Research on Efficient Mining Technology of Highly Confidential Data in Cloud Platform Network

Liang Cai[1], Yaqiong Cheng[2]
[1]Lanzhou University of Arts and Science, Lanzhou, 730010, China
[2]Lanzhou Vocational and Technical College, Lanzhou, 730010, China

Abstract: In Cloud Computing area, many important information is hidden in large data, to find an encryption method that can process the data effectively, which means that no large impact is exerted on inquiry performance while the safety of data can be ensured, the research on efficient mining technology of highly confidential data in cloud platform network is necessary. Based on the principles of mining technology, the overall structure of mining system is constructed, and through the data encryption and decryption process, the data mining interaction is realized, then the flattening mining is performed on the data after interaction, thus to realize the highly confidential data mining. Simulated experiment of efficient mining technology of highly confidential data in cloud platform network is performed and experimental results show that compared with traditional algorithm, the highly confidential data mining technology in cloud platform network is able to improve the mining efficiency.

Keywords: Compensation technique; Electricity; Energy-saving optimization; Control system

1. Introduction

In the era of knowledge economy, information and knowledge have become the most important strategic resources. How to use knowledge discovery and data mining techniques to convert the data collected in social development and national economic operations and during business operations into useful information and knowledge, is the core to increase the competition of an organization and even a country [1]. However, there are large amount of important data, and its security issues cause more and more attention. Access control mechanism is used in traditional database to protect the sensitive information, but it cannot prevent external personnel from attacking effectively, nor internal personnel nor DBA from deliberately sabotage [2]. How to ensure the security of database in such non-credible environment has become a hot topic of research. The protection of sensitive information in database through encryption mechanism is an effective means, such as identity cards, bank accounts, personal health information, and so on. Using traditional encryption algorithms (such as: DES, ADS) to encrypt the data, for the exact matching query of the string, the encrypted data can be directly queried without decrypting the data [3]. After the data is encrypted, some inherent attributes change, and there is no way to directly compare the ciphertext data. If all encrypted data is decrypted and then queried, this operation is costly and will greatly affect the query performance, which is not practical in practice. Therefore, it is necessary to find an encryption method that can effectively process character data, which can ensure its security without having a large impact on query performance.

2. Construction of Highly Confidential Data Mining System in Cloud Platform Network

2.1. Efficient mining technology flow of highly confidential data

When the ciphertext data is stored in the database, the performance for query of ciphertext data is drastically reduced. To balance the security and query performance of sensitive information in the database, some staff conducted research on database encryption [4]. Based on a relatively simple encryption method, for an integer P, its corresponding encryption value $c = \sum_{j=0}^{p} R_j$, where Rj is the j-th pseudo-random number generated by the pseudo-random generator R. Since the difference between the two ciphertext values and their corresponding two plaintext values are proportional to each other, the attacker can derive the probability distribution of its plaintext value from the probability distribution of the ciphertext value, and derive its corresponding plaintext value [5]. This encryption method has poor security and is vulnerable to known plaintext attacks and statistical attacks. In the context that database is taken as a service (Database As a Service), the method for querying encrypted data is proposed. When storing, in addition to regular encryption for the tuples in the relational table, a barrel number is added to each attribute value. The barrel number indicates that the value of the plaintext data falls within a certain interval. When querying, the SQL query submitted by the client can be directly executed on the encrypted data without decryption. And it is further bucketed for the optimal bucketing algorithm to minimize the cost of the query [6]. However, in this method, the recordset returned to the client may contain some records that do not meet the query conditions and need to be decrypted and processed again. Moreover, this method is very expensive for multi-table join queries. By an orderly encryption method, a target distribution function is given, the plaintext value is converted to ciphertext, so that the ciphertext not only maintains order, but also obeys the distribution of an objective function. Since the ciphertexts are kept in order, the ciphertexts can be directly equated and scoped without decryption, and MAX, MIN, COUNT, and ORDER BY queries can also be performed. Obviously, this method is less secure because the ciphertext remains orderly and vulnerable to selective ciphertext attacks. That is, if the attacker can select a certain number of plaintext (or cipher texts) and encrypt (or decrypt) them into the corresponding ciphertext (or plaintext), then he can estimate the plaintext value of the ciphertext with a greater probability [7].

