Differential Privacy-Based Data Mining in Distributed Scenarios Using Decision Trees

Lanqin Wang, Yanmei Lv*
School of Information Engineering, Heilongjiang Polytechnic, Harbin 150070, China
E-mail: w18686761200@163.com
*Corresponding author

Keywords: distributed scene, differential privacy, data mining, decision tree, data classification

Received: September 26, 2024

In the Internet era, data mining is an important means to seize users. However, data exist on different platforms, which are incompatible with each other, and user privacy is easily leaked when mining data. To address this issue, a distributed data mining method based on differential privacy is proposed. The method aggregates frequent itemset data from the top m items of branch nodes through a central node. Noise is added using the Laplace mechanism. The decision tree algorithm is used as a data classification method to set privacy budgets, optimize count queries, and perform importance attribute filtering. The experimental results showed that the maximum data mining accuracy of the improved algorithm was 0.72, which was an average improvement of 0.1 compared with other algorithms. When facing more complex datasets, the decrease in accuracy was relatively small. The minimum relative error of the improved algorithm was 0.1, which was an average improvement of 0.115 over other algorithms. The minimum privacy leakage probability was 0.04%, with an average reduction of 0.08%. The improved algorithm had an average improvement of 0.28 in data classification accuracy and an average reduction of 0.06% in privacy leakage probability when classifying data. When the depth of most decision trees was 4, the maximum classification accuracy was 0.8. From this, the improved algorithm can significantly improve the accuracy of data mining and classification, significantly reduce the privacy budget required for data mining and classification, reduce the probability of privacy leakage, and greatly improve the security of user data.

Povzetek: V prispevku opisana metoda temelji na diferencialni zasebnosti in odločitvenih drevesih za porazdeljeno rudarjenje podatkov. Dosega boljšo klasifikacijo, zmanjšuje verjetnost razkritja zasebnosti ter izboljšuje varnost in učinkovitost analize podatkov.

1 Introduction

Internet technology is increasingly penetrating into people's lives. Especially since the 21st century, relying on the popularity of smart phones, mobile Internet has become an integral part of daily life [1-2]. A large amount of data has been generated in the Internet, which contains countless explicit or implicit information. Previously unknown data have huge potential value [3]. Data mining is to find the required data from a large amount of data, search for the patterns contained in these data, and classify these data according to the pattern, obtaining the potential value existing in the data [4]. However, existing data mining often neglects the user privacy, leading to issues such as privacy breaches and online fraud, which seriously endanger the safety of users' lives and property. Amoozad Mahdiraji et al. proposed a hybrid data mining method that combined rule extraction and service operation benchmarks to extract differences and similarities among bank customers. This method used two-step K-means clustering quality analysis and average distance evaluation method to determine the number of clusters. The best-worst method and total area method were used for clustering ranking. The experimental results showed that this method could accurately identify frequent behaviors of customers [5]. Bhuyan et al. proposed a collaborative computing method for data mining based on optimization models to address privacy protection issues in data mining. This method adopted a fuzzy multi-objective optimization model to generate fuzzy constraints based on the optimization privacy requirements. The experimental results showed that this method could meet the personal privacy requirements of different users in the network and had high flexibility [6]. Dhinakaran and Prathap proposed a new fruit fly whale optimization algorithm that combined association rule algorithms to prevent user information from being stolen by attackers. This algorithm used an adaptive k-anonymity method to convert raw data into encrypted data. Then, it was combined with a bio-inspired algorithm to reduce the low performance when processing large datasets. The experimental results showed that the algorithm provided real-time protection for data privacy, with high robustness [7]. Gai et al. proposed a local differential privacy protection aggregation scheme based on random response to reduce the risk of privacy leakage during power grid data collection. This scheme introduced a random response function based on local differences to dynamically aggregate power grid data. The experimental results showed that this scheme could effectively protect

users' personal privacy, while having lower data transmission and computing costs [8].

Zhao et al. proposed a label differential privacy frequency prediction method for item set local differential privacy in order to shorten the balanced variance and bias of data mining. This method introduced Hadamard encoding into a set of values and encoded the items as fixed vectors, applying perturbations to the vectors. The frequency prediction based on padding sampling and the frequency prediction based on Hadamard transform were combined. The experimental results showed that this method could obtain accurate frequency item sets and their frequencies, and the calculation speed was greatly improved [9]. Singh and Gupta proposed a new K-anonymity model based on differential privacy to improve privacy and security during data sharing. This model divided data into sensitive and non-sensitive categories. The differential privacy and machine learning method were used to perform various tasks and specified multi-party communication protocols. The experiment showed that the model outperformed other algorithms in accuracy, F1 score, and recall, with improvements of 16%, 12%, and 11%, respectively [10]. Lin et al. proposed a privacy protection learning framework based on graph neural networks to improve the privacy protection ability of network graph learning. This framework adopted edge local differential privacy and utilized the common characteristics of real-world graphs to calibrate the noise introduced from dispersed graphs. Experiments showed that this framework could effectively protect node feature privacy and edge privacy, improving the generalization ability of neural networks [11]. Wang et al. proposed a new local differential privacy mechanism to prevent servers from stealing users' private data. This mechanism adopted a three plane framework to protect cross silo data and used machine learning models for decentralized training of the data. Experiments showed that this mechanism could provide effective privacy data protection and expose user data statistics [12].

In summary, existing research methods have explored issues such as accuracy and user privacy protection in distributed scenarios from multiple perspectives, and have achieved certain results. However, existing methods have been unable to meet the increasing demand for data mining accuracy. Therefore, the study adopts differential privacy to optimize data mining and classification methods, innovatively dividing different platforms into multiple nodes for data mining. The decision tree method is used to classify data and optimize counting query results. The optimization method aims to improve the accuracy of data mining and classification, and reduce the probability of user privacy exposure.

Based on relevant research at home and abroad, Table 1 summarizes the themes, main index, methods, and shortcomings of relevant research.

