_{Link()}$ method means that node $N_j$ and other nodes $N_j, N_k$ have a parent-child relationship, and there can be multiple offspring child nodes, so the output result is a sequence of offspring nodes. Each node contains the $Global_Count$ attribute, so the possible support count of the node set can be expressed as:

$$NS_{sup} = \sum_{i=1}^{n} N_{i,support}. \quad (15)$$

Based on the Spark-distributed computing platform, an optimization idea of association mining is designed, grouping strategies are improved, and storage resource occupancy ratio is compressed to conduct efficient mining. The FP-tree structure contains a root node. The root node only represents the hidden value. It is initially included in the empty structure of other distributed computing threads. Its $TV$ value includes all possible original data items and nonvirtual additional data items. If there are $n$ unique data items $i_1, i_2, ..., i_n$ and the root $Global_Count$ is set to zero, the data set sequence structure can be expressed as follows:

$$D_n = \prod_{j=n+1}^{m} p(i_j). \quad (16)$$

Owing to the introduction of Spark’s distributed processing mechanism, building a tree needs to be executed in two parts: The first part is to calculate the $Global_Count$ of the $TV$ transaction item in parallel to obtain the $\langle item, sup_value \rangle$ data set. The second part is to construct the parent-child relationship link of the tree in order to effectively traverse the search.

3.2. Choose the Characteristics of Real Estate Marketing Attributes. An important step in establishing an adaptive
A decision-making model for real estate marketing is to determine the set of real estate marketing attributes. In the real estate marketing data, the amount of data information is too large, and there is a lot of redundancy, which greatly affects the construction of the decision tree. Based on this, the information gain rate is used for attribute selection. Attributes are divided into discrete and continuous types.

For discrete attributes, the general method is based on the n values of discrete attributes A, sample S divides the sample set into n subsets $S_1, S_2, ..., S_n$, and calculates the information gain rate of A division S [35].

The range of continuous-valued attributes is usually divided into discrete interval sets. Assuming that A is a continuous attribute with values in a continuous interval, the first step is to arrange the samples of the training set X from small to large according to the value of the attribute A. If there are m values of A in the training sample set, $V_1, V_2, ..., V_m$ is the sample sorting, and the average value of adjacent values is calculated one by one as the split point:

$$V = \frac{(V_i + V_{i+1})}{2} \quad (1 \leq i < m).$$  

3.3. Building an Adaptive Decision-Making Model for Real Estate Marketing. The characteristics of real estate marketing data are as follows: in the development and operation of real estate and real estate, the real estate industry is an infrastructure industry. It belongs to the field of fixed asset investment and is greatly affected by the national macro-economic policies. Therefore, just like the national economy, the real estate marketing data also shows strong periodicity. Moreover, real estate marketing data also occupy this leading real estate in the national economy. The supply and demand of the real estate market is mainly subject to the local economic development level. If the economic development of a region is good, the income level of residents in the region will be higher, and the demand of the real estate market in the region will increase. Therefore, with the increase of real estate prices, the development of the real estate industry in the region will naturally show a good trend. According to the characteristics of real estate marketing data, the decision tree pruning algorithm is optimized using the weight-based discrimination method and three criteria of classification accuracy, stability, and complexity, so as to build an adaptive decision model for real estate marketing.

1. Classification accuracy: It is one of the most important evaluations of the performance of decision tree classifiers. Define $a_T(t)$ as the classification accuracy of node t on T, $N_t$ is the number of sample instances that enter t in T, $N_s$ is the number of samples that are correctly classified, there are a total of N sample instances, and the proportion of instances that enter t in the total instances is $P_T(t)$, the classification accuracy of the decision tree is as follows:

2. Stability: Owing to the existence of internal deviation of samples, there are certain differences in classification accuracy due to different sample sets. The stability of decision tree is affected by classification accuracy. The higher the classification accuracy, the higher its stability. The stability of the decision tree is as follows:

3. Complexity: The complexity of the decision tree is affected by the reduction rate. The higher the reduction rate, the smaller the complexity. Let $C_T$ denote the reduction rate of the decision tree, $N_t$ the number of nodes in the decision tree, and $N_T$ the total number of samples. The complexity of the decision tree is as follows:

Thus, an adaptive decision-making model for real estate marketing is constructed, which is expressed as follows:

With the rapid development of Internet technology, the application of information technology in China’s real estate industry is becoming more and more popular, and the real estate marketing has accelerated the trend towards e-commerce. There are a large number of housing commodity information display and consumer query demand records on the Internet, and these huge amounts of data are undoubtedly a gold mine of real estate marketing data to be developed, which also brings various problems to the real estate information analysis technology. Therefore, it is necessary to break through the tradition of real estate marketing and information investigation and analysis, apply new information technology, and make decisions through the idea and technology of data mining, so as to realize the adaptive decision-making of real estate marketing.