2.2. Structure of data mining system

In view of the quite complex internal structure of the DBMS, it is difficult to modify the existing DBMS (such as Oracle, SQL SERVER, etc.), so DBMS external methods are used for data encryption, as shown in Figure 1,The encryption/decryption layer is added between the DBMS and the application program, and the data is encrypted and stored and queried exclusively through this layer. When storing, the data in the application needs to be processed in two ways before it is stored in the database. On the one hand, the encryption/decryption layer invokes the encryption algorithm to encrypt the character data and store it in the database; On the other hand, in order to improve the performance of the encrypted data query, a new field is added to the original data table, and the characteristic value of the character data is stored [8]. When querying, the two-stage query method is used. Firstly, a coarse query is performed on the encrypted data, and the new fields in the encrypted data table are mainly used to filter most records that are not related to the query conditions; then, the encrypted data in the remaining records is decrypted, and a refined query (Refined Query) is performed on the decrypted data, finally achieving the query target. In the encryption/decryption layer in Figure 1, metadata is some function mapping rules that are used to modify storage and query statements. When storing, in addition to the encrypted data, these rules are used to modify the SQL statement and store the characteristic values of the encrypted data. When querying, the query statement is modified by these rules and converted into SQL statement for querying encrypted data. Encryption and decryption are function modules for encryption and decryption of sensitive information columns in the database by the encryption/decryption function [9]. In traditional relationship model R(X1, ···, Xr, ···, Xn), X1 represents the attributes to be encrypted and stored into database with encryption relationship model RE(X1, …, XrE, …, Xn, XrS), where XrE in RE is a mapping of Xr in R, XrE =E(Xr), E is the encryption algorithm, XrS is an additional attribute used to store the characteristic value of Xr, and called characteristic attribute. Obviously, XrE in RE can be obtained only by using the encryption algorithm to process Xr in R. While XrS in RE is used to store the characteristic value of data in Xr, how to extract the characteristic value of data in Xr is a key problem. In general, when extracting characteristic values, the following two aspects should be considered: (1) To ensure the security of encrypted data, when extracting the characteristic values of data, the characteristic value itself cannot leak information in Xr; (2) To improve the query performance of encrypted data. In the first phase of the coarse query, most of the records that are irrelevant to the query conditions are filtered by using characteristic value. This is an important part of improving query performance [10].

2.3. Encryption and decryption in data mining

In order to better understand the terminology used in encryption algorithms, this section will detail the related terms in cryptography. The original message is called plaintext, and the process of encryption is to use a certain method or algorithm to disguise the message to hide its real content. The message encrypted by the key is called ciphertext, decryption is the process of converting ciphertext into plaintext [11]. Cryptography Algorithm: its essence is a mathematical function used for encryption and decryption operations. Key: key information other than the cryptography algorithm required for encryption or decryption operations. Encryption algorithm: Some rules used by message encryptors for encrypting plaintext [12]. The specified object transmitted by the message is called Receiver, and some rules used by the receiver to decrypt the ciphertext are called Decryption Algorithm. The encryption and decryption operations are shown in Figure 2. In recent decade, with the increasing size of data, data mining technology has become one of the important technical research hotspots for scientists worldwide. The accepted definition of data mining is given by Fayyad: Data mining is a process to determine the valid, unknown, novel, potentially usable, and ultimately understandable patterns in data [13]. The data mining process is shown in Figure 3.