Table 1 Summary of relevant research information

Author	Research theme	Main index	Method	Insufficient
Amoozad Mahdiraji et al. [5]	Customer Information Differential Extraction	Accuracy and precision	K-means and the best-most method	Too long calculation time
Bhuyan et al. [6]	Data Privacy Protection	Privacy exposure probability	Collaborative computing and fuzzy multi-objective optimization	Higher parameter requirements
Dhinakaran et al. [7]	User Information Protection	Computational speed and information exposure probability	k-anonymization method and bio-inspired algorithm	Poor generalization ability
Gai et al. [8]	Privacy Protection	Running cost and risk of data leakage	Stochastic response functions and dynamic aggregation	Multi-stage computation time is too long
Zhao et al. [9]	Reducing Bias in Data Mining	Calculation speed and precision	Local differential privacy and Adama transform	Low overall robustness
Singh et al. [10]	Privacy Security for Data Sharing	Accuracy and recall	K-anonymization models based on differential privacy	Faster model performance degradation when the scene is complex
Lin et al. [11]	Graph Learning Privacy Protection	Risk of privacy leakage	Edge local differential privacy	Longer learning time
This study	Data mining and privacy protection	Data mining accuracy and privacy breach probability	Frequent item set data aggregation and decision tree algorithms	/

In Table 1, the current research has a high ability to protect privacy data when facing attacks, but some methods have certain bottlenecks in terms of calculation speed and detection efficiency in dealing with new threats and more complex scenarios. Therefore, this study uses differential privacy to optimize data mining and classification methods, innovatively dividing different platforms into multiple nodes for data mining. The decision tree method is introduced to classify data and optimize counting query results. The method can effectively improve the accuracy of data mining and classification and reduce the probability of user privacy exposure.

2 Methods and materials

2.1 Distributed frequent item set mining method based on differential privacy

Traditional privacy protection models such as data desensitization, anonymization, and homomorphic encryption mainly protect individual information by masking or blurring some data [13]. However, it is easy to change the original data structure, affecting the value of

extracted information. Meanwhile, its security can only rely on assumptions about the attacker's ability, which cannot effectively define the model's ability to protect privacy.

Therefore, a Distributed Data Frequent Item set Mining-Differential Privacy (DDFIM-DP) is proposed to ensure that inserting and deleting information in the dataset does not affect the data mining results. Noise interference is added to the dataset to protect user privacy while outputting correct results [14]. In distributed scenarios, the diversity of data sources is high, and there is a significant difference in the size of datasets from different sources. Therefore, the DDFIM-DP algorithm selects the largest dataset as the central node and the remaining datasets as branch nodes. The running process of DDFIM-DP algorithm is shown in Figure 1.

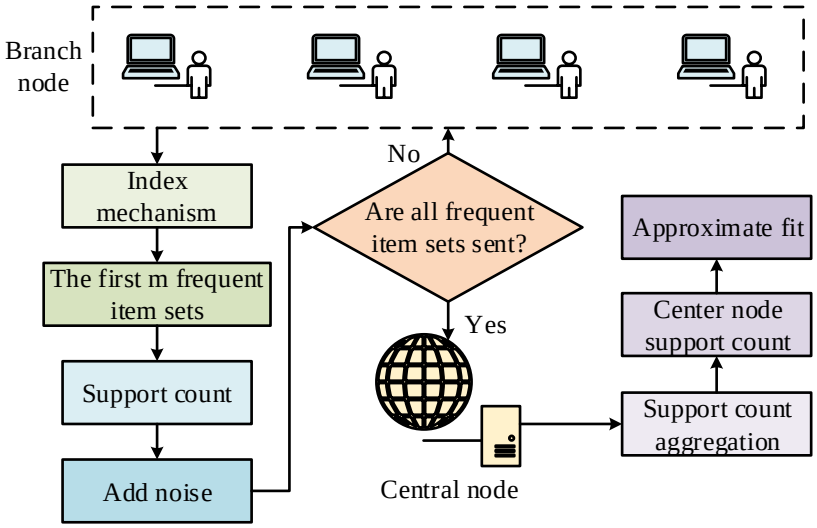

Figure 1: Data mining process of DDFIM-DP algorithm

In Figure 1, the central node first determines the required top n frequent item sets for each node. Each branch node selects m frequent item sets based on the exponential mechanism, where m needs to be greater than n. The support count of m frequent item sets is calculated. After calculation, noise interference is add to the support count to protect data privacy. Then, the processed support count is sent to the central node. The central node sends support count requests for the first m frequent item sets to the branch nodes again. If there are still unsent ones, the branch nodes send the remaining support counts. The second time adds noise to the grouping to reduce the impact of noise on data mining. The central node performs calculations on the support count values of all nodes and returns a single value. For errors caused by noise interference in the calculated data, the support count values of the central node are used to fit the global support count and improve the data value.

When calculating the frequent item sets of the entire scene, each branch node needs to provide the frequent item sets separately. However, if the support counts of all frequent item sets are calculated, it will significantly reduce the calculation speed and increase the computational cost [15]. At the same time, there may be a frequent item set that ranks in the top n items in the entire scene, but ranks less than n on branch nodes. Therefore, the top m frequent item sets sent by branch nodes need to have m greater than n to further improve the accuracy of data mining. When protecting data privacy, association rule mining algorithm is used to extract the set of frequent item sets C1 with support greater than the minimum support threshold in each node. Then, exponential mechanism is used to select the top m frequent item sets. Laplacian mechanism is used to add noise interference to reduce the privacy leakage. The exponential mechanism uses the support count in set C1 as the availability function. The output probability of a frequent item set in set C1 is shown in equation (1) [16].

$$P_i = \frac{\exp\left(\varepsilon \cdot q_i / 2\right)}{\sum_{i \in C} \exp\left(\varepsilon \cdot q_i / 2\right)} \quad (1)$$

In equation (1), P_i represents the output probability of frequent item sets. ε represents the allocated privacy budget. q_i represents the availability function. After calculating the support counts of each set, the Laplace mechanism is also used to add noise interference to them. The core of the Laplace mechanism is to improve data security and achieve privacy protection by adding noise obtained using Laplace distribution to the query results. The Laplace mechanism requires choosing the right privacy budget and sensitivity to make the data as usable as possible while protecting privacy. Adding noise will affect the size and order of the support count values, directly affecting the aggregation of count values at the central node. Therefore, it is necessary to use post-processing method to reorder the count values. The quadratic programming calculation is shown in equation (2).

$$\begin{cases} \min \sum_{i=1}^{n} \left\| Sc_i^{''} - Sc_i^{'} \right\|_2 \\ s.t. Sc_1^{''} \geq \ldots \geq Sc_n^{''} \end{cases} \quad (2)$$

In equation (2), $Sc_i^{'}$ represents the support count after adding i -th noise. $Sc_i^{''}$ represents the i -th support count after post-processing. Although the post-processing rearranges the order, it reduces the gap between the support counts and lowers the availability of the count value. Therefore, it is necessary to segment the item set. Each segment is individually post-processed. The noise count value with the smallest support change in each segment is used as the constraint condition to improve the difference between the support counts [17-18]. The frequent item set generation process for each branch node is shown in Figure 2.