4. Experimental Analysis

4.1. Experimental Environment and Data. In order to verify the effectiveness of the real estate marketing adaptive decision algorithm based on Big Data analysis, the experiment is mainly based on the Spark-distributed computing platform. The scale of the Spark cluster is 4 nodes, including 1 Master main drive node and 3 Slave work nodes. The hardware configuration of each node in the cluster is Intel E5 3.70 GHz 4-core CPU, 16G memory and 500 GB high-speed 1/O hard disk. The software configuration is a 64-bit Cent OS 6.5 stable version. The Hadoop version is 2.7.0, the Apache

$$A_T(t) = \sum_{i \in T} P_T(t)a_T(t).$$  

$$S_T = \min \left\{ \frac{a_T(t)}{a_T(t)} \right\}.$$  

$$C_T = 1 - \frac{N_t}{N_T}.$$  

$$I(T) = \sum_{j=1}^{3} w_j v_j.$$
Spark version is 2.2.0, the JDK version is 1.8.144, and the programming language is Scala 2.10.0. The data used in the experiment is from the IBM Quest Synthetic Data Generator association rule simulation data set generation tool, which produces two sparse data sets T10I4D5000K and Mushroom. The number of transactions in the T10I4D5000K data set is 5 MB, the average transaction length is 10, the average simulation length is 4, and the data set is relatively rare. The Mushroom data set contains relevant feature data, including quantity, items, and average transaction length, to verify the effectiveness of the algorithm.

4.2. Comparison of Stability of Adaptive Decision-Making in Real Estate Marketing. In order to verify the stability of the proposed algorithm, the classification accuracy of real estate marketing adaptive decision is taken as the evaluation index. The higher the classification accuracy of real estate marketing adaptive decision is, the higher the stability of real estate marketing adaptive decision is. By comparing the algorithm in Lessmann et al.’s study [8] and the algorithm in Gáti and Bauer’s study [9] with the proposed algorithm, we get the comparison results of the classification accuracy of real estate marketing adaptive decision-making of different algorithms, as shown in Figure 5.

It can be seen from Figure 5 that under different characteristic data sets, the average classification accuracy of real estate marketing adaptive decision of the algorithm by Lessmann et al.’s [8] is 86.2%, the average classification accuracy of real estate marketing adaptive decision of the algorithm by Gáti and Bauer [9] is 77%, while the average classification accuracy of real estate marketing adaptive decision of the proposed algorithm is only 93%. It can be seen that compared with the algorithm by Lessmann et al. [8] and the algorithm by Gáti and Bauer [9], the proposed algorithm has higher classification accuracy and higher stability of real estate marketing adaptive decision.

4.3. Comparison of Complexity of Adaptive Decision-Making in Real Estate Marketing. Further verify the complexity of adaptive decision-making in real estate marketing of the proposed algorithm and take the reduction rate of adaptive decision-making in real estate marketing as the evaluation index. The higher the reduction rate of adaptive decision-making in real estate marketing, the smaller the complexity of adaptive decision-making in real estate marketing. The algorithm by Lessmann et al. [8], the algorithm by Gáti and Bauer [9], and the proposed algorithm are compared respectively to obtain the reduction rate of adaptive decision-making in real estate marketing of different algorithms. The comparison results are shown in Figure 6.

It can be seen from Figure 6 that under different characteristic data sets, the average reduction rate of real estate marketing adaptive decision of the algorithm by Lessmann et al. [8] is 84.4%, the average reduction rate of real estate marketing adaptive decision of the algorithm by Gáti and Bauer [9] is 72.2%, and the average reduction rate of real estate marketing adaptive decision of the proposed algorithm is as high as 95.8%. Therefore, compared with the algorithms by Lessmann et al. [8] and the algorithm by Gáti and Bauer [9], the proposed algorithm has higher simplification rate of real estate marketing adaptive decision and less complexity of real estate marketing adaptive decision.