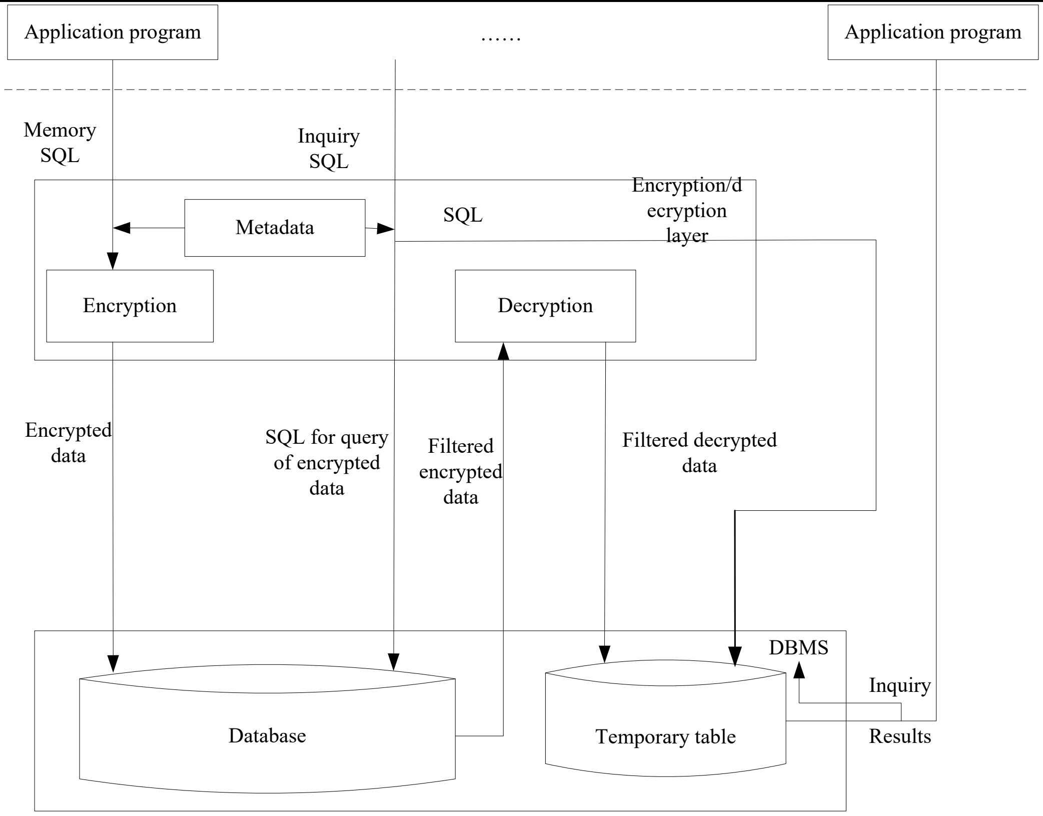


Figure 1. Structure of encrypted data storage and query system

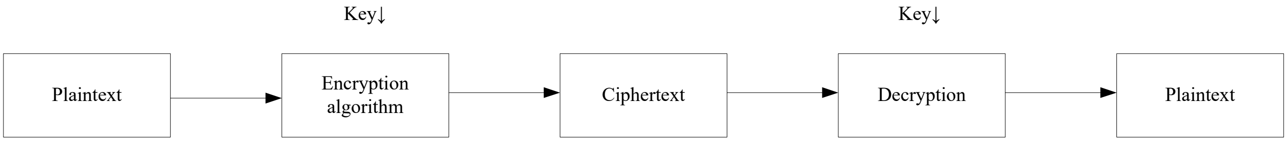


Figure 2. Encryption and decryption operations

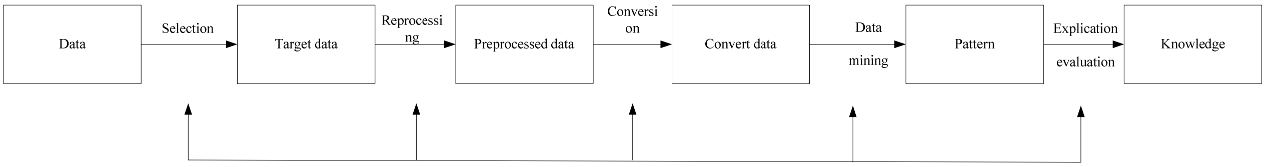


Figure 3. Data mining process

The types of these data can be structured, semi-structured, or heterogeneous. Knowledge discovery methods can be digital, non-numeric, or inductive. The knowledge that is eventually discovered can be used for decision support information management, self-maintenance, and data query optimization [14]. At the present stage of research at home and abroad, data mining algorithms mainly include association rule method, neural network method, decision tree method, rough set method, fuzzy set method, genetic algorithm and so on. The algorithm used in efficient data mining is the Apriori algorithm for association rules in data mining algorithms.