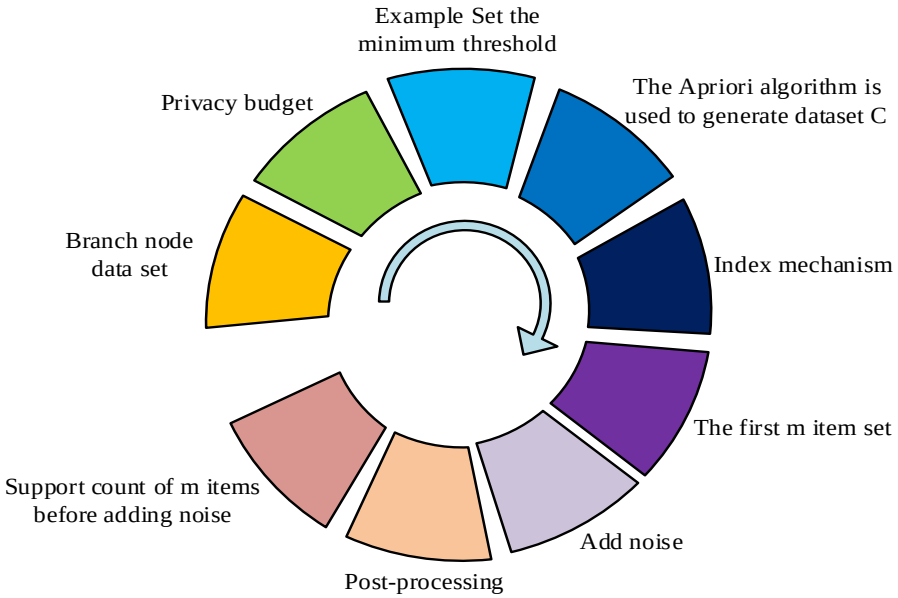

Figure 2: Process of generating frequent item sets for branch nodes

In Figure 2, all frequent item sets for each branch node are input, and the privacy budget values are set. The Apriori algorithm is used to generate frequent item set C1 for each node. The exponential mechanism is used to select the top m frequent item sets, the support count of item sets is calculated, and noise interference is added to the count result. The last is to perform post-processing on the count. In order to avoid frequent item sets with the top n global terms ranking lower than m in branch nodes, resulting in item set omissions and reducing data mining accuracy, it is necessary to supplement the item set support count, that is, to mine candidate frequent item sets. The mining process is shown in Figure 3.

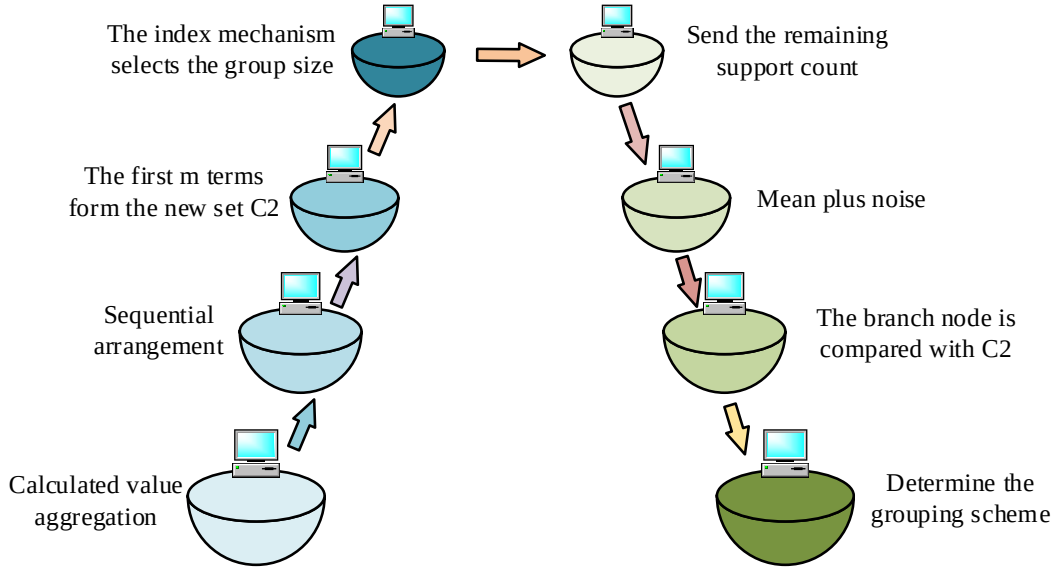

Figure 3: Mining of candidate frequent item sets

In Figure 3, the aggregated support counts of each branch node and the central node are arranged. The frequent item sets of the top m items are selected to form set C2. The central node sends set C2 to each branch node. The branch node is compared with the set C2. C2 contains the support count of frequent item sets that it has sent before. When adding noise, in order to improve the accuracy of data mining, an exponential mechanism is used for grouping and adding noise. It divides adjacent item sets into groups with the same number of item sets in each group. The number of item sets in each group is shown in equation (3).

$$q(S,t)=-\sum_{g\in G_t}\sum_{item\in g}\left(\left|Sc_i-avg_g\right|+\frac{|G_t|}{\varepsilon}\right) \quad (3)$$

In equation (3), q represents the availability function. S represents the existing item sets of support count terms. t represents the number of item sets in each group. Sc_i represents the support count before adding noise. avg_g represents the average support count in group g. G_t represents the grouping scheme. ε represents the privacy budget allocated. In the central node, due to the large amount of data, it is necessary to approximate the data distribution of the central node to conform to the global data distribution. This data distribution is used to optimize all support counts and improve data mining accuracy. The grouping effect of the central node is shown in equation (4).

$$F(Sz,\alpha)=\sum_{b\in group(\alpha)}\mathrm{var}(i) \quad (4)$$

In equation (4), Sz represents the frequent item set of the top m items in the central node. α is the difference between the support count of existing item sets and the support count of the newly added item sets. $group(\alpha)$ represents the grouping scheme for item sets with a threshold of α. $\mathrm{var}(i)$ represents the variance of the frequent item set count value in group i.

2.2 Data classification based on differential privacy and decision tree method

After completing the data mining work on various platforms, it is necessary to classify the mined data information and determine the optimal classification method to ensure the usability of subsequent data query results. To address this issue, the research adopts the Distributed Decision Tree Algorithm-Differential privacy (DDTA-DP). The algorithm calculates and sends information from third-party servers where each node is located. Each branch node jointly constructs a decision tree for data classification while ensuring user privacy. The running process of the DDTA-DP algorithm is shown in Figure 4.