4.4. Comparison of Adaptive Decision-Making Efficiency of Real Estate Marketing. On this basis, the adaptive decision-making efficiency of real estate marketing of the proposed algorithm is further verified. Taking the adaptive decision-making time of real estate marketing as the evaluation index, the shorter the adaptive decision-making time of real estate marketing, indicating that the adaptive decision-making efficiency of real estate marketing is higher. By comparing
the algorithm by Lessmann et al. [8], the algorithm by Gáti and Bauer [9], and the proposed algorithm, the comparison results of real estate marketing adaptive decision-making time of different algorithms are obtained, as shown in Figure 7.

As can be seen from Figure 7, with the increase of feature data sets, the adaptive decision-making time of real estate marketing with different algorithms increases. When the feature data set is 1000, the real estate marketing adaptive decision-making time of the algorithm by Lessmann et al. [8] is 50 s, the real estate marketing adaptive decision-making time of the algorithm by Gáti and Bauer [9] is 48 s, while the real estate marketing adaptive decision-making time of the proposed algorithm is only 23 s. It can be seen that compared with the algorithm by Lessmann et al. [8] and the algorithm by Gáti and Bauer [9], the proposed algorithm has shorter adaptive decision-making time in real estate marketing and can effectively improve the efficiency of adaptive decision-making in real estate marketing.

To sum up, the average decision classification accuracy of the algorithm by Lessmann et al. [8] is 86.2%, the average decision classification accuracy of the algorithm by Gáti and Bauer [9] is 77%, and the average decision classification accuracy of the proposed algorithm is only 93%. The average decision reduction rate of the algorithm by Lessmann et al. [8] is 84.4%, the average decision reduction rate of the algorithm by Gáti and Bauer [9] is 72.2%, and the average decision reduction rate of the proposed algorithm is as high as 95.8%. When the feature data set is 1000, the decision time of the algorithm by Lessmann et al. [8] is 50 s, the decision time of the algorithm by Gáti and Bauer [9] is 48 s, and the decision time of the proposed algorithm is only 23 s. The proposed real estate marketing adaptive decision algorithm has good results and solves the problems of low marketing decision stability and efficiency and large marketing decision complexity in the current marketing adaptive decision algorithm.

5. Conclusion

This paper proposes an adaptive decision algorithm for real estate marketing based on Big Data analysis, which gives full play to the advantages of Big Data technology. In spark-distributed computing platform, combined with association rules and decision tree algorithm, the adaptive decision of real estate marketing is realized. Its real estate marketing adaptive decision-making has high stability and efficiency, which can effectively reduce the complexity of real estate marketing adaptive decision-making. However, in the process of adaptive decision-making of marketing data because the tested data set is not multidimensional data and the memory resources of computer hardware are sufficient, there are no abnormal phenomena such as memory overflow in the process of building FP-tree. Therefore, in the next research, it needs to be applied to the actual large-scale cluster, and the spark cache mechanism is used for voltage division processing, so as to avoid the shortage of shared memory resources.

Data Availability

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding this work.

References

[1] K. Takamura, K. Tachibana, T. Kusakabe, K. Nakai, and M. Kondoh, “New Japanese regulatory frameworks for post-marketing management of pharmaceutical products,” *Pharmaceutical Research*, vol. 37, no. 7, Article ID s11095, 2020.
[2] A. Hemonnet-Goujot, D. Manceau, and C. Abecassis-Moedas, “Drivers and pathways of NPD success in the marketing-external design relationship,” *Journal of Product Innovation Management*, vol. 36, no. 2, pp. 196–223, 2019.
[3] T. Andrade, T. Z. Albertini, L. G. Barioni, and S. R. D. Medeiros, “Perception of consultants, feedlot owners, and packers regarding management and marketing decisions on feedlots (Part II): A national survey in Brazil,” *Canadian Journal of Animal Science*, vol. 100, no. 4, pp. 1139–1145, 2020.
[4] Y.-T. Chen, E. W. Sun, and Y.-B. Lin, “Coherent quality management for big data systems: A dynamic approach for stochastic time consistency,” *Annals of Operations Research*, vol. 277, no. 1, pp. 3–32, 2019.
[5] Z. Xiang and M. Xu, “Dynamic game strategies of a two-stage remanufacturing closed-loop supply chain considering Big Data marketing, technological innovation and overconfidence,” *Computers & Industrial Engineering*, vol. 145, no. 7, Article ID 106538, 2020.
[6] Q. Fan, “Architecture model of intelligent marketing system based on big data,” *Advances in Management*, vol. 12, no. 21, pp. 1275–1279, 2019.
[7] A. Jabbar, P. Akhtar, and S. Dani, “Real-time big data processing for instantaneous marketing decisions: A problematization approach,” *Industrial Marketing Management*, vol. 90, no. 10, pp. 558–569, 2020.
[8] S. Lessmann, J. Haupt, K. Coussément, and K. W. De Bock, “Targeting customers for profit: An ensemble learning framework to support marketing decision-making,” Information Sciences, vol. 557, no. 1, pp. 286–301, 2019.