2.4. Data mining interaction

In the interaction process, the communication between the client and the server is performed to complete frequent pattern mining, the main task of the client is to send the original encrypted data to the server, and send the ciphertext, candidate items set and the shadow item set of support degree to the server. While the main function of server is to count up the candidate items set of support degree, and resend the support degree in form of ciphertext to the client. The intercommunication is shown in Figure 4:

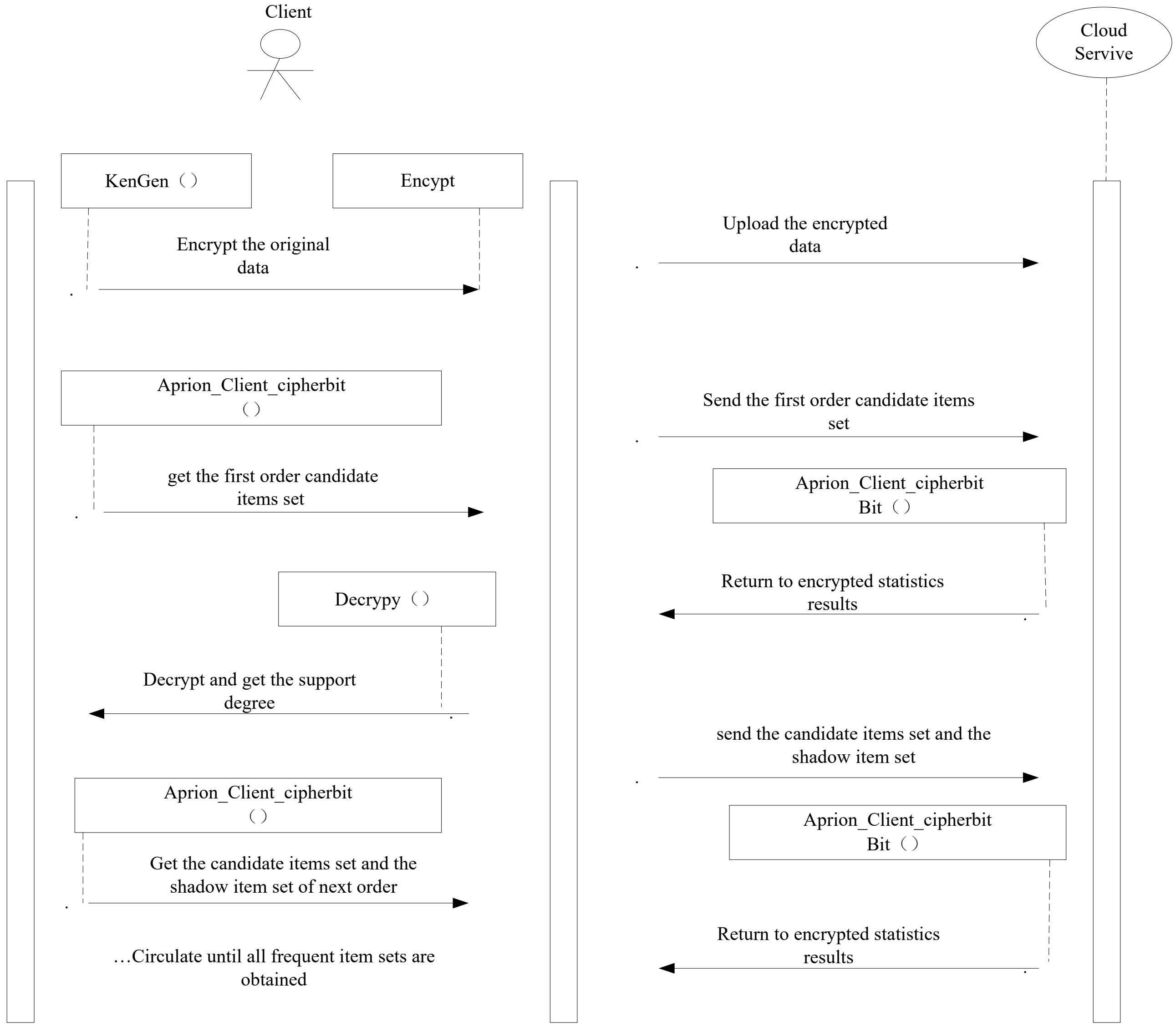

Client
Cloud Servive
KenGen（）
Encypt
Encrypt the original data
Upload the encrypted data
Aprion_Client_cipherbit（）
Send the first order candidate items set
get the first order candidate items set
Aprion_Client_cipherbit Bit（）
Decrypy（）
Return to encrypted statistics results
Decrypt and get the support degree
send the candidate items set and the shadow item set
Aprion_Client_cipherbit（）
Aprion_Client_cipherbit Bit（）
Get the candidate items set and the shadow item set of next order
Return to encrypted statistics results
…Circulate until all frequent item sets are obtained

Figure 4. Intercommunication

To ensure the the safety of characteristic value and improve the query performance of encrypted data, there are three steps of rank, flattening and hash for the extraction of characteristic value, after the extraction, the corresponding characteristic value of each character can be obtained. The main function of rank and flattening is to distribute the characters uniformly in various barrels according to their frequency of occurrence to ensure that the extracted characteristic values will not exhibit large bias, and the main function of the hashing process is to disrupt the distribution sequence of the data [15]. Through the above three processes, it is difficult for attackers to obtain sensitive information from characteristic values.

2.5. Flattening data mining management

The main idea of flattening is to assign each characters to a certain number of barrels so that the sum of the frequency of the characters contained in each barrel is approximately equal. During the flattening, the relationship between barrels and character is many-to-many, which means that various characters may be included in one barrel, and one characters may appear in various barrels. It’s assuming that the number of barrel is m, sum（f） is the sum of frequency of all characters, f represents the total frequency of distributional characters for one barrel and should meet the following constraints:

$$(fB)=\left[\frac{\mathrm{sum}\ (f)}{m}\right]+\gamma \tag{1}$$

λ is the volatility, whose scale is given by:

$$\left[\frac{\mathrm{sunm}\ (f)}{m}\right]\times 5\% \tag{2}$$

Specifically, the process of flattening is as follows: (1) total frequency:

$$\text{sum}（f）=\sum_{i}^{n}\sum_{j}^{n} f_i f_j = f_{11}+f_{12}+\ldots+f_{nn} \qquad (3)$$

The orderly character string is distributed in order to barrels.

Distribution principles: each barrel is distributed characters in order, only the current barrel is allocated, the next barrel will be allocated until the end of the distribution; when the current barrel has free space, one characters with most frequently is found in the unassigned characters and is assigned to the barrel until there is no space left in the barrel. When the frequency of one characters is greater than the barrel space (fB) or the remaining space of the barrel, the characters is assigned to the barrel, and at the same time, the characters is continuously allocated in the next barrel. After the characters has been flattened, it has been uniformly distributed in each barrel. However, since the characters are assigned to the barrels in the order of the frequency, the attacker can derive the correspondence between the characters and the barrels based on prior knowledge, that is, they know which characters is assigned to the barrels. In order to prevent such attacks, a scramble function S is used to disrupt the correspondence between characters and barrels. Specifically, the scrambling function S should satisfy the following properties:

The scrambling function S hashes the key value to a random value;

If $K_1 \neq K_2$, then $S(K_1) \neq S(K_2)$.

The property (1) can ensure that, after hashing, the sequence of original barrels is disrupted, making it difficult for attacker to analyze the correspondence between characters and barrel numbers. The property (2) ensures that the hashed barrel numbers do not collide, so the sum of frequency of characters contained in the barrels remains unchanged, so that the frequency of the characters in the barrels remains uniform and does not exhibit large bias. In general, functions that have both hashing and no conflicts are unusually rare. According to the multiplication hash function, the above properties of the scrambling function can be exactly satisfied.

It's assuming that w is the size of computer word, it's usually 230 or 1010. A is certain integer constant which has relatively prime with w, let $S(K)=\left(\left(\frac{A}{\omega}K\right)\text{mod}1\right)$, then S(K) is called the multiplication hash function. After the mapping of the multiplicative hash function, S(K) takes a value in the range [0,1]. Figure 5 shows the entire extraction process of sorting, flattening, and scrambling of character string characteristic values.