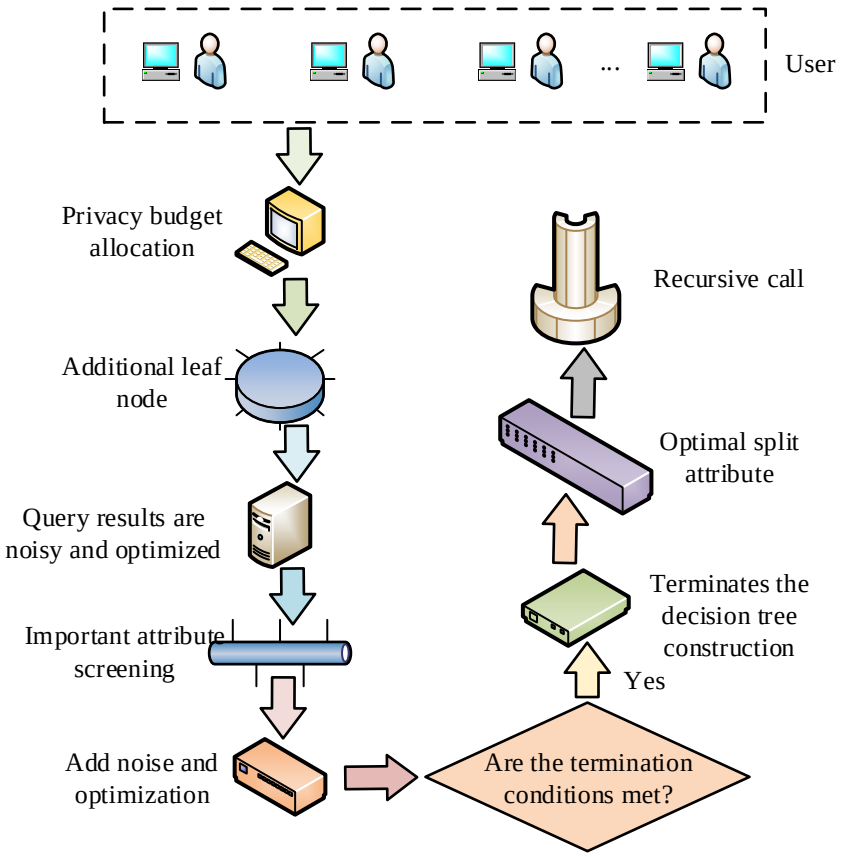

Figure 4: The running process of the DDTA-DP algorithm

In Figure 4, the privacy budget allocated to each node in the decision tree is first set, and a maximum tree depth threshold is set. When the threshold is reached, the decision tree begins to construct leaf nodes. The result of the first count query is added with noise and optimized to support count values. Based on the optimized calculation values, the importance of attributes is determined, and the more important attributes are extracted. The query results of important attributes are optimized by adding noise and count values again. Whether the query results meet the criteria for stopping the decision tree leaf nodes is judged. If it satisfied, the decision tree construction stops. The global gain of the data is calculated to determine the optimal classification parameters. Finally, based on recursive thinking, the DDTA-DP algorithm is used to construct the decision tree. The decision tree construction process in the DDTA-DP algorithm is the process of counting and querying all data. There are two places where user privacy is easily leaked during the query process. Therefore, the privacy budget is divided into two parts, one for the first counting query and another for the important attribute query.

When performing node count queries, the size of the query results for each data is different. The small result indicates that noise has a greater impact on it and requires more privacy budget. In the DDTA-DP algorithm, an adaptive privacy budget allocation method is proposed, where deeper nodes in the decision tree allocate more budget. The calculation is shown in equation (5) [19].

$$K = \sum_{de=1}^{\max_{de}} (de) = \frac{(\max_{de}+1)\cdot \max_{de}}{2} \quad (5)$$

In equation (5), K represents the number of privacy budget allocations. The number of layers where the node is located is the same as the number of privacy shares it receives. $\max_{de}$ represents the maximum decision tree depth. de represents the number of layers where the node is located. In the de-layer, the privacy budget for the first count query score is shown in equation (6).

$$\varepsilon_{1(de)} = \frac{2\varepsilon_1 \cdot de}{(1+\max_{de})\cdot \max_{de}} \quad (6)$$

In equation (6), ε_1 represents the privacy budget allocated to all nodes during the first count query. In the de-layer nodes, the privacy budget count allocated to important attribute count queries is calculated, as shown in equation (7).

$$\varepsilon_{2(de)} = \frac{2\varepsilon_2 \cdot de}{(1+\max_{de})\cdot \max_{de}} \quad (7)$$

In equation (7), ε_2 represents the privacy budget of all node important attribute count queries. Adding noise to count queries can improve data security, but it can also reduce query accuracy. Therefore, it is necessary to perform noise correction on the query results. The optimized algorithm running process is shown in Figure 5.

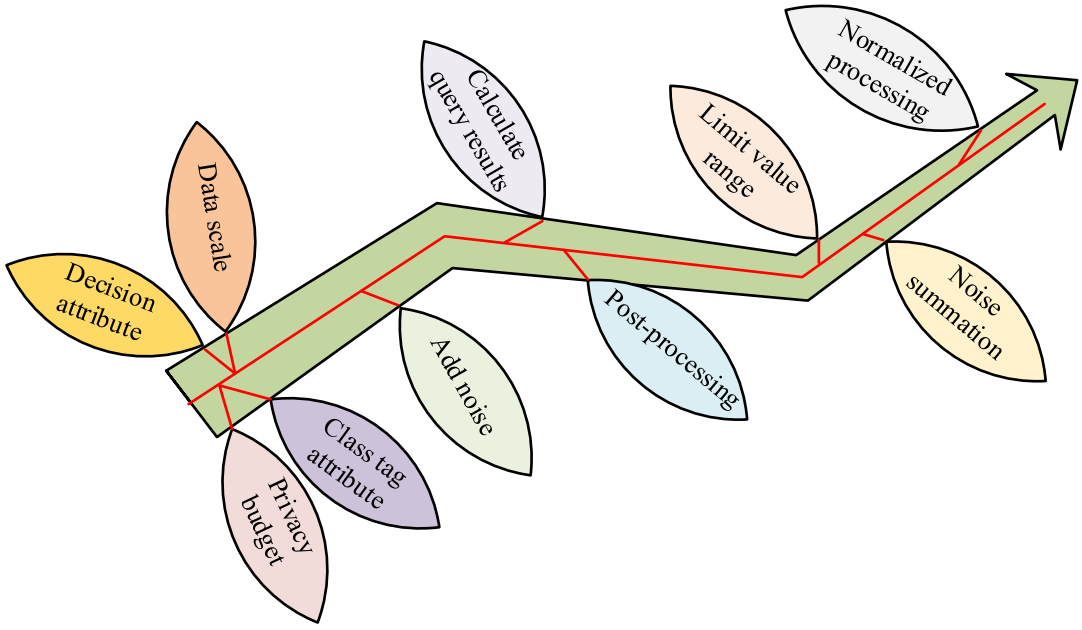

Figure 5: Noise optimization of count query results

In Figure 5, firstly, the decision attribute, scale, privacy budget, and other parameters of the dataset are input. The query results after adding noise are judge to find which constraint condition is satisfied. The query results that meet the constraint condition is post-processed, and the value range of the query results is limited. Next, the noise that satisfies the constraint conditions is summed up. The sum of noise values is normalized and scaled proportionally to correct the query results. Finally, the optimized query results are output [20]. According to the constraint conditions, it is determined that the count query result satisfies equation (8).