[9] M. Gati and A. Bauer, “Marketing decision-making in Hungarian SMEs,” Market-Tržište, vol. 31, no. 1, pp. 39–59, 2019.

[10] R. Pourmoyadjie and L. Rulundnielsen, “Optimizing pig marketing decisions under price fluctuations,” Annals of Operations Research, vol. 6, no. 3, pp. 12–19, 2020.

[11] R. Chierici, A. Mazzucchelli, A. Garcia-Perez, and D. Vrontis, “Transforming big data into knowledge: The role of knowledge management practice,” Management Decision, vol. 57, no. 8, pp. 1902–1922, 2019.

[12] J. M.-T. Wu, G. Srivastava, M. Wei, U. Yun, and J. C.-W. Lin, “Fuzzy high-utility pattern mining in parallel and distributed Hadoop framework,” Information Sciences, vol. 553, no. 4, pp. 31–48, 2020.

[13] C. Kavitha and X. Anita, “Task failure resilience technique for improving the performance of MapReduce in Hadoop,” ETRI Journal, vol. 42, no. 5, pp. 748–760, 2020.

[14] A. Mostafaeipour, A. J. Rasbanjani, M. Ahmadi, and J. A. Dhanraj, “Investigating the performance of Hadoop and Spark platforms on machine learning algorithms,” The Journal of Supercomputing, vol. 77, no. 6, pp. 1273–1300, 2021.

[15] M. Maghsoudloo and N. Khoshavi, “Elastic HDFS: Interconnected distributed architecture for availability-scalability enhancement of large-scale cloud storages,” The Journal of Supercomputing, vol. 76, no. 1, pp. 174–203, 2019.

[16] K. Manjula and S. M. Sundaram, “Optimized approach (SPCA) for load balancing in distributed HDFS cluster,” SN Computer Science, vol. 1, no. 2, p. 42979, 2020.

[17] S. Zhang, “User interest based teaching information resource management method in HDFS mode,” Modern Electronics Technique, vol. 42, no. 11, pp. 87–89, 2019.

[18] P. Sowkuntla and P. Prasad, “MapReduce based parallel fuzzy-rough attribute reduction using discernibility matrix,” Applied Intelligence, vol. 162, no. 4, pp. 1–20, 2021.

[19] O. Kulkarni, S. Jena, and V. Ravi Sankar, “MapReduce framework based big data clustering using fractional integrated sparse fuzzy C means algorithm,” IET Image Processing, vol. 14, no. 12, pp. 2719–2727, 2020.

[20] P. Sowkuntla, S. Dunna, and P. S. V. S. S. Prasad, “MapReduce based parallel attribute reduction in Incomplete Decision Systems,” Knowledge-Based Systems, vol. 213, no. 2, Article ID 106677, 2021.

[21] S. Kang, S. Lee, and J. Kim, “Distributed graph cube generation using Spark framework,” The Journal of Supercomputing, vol. 76, no. 2, pp. 8118–8139, 2019.

[22] T. R. Rao, S. K. Ghosh, and A. Goswami, “Mining user-user communities for a weighted bipartite network using spark GraphFrames and Flink Gelly,” The Journal of Supercomputing, vol. 77, no. 6, pp. 5984–6035, 2021.

[23] W. Xiao and J. Hu, “SWEclat: A frequent itemset mining algorithm over streaming data using Spark Streaming,” The Journal of Supercomputing, vol. 76, no. 3, pp. 7619–7634, 2020.

[24] Z. He, Q. Huang, Z. Li, and C. Weng, “Handling data skew for aggregation in spark SQL using task stealing,” International Journal of Parallel Programming, vol. 48, no. 9, pp. 941–956, 2020.

[25] L. Li, S. Dai, Z. Cao, J. Hong, S. Jiang, and K. Yang, “Using improved gradient-boosted decision tree algorithm based on Kalman filter (GBDT-KF) in time series prediction,” The Journal of Supercomputing, vol. 76, no. 9, pp. 6887–6900, 2020.