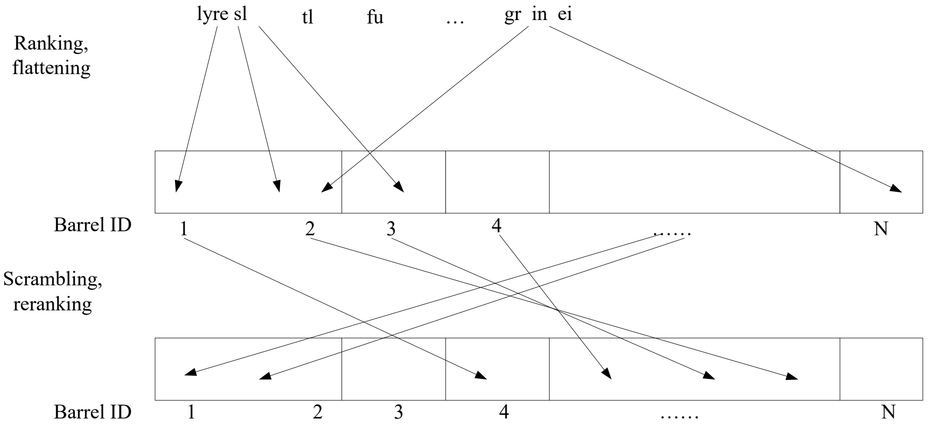


Figure 5. Extraction process of character string characteristic values

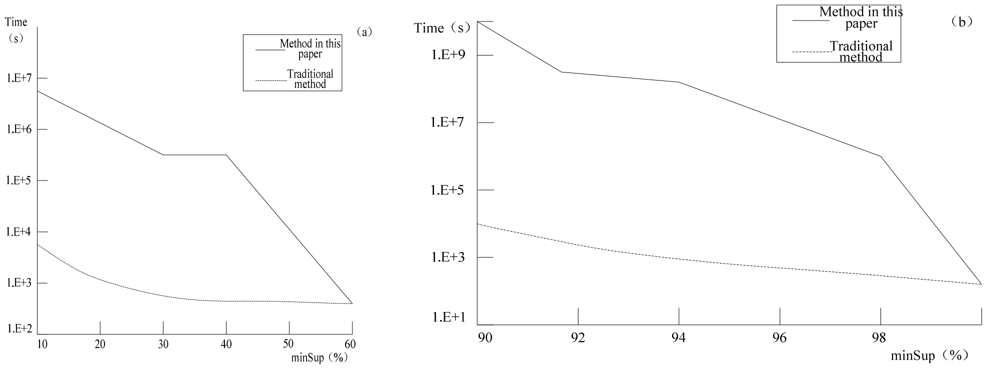


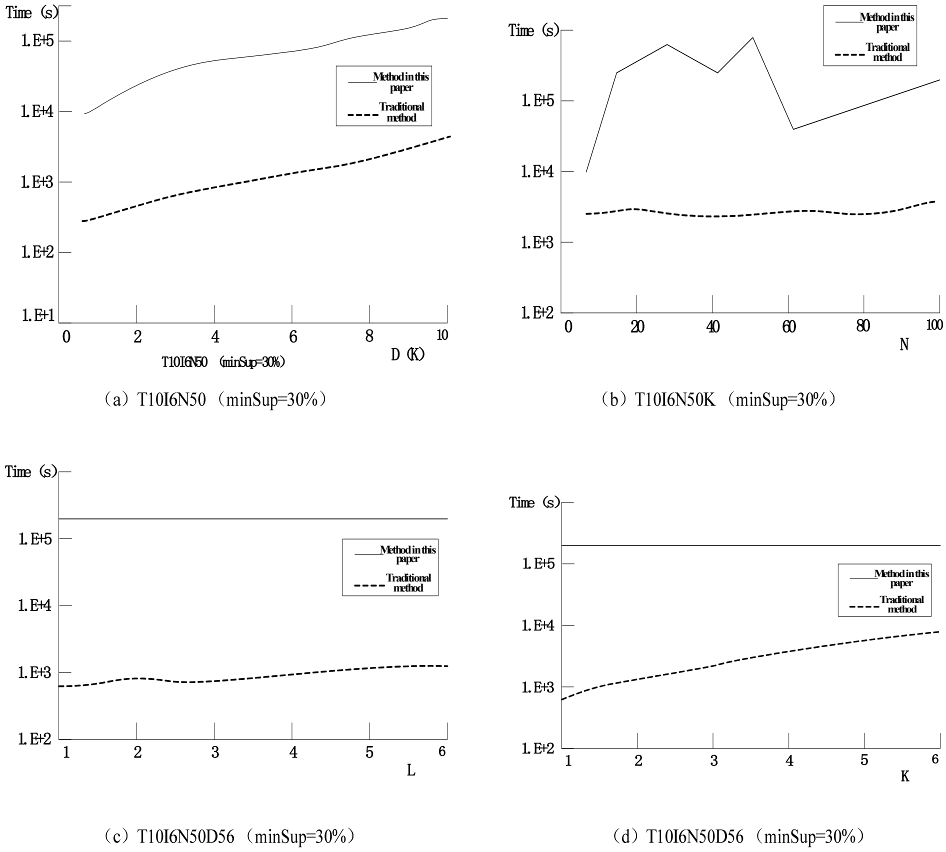
Figure 6. Operating time of two algorithms in various ranges