$$Co_{ci} = \sum_{a \in A_1} Co_{a,ci} = ... = \sum_{a \in A_d} Co_{a,ci} \quad (8)$$

In equation (8), Co_{ci} represents the number of classes ci in the dataset. $Co_{a,ci}$ represents the number of classes $Co_{a,ci}$ in the query results when a certain attribute value is a. Another constraint satisfied by the decision tree is shown in equation (9).

$$Si_D = \sum_{ci \in class} Co_{ci} \quad (9)$$

In equation (9), Si_D represents the total number of records in the dataset. $class$ represents the class label attribute in the dataset. To improve the accuracy of counting query results, it is necessary to find the value of total noise that is closest to Co_{ci}. The total noise value needs to satisfy equation (10).

$$\min_{A \in \Omega} \left| \sum_{a \in A} Co'_{a,ci} - Co_{ci} \right| \quad (10)$$

In equation (10), A represents a certain attribute. Ω represents the set of decision attributes. $Co'_{a,ci}$ represents the number of ci classes when attribute A has a value of a. The sum of noise values is normalized, as shown in equation (11) [21].

$$Co''_{ci} = \frac{Si_D}{\sum_{C \in class} Co'_{ci}} \cdot Co'_{ci} \quad (11)$$

In equation (11), Co'_{ci} represents the sum of noise values. Next, the normalized values of all class labels are subjected to interval constraints, as shown in equation (12).

$$Co''_{a,ci} = \frac{Co''_{ci}}{\sum_{b \in A} Co'_{b,ci}} \cdot Co'_{a,ci} \quad (12)$$

In equation (12), b represents the value of attribute A at this time. When using decision trees to classify data, the SelectAttrs algorithm is used to determine the importance of each attribute, eliminate unimportant attributes, reduce query computation load, and improve computation speed. The operation process of the SelectAttrs algorithm is shown in Figure 6.

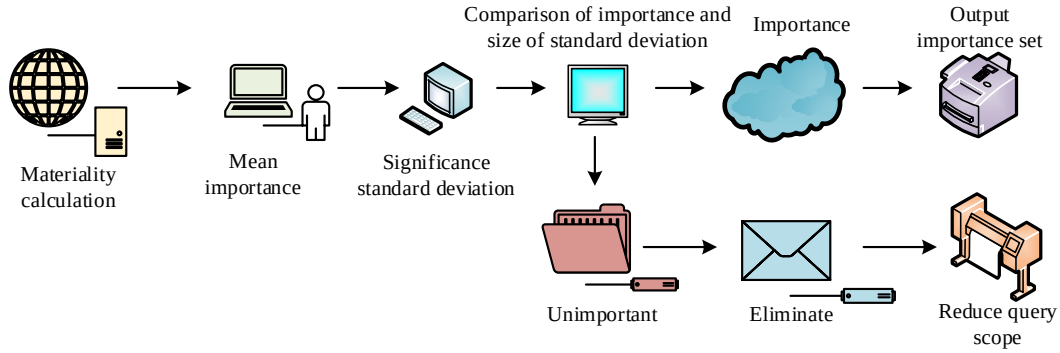

Figure 6: Operation flow of SelectAttrs algorithm

In Figure 6, firstly, the importance level of all attributes is calculated. Then, the average and standard deviation of the importance level of all attributes are calculated. The relationship between the importance level and standard deviation of each attribute is compared, dividing the attributes into important and non-important categories. In the subsequent calculation, non-important attributes are removed to reduce the counting query range. Finally, the set of important attributes is output. In order to reduce the bias caused by importance classification, the correction coefficient is used to adjust attributes, as shown in equation (13).

$$\begin{cases} f(x) = \sin\dfrac{1}{3}, x < 4 \\ f(x) = \sin\dfrac{1}{x}, x \geq 4 \end{cases} \quad (13)$$

In equation (13), x represents the number of attribute values. The attribute importance is shown in equation (14) [22].

$$\mathrm{Im}(A)=\sum_{i=1}^{N}\frac{|D_i|}{\sum_{j=1}^{N}|D_j|}\cdot f(|A|)\cdot In(D_i,A) \quad (14)$$

In equation (14), D_i represents a dataset with horizontal distribution. $\frac{|D_i|}{\sum_{j=1}^{N}|D_j|}$ represents the proportion of D_i in all datasets. $In(D_i,A)$ represents the information gain of A in D_i. The important attribute satisfies equation (15).

$$\mathrm{Im}(A)>\mu-\sigma \quad (15)$$

In equation (15), μ represents the average value of the set of known important attributes. σ represents the standard deviation of the set.

3 Results

3.1 Experimental analysis of frequent itemset mining method for distributed data

The experiment uses three publicly available datasets, Kaggle, UCI KDD, and Accidents, with average record lengths of 27, 52, and 33.6, respectively. Multiple computers are used to simulate each node for simulation experiments. 30% of the data in each dataset is allocated to the central node, while the remaining data is evenly distributed to each branch node. The comparative algorithms used in the experiment include Apriori, K-means, and DDFIM-DP algorithm without post-processing (Pre DDFIM-DP). The data mining accuracy of different algorithms varies with the privacy budget, as shown in Figure 7.