Example 2 deals with the characters of Example 1, where the parameters are set to: m=16, w=230, and the correspondence between each character and barrel can be obtained.

3. Frequent Pattern Experiment based on Homomorphic Encryption in Cloud Environment

The high-confidence data mining technology of the cloud platform proposed in the efficient data mining technology is a related combined algorithm. Therefore, no similar algorithms have been compared so far. Therefore, a comparison is made between the high-confidence data efficient mining technology of the cloud platform and the traditional mining method algorithm. Two data sets will be used for the experiment. The first one is an artificial data set. The second one is executed by Ubuntu 12.04.4 system. The result is shown in Figure 6:

Firstly, on the T10I6N50D5KL1K dataset and chess dataset, the high-confidence data efficient mining algorithm in cloud platform's network and traditional mining algorithm are evaluated, and minSup, namely, different minimum support degree thresholds are tested. The result is shown in Figure 7 (a)-(b):



（a）T10I6N50（minSup=30%）

（b）T10I6N50K（minSup=30%）

（c）T10I6N50D56（minSup=30%）

（d）T10I6N50D56（minSup=30%）

Figure 7. Data testing

Figure 7 (a) shows the results of testing on the T10I6N50D5KL1K data set. As it can be seen from the above figure: the efficiency value of traditional mining algorithm is higher than that of efficient mining technology of highly confidential data in cloud platform network by 3 orders of magnitude. According to the test of Chess dataset in Figure 7(b), it can be seen that the efficiency value of the traditional mining algorithm is higher than that of the ItFFP algorithm by 5 orders of magnitude. Then in the artificial data set T 1 OI6NSODSkL 1 k, performance tests were performed for the two algorithms under different data attributes and privacy protection requirements, and in each test, minSup=30%. Among them, Transactions is the total number of transaction records, and Distinct Items are the total number of different items included in the test. Firstly, Figure 7 (a) is a test result in which the transaction record number transactions (i.e., the size D of the data set) is from 1,000 (D 1 k) to 10,000 (D 1 Ok). It can be seen that the computing time of traditional mining methods and high-confidence data efficient mining algorithm in cloud platform network is linearly upward with the growth of the database size. Secondly, Figure 7 (b) is the test result of the number of different items Distinct Items (N) from 10 (N10) to

100 (N100). It can be seen that the operating time of the traditional mining algorithm does not appear highly correlated with the change of this property. However, the operating time of high-confidence data efficient mining algorithm in cloud platform's network has two fluctuation characteristics in this data test. That is, the computing time of each iteration of the ItFFP algorithm changes to a positive correlation with the number of candidate item sets. However, it is inversely related to the number of iterative executions in the high-confidence data efficient mining algorithm in cloud platform network. Thirdly, Figure 7 (c) is the test result of the number of potentially frequent patterns (L) from 1000 (Llk) to 6000 (L6K). The experimental test results show that there is a slight difference between the operation time of the traditional mining method and FFP algorithm on this attribute with the increase in attributes. Finally, Figure 7 (d) is the experimental result of the pattern anonymity parameter k setting from 1 to 6. Because in the high-confidence data efficient mining algorithm in cloud platform network does not use the corresponding mode anonymity privacy protection measures, so this attribute does not affect its operation time. It can be seen from the figure that with the increase of the parameter k, the operation time of the traditional mining algorithm grows linearly.