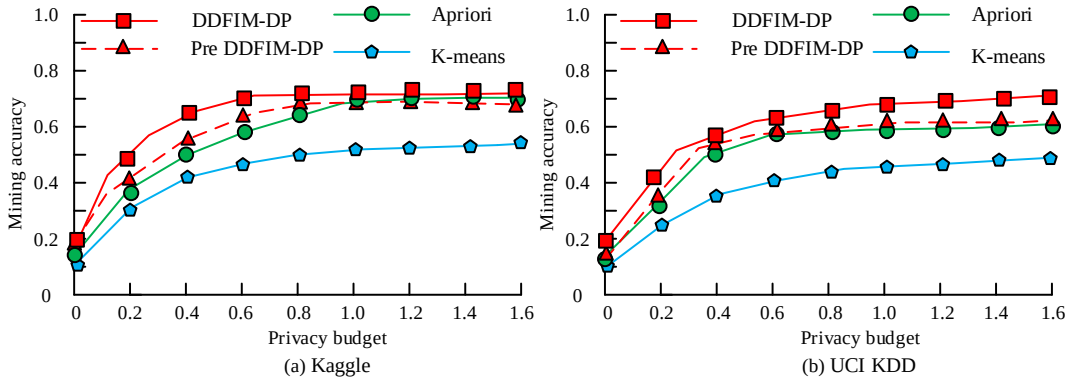

Figure 7: The variation trend of data mining accuracy of different algorithms with privacy budget

In Figure 7 (a), as the privacy budget gradually increased, the data mining accuracy of all algorithms also increased. At the same privacy budget, the mining accuracy of DDFIM-DP algorithm was significantly higher than other algorithms. The DDFIM-DP algorithm achieved maximum accuracy with a privacy budget of 0.6, and its maximum accuracy was 0.04, 0.07, and 0.21 higher than other algorithms, respectively. In Figure 7 (b), the DDFIM-DP algorithm achieved a significant increase in privacy budget for maximum accuracy. Because the average record length in the UCI KDD dataset is about twice that of Kaggle, the data is divided into more segments. The maximum accuracy of the DDFIM-DP algorithm was 0.11, 0.13, and 0.25 higher than other algorithms, respectively. The accuracy of the K-means algorithm was the lowest. Because the K-means algorithm uses random responses to protect privacy, data are severely affected by noise interference, resulting in a significant decrease in usability. The relative error of different algorithms varies with privacy budget, as shown in Figure 8.

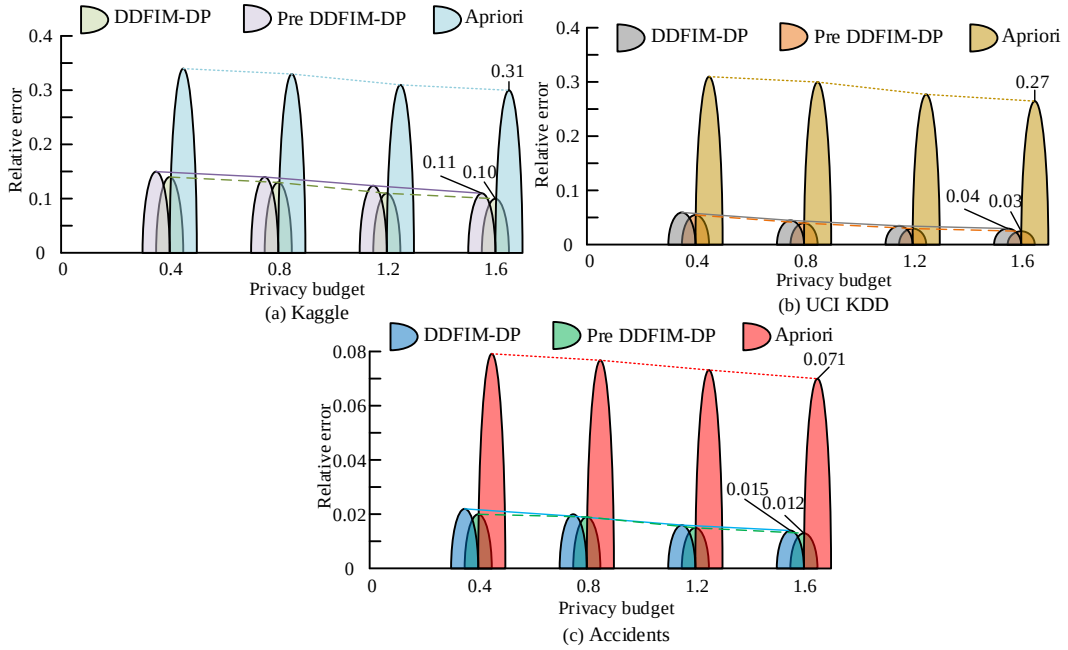

Figure 8: Comparison of relative errors of different algorithms

In Figure 8 (a), the relative error of all algorithms gradually decreased with the increase of privacy budget. At the same privacy budget, the relative error of the DDFIM-DP algorithm was the lowest, averaging 0.02 and 0.21 lower than the Pre DDFIM-DP and Apriori algorithms, respectively. In Figure 8 (b), the relative error of the DDFIM-DP algorithm was on average 0.01 and 0.26 lower than that of the Pre DDFIM-DP and Apriori algorithms. In Figure 8 (c), the relative error of the DDFIM-DP algorithm was on average 0.006 and 0.075 lower than that of the Pre DDFIM-DP and Apriori algorithms. The relative error of the DDFIM-DP algorithm is lower, because it performs support optimization operations to reduce the impact of noise. The relative error of the Apriori algorithm is significantly higher, because it requires high computing power and cannot effectively eliminate noise interference when the number of terminals is limited. The comparison of privacy budget and privacy leakage probability for each algorithm in different datasets is shown in Figure 9.

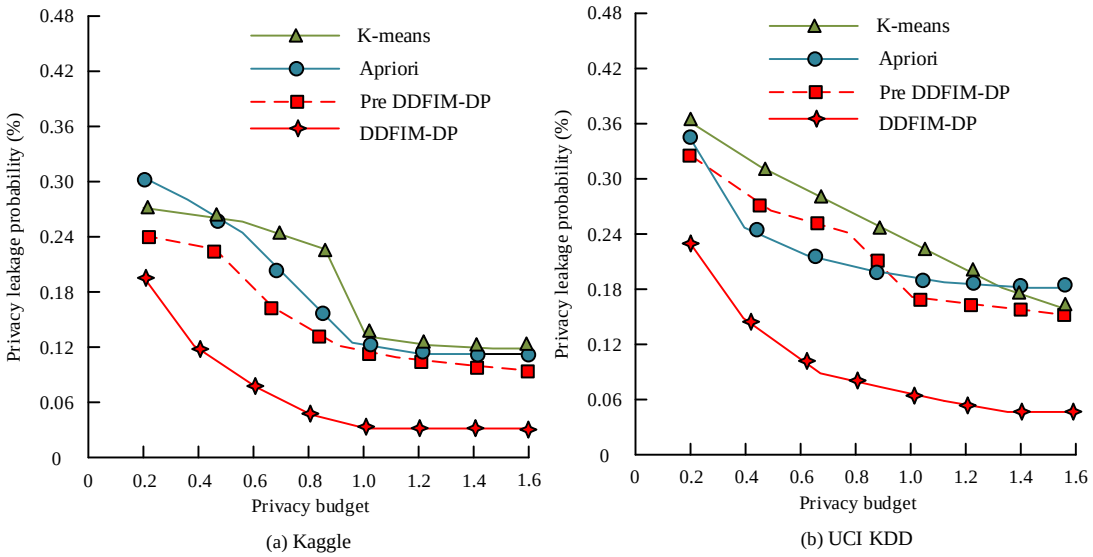

Figure 9: The privacy budget and privacy leakage probability of each algorithm in different datasets

In Figure 9 (a), the DDFIM-DP algorithm had the lowest privacy leakage probability, and the required privacy budget to reach the lowest point was also low. The privacy leakage probability was 0.04% at 1.0 privacy budget, which was 0.07%, 0.09%, and 0.10% lower than the other three algorithms, respectively. In Figure 9 (b), the privacy budget required for the DDFIM-DP algorithm to achieve the minimum privacy leakage probability was 0.2 higher than before. Because the average record length of the UCI KDD dataset is longer, it

requires more noise to be added, and the cost of removing noise at the central node is also higher. The minimum privacy leakage probability of DDFIM-DP algorithm was 0.11%, 0.16%, and 0.13% lower than the other three algorithms, respectively.

3.2 Experimental analysis of data classification based on differential privacy and decision tree method

The simulation experiment adopts the DDTA-DP algorithm, logistic regression algorithm (Logistic), ID3 algorithm, and Privacy Protected ID3 algorithm (PP-ID3) for comparative analysis. The experimental dataset consists of two open datasets, Iris and Wine, with categories of 2, 5, and 7, respectively. The experiment uses multiple computers to simulate each node for simulation experiments. 30% of the data in each dataset is allocated to the central node, and the remaining data is evenly distributed to each branch node. The maximum tree depth for the experiment is set to 5. Each experiment is conducted 20 times. The final results are averaged. The data classification accuracy of different algorithms is shown in Figure 10.

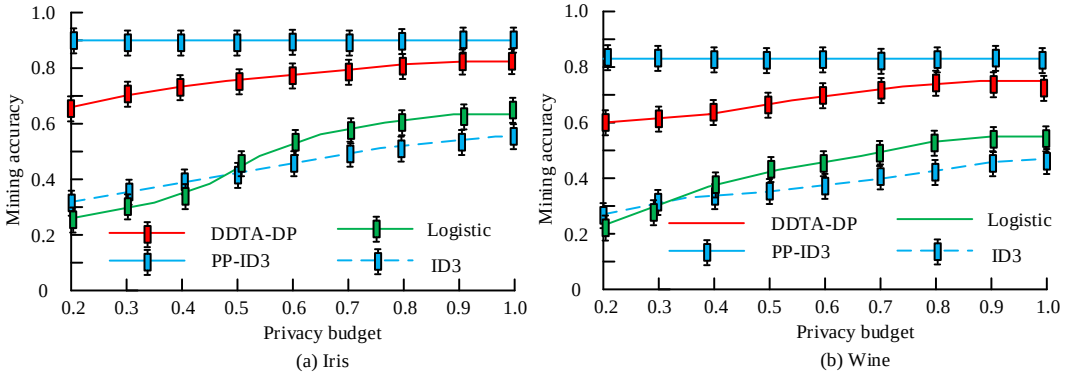

Figure 10: The variation trend of data classification accuracy of different algorithms with privacy budget

In Figure 10 (a), the data classification accuracy of the DDTA-DP algorithm increased with the increase of privacy budget, reaching a maximum of 0.82 at a privacy budget of 0.9, which was close to the ID3 algorithm without privacy protection. It was 0.33 and 0.24 higher than other two algorithms, respectively. In Figure 10 (b), the data classification accuracy of the DDTA-DP algorithm was the highest at 0.74, which was 0.31 and 0.22 higher than the maximum accuracy of other algorithms, respectively. Because the DDTA-DP algorithm optimizes count queries while filtering important attributes, it can effectively reduce the interference of noise. The relationship between classification accuracy and decision tree depth of different algorithms is shown in Figure 11.

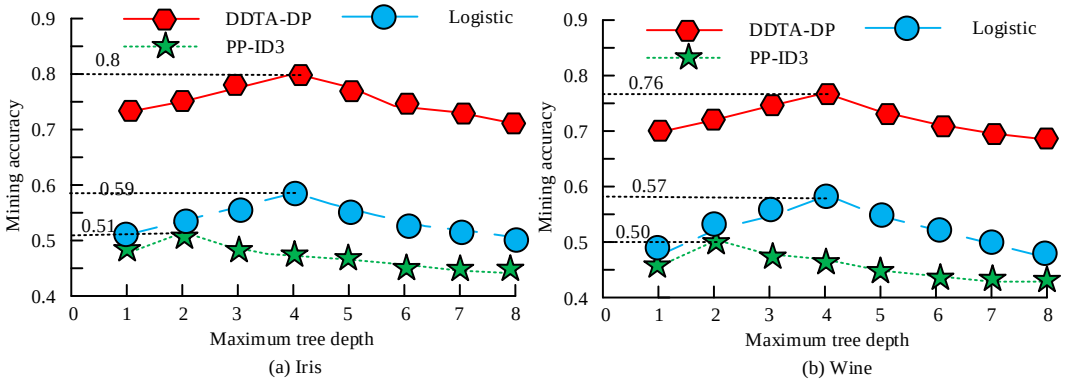

Figure 11: The relationship between classification accuracy and decision tree depth of different algorithms

In Figure 11 (a), the maximum accuracy of the DDTA-DP algorithm reached 0.8 at a maximum tree depth of 4. The maximum accuracy of the logistic regression algorithm also achieved at a tree depth of 4. PP-ID3 showed its maximum accuracy at a tree depth of 2. The maximum accuracy of different algorithms increases first and then decreases. Because when the tree depth is small, the privacy budget is sufficient. However, the decision tree is too low, resulting in insufficient training and affecting classification accuracy. When the tree depth is large, the model is fully trained, but the privacy budget is excessively segmented. The model is greatly affected by noise interference, which also reduces the classification accuracy. In Figure 11 (b), the classification accuracy changes of the three algorithms were the same as before. However, due to the dataset having more attribute categories, the decrease in classification accuracy was relatively small. The

probability of privacy leakage during data classification is shown in Figure 12.