4. Conclusion

In cloud era, big data has attracted more and more attention. People have found that there are many important information behind huge data. Data mining is a kind of in-depth data analysis method. It can perform complex analysis and modeling of a large amount of data and discover various regular and useful information. However, the misuse and abuse of data mining in the cloud environment may lead to the leakage of user data, especially sensitive information. Personal privacy and information security issues in data mining are particularly concerned. Therefore, the research on high-confidence data efficient mining technology in cloud platform's network is performed to find an effective encryption method to deal with character data, which not only can guarantee the security of data, but will not have great impact on the query performance.

5. Acknowledgment

Lanzhou science and technology project project(2017-4-10)

References

[1] Shi Fangxia. Optimized Simulation of High Secret Data Separation and Destruction in Cloud Environment[J]. Computer Simulation, 2017, 34(4):319-322.

[2] Tian Hongliang, Zhang Yong, Li Chao. A Survey of Confidentiality Protection for Cloud Databases[J]. Chinese Journal of Computers, 2017(10):2245-2270.

[3] Li Na, Yu Shengwei. High efficiency mining method of multi-server multi-partition data in cloud computing environment[J]. Modern Electronics Technique, 2017,40(10):43-45.

[4] Zhang Hao, Huang Tao.A Privacy-Preserving Bucket Partition Mechanism in Cloud[J].Chinese Journal of Computers, 2016, 39(2):429-440.

[5] Zeng Shan, Chen Gang, Qi Fazhi. Research on the High Performance Elastic NetworkofCloud Data Center[J].Computer Engineering and Applications, 2017, 34(2):429-440.

[6] Shen Changhong, Zhang Bo, Zeng Zichuan. Cloud data confidentiality protection and integrity verification scheme[J].Journal of Computer Applications, 2016, 36(s2):54-56.

[7] Dang Hongen, Zhao Erping, Sun Haixia. Real-Time Mining Method Simulation of Non-Significant Feature Data under Cloud Computing[J]. Computer Simulation, 2017, 34(7):203-206.

[8] Sun Xu, Wen Mi, Zhang Xu. A Scheme for Dynamic Data Integrity Verification in Smart Grid[J].Computer Engineering, 2017, 43(8):38-43.

[9] Ren Jingsi, Wang Jinlin, Chen Xiao. A method for ensuring data confidentiality in cloud storage[J]. Computer Engineering and Science, 2016, 38(12):2402-2408.

[10] Huang Meidong, Xie Weixin, Zhang Peng. Research on Similar Keyword Search over Encrypted Data in Cloud[J].Journal of Signal Processing, 2017, 33(4):472-479.

[11] Zheng Zhiheng, Zhang Minqing, Dai Xiaoming. Access control with high efficiency scheme for cloud storage based on proxy re-encryption[J]. Application of Electronic Technique, 2016, 42(11):99-101.

[12] Kong Yan, Zhao Shuaibing, Liu LRuolin. Multi-cloud storage system based on Android platform[J].Journal of Computer Applications, 2017, 37(a01):39-44.

[13] Wang Kaixuan, Li Yuxi, Zhou Fucai. Multi-Keyword Fuzzy Search over Encrypted Data[J].Journal of Computer Research and Development, 2017, 54(2):348-360.

[14] Lei Lei, Cai Quanwei, Jing Jiwu. Enforcing Access Controls on Encrypted Cloud Storage with Policy Hiding[J].Journal of Software, 2016, 27(6):1432-1450.

[15] Shi Jiaoli, Huang Chuanhe, Wang Jing. Multi-user collaborative access control scheme in cloud storage[J]. Journal on Communications, 2016, 37(1):88-99.