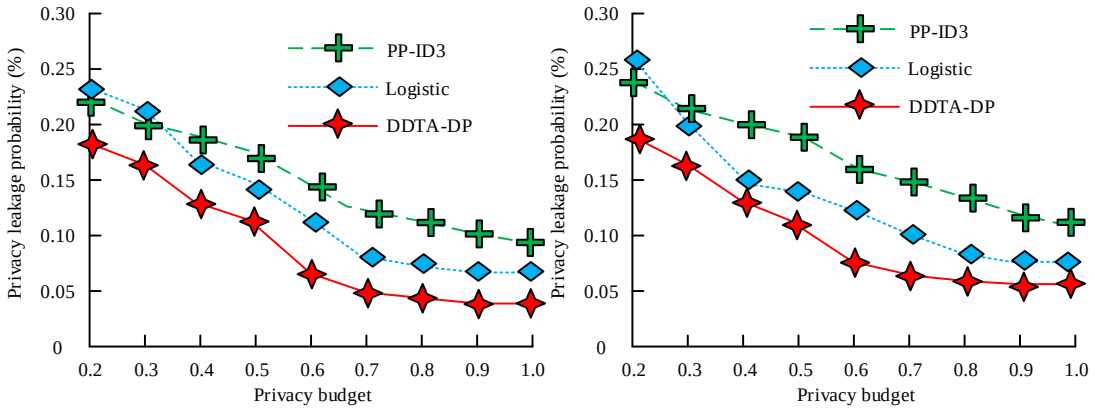

Figure 12: The relationship between privacy leakage probability and privacy budget in data classification

In Figure 12 (a), the DDTA-DP algorithm had the lowest privacy leakage probability, with a minimum leakage probability 0.04 and 0.09 lower than the logistic regression algorithm and PP-ID3 algorithm, respectively. The DDTA-DP algorithm also required less privacy budget to solve the minimum leakage probability. In Figure 12 (b), the DDTA-DP algorithm had the minimum privacy leakage probability, which was 0.03 and 0.10 lower than the logistic regression algorithm and PP-ID3 algorithm, respectively. In order to analyze the computational complexity of DDFIEM-DP algorithm and DDTA-DP algorithm, the time required by the two algorithms to process the same amount of data under the same hardware conditions is measured by experiments. It is compared with the traditional K-means and Apriori algorithm. The time complexity comparison results of the three algorithms are shown in Table 2.

Table 2: Comparison of time complexity of three algorithms

Sample size	Model type	Average processing time (ms)	Standard deviation (ms)	Time complexity $O(n^2)$ evaluation
1000	DDFIEM-DP	18.4	2.1	Lower
	DDTA-DP	22.5	2.9	Lower
	K-means	29.6	3.6	Normal
	Apriori	30.4	3.5	Normal
2000	DDFIEM-DP	72.6	5.2	Lower
	DDTA-DP	114.2	7.7	Lower
	K-means	136.8	9.4	Normal
	Apriori	142.5	8.9	Normal
3000	DDFIEM-DP	169.8	8.3	Lower
	DDTA-DP	254.6	11.9	Normal
	K-means	276.3	13.1	Higher
	Apriori	268.4	12.5	Higher
4000	DDFIEM-DP	315.8	12.3	Lower
	DDTA-DP	396.5	18.5	Normal
	K-means	472.9	20.8	Higher
	Apriori	469.2	20.1	Higher

In Table 2, when the sample size was 1000, 2000, 3000, and 4000, the average processing time of DDFIEM-DP algorithm was 11.2ms, 64.2ms, 106.5ms, and 157.1ms lower than K-means, respectively, while DDTA-DP algorithm was 7.1ms, 22.4ms, and 76.4ms lower, respectively. Both algorithms had lower standard deviation than K-means and Apriori, lower computational complexity, and faster computational efficiency than traditional algorithms. The time complexity of all four methods increased accordingly with the increase of sample size, but the DDFIEM-DP and DDTA-DP algorithms increased more slowly than K-means and Apriori. Decision tree algorithms are able to satisfy the requirements of differential privacy by adding noise to data analysis and modelling without compromising personal privacy. Meanwhile, it can effectively process high-dimensional data and have high classification accuracy while ensuring privacy and security. In large-scale sample computation, decision tree algorithm can effectively reduce prediction time complexity and improve prediction efficiency. Due to excessive noise in deep nodes, further increasing the complexity of the model will reduce prediction accuracy. The study compares the DDTA-DP algorithm with the current state-of-the-art Random Forest (RF) algorithm and Gradient Boosting Decision Tree (GBDT). As shown in Table 3, the results of the experimental statistical test p-values are also analyzed.

Table 3: Comparison of data classification performance of different algorithms

Arithmetic	Classification accuracy	Probability of privacy breach (%)	Running time (ms)	*P*
DDTA-DP	0.82	0.049	72.6	0.004
RF	0.74	0.082	84.9	0.002
GBDT	0.79	0.074	89.2	0.004

In Table 3, the classification accuracy of DDTA-DP algorithm was 0.08 and 0.03 higher than that of RF and GBDT, respectively. The privacy leakage probability was 0.033% and 0.025% lower than that of the two algorithms, and the running time was 12.3ms and 16.6ms lower than that of the two algorithms, respectively. The *P*-value of all three algorithms was less than 0.05, indicating that the experiment complied with statistical principles.

4 Discussion

With the continuous development of mobile Internet, data mining has become an important means to seize users. However, due to the large number of separate platforms for data, there are problems such as user privacy leakage. To address this problem, this study proposes a constant item set data mining based on differential privacy and data classification using decision trees. Through experimental analyses, the good performance of the method in terms of data mining accuracy and user privacy protection is verified. The data mining accuracy of this algorithm gradually increased with the increase of privacy budget, which was higher than Apriori algorithm and K-means algorithm. Due to traditional methods that protect personal information by masking or blurring certain data, this can easily lead to data loss. Moreover, the larger the dataset, the higher the likelihood of data loss. The privacy budget of the DDFIM-DP algorithm is 0.2, and the mining accuracies in the two datasets are 0.49 and 0.42, respectively. Because the smaller the privacy budget, the more noise needs to be added, which seriously reduces the accuracy of the results. On the other hand, DDFIM-DP adds noise to the data and uses a central node to approximate the global support count, which can effectively improve data utilization. The mining accuracy of the Pre DFIM-DP algorithm without post-processing increases slowly, with a privacy budget of 0-0.6 and a growth rate 17.2% lower than that of the DDFIM-DP algorithm. This is because post-processing can effectively improve the accuracy of supporting counting. DDFIM-DP grouping adds noise through an exponential mechanism. The grouping method can effectively reduce noise interference. As the decision tree iterates, the dataset is continuously partitioned. Excessive additional noise may mask the counting query results. Therefore, more privacy budget needs to be allocated to reduce noise interference. The data classification accuracy of the DDTA-DP algorithm increased with the increase of the privacy budget, and approached the convergence state when the privacy budget was 0.9. Increasing the privacy budget can effectively reduce the interference of noise in the deep decision tree on the counting query results and improve the accuracy of data classification. The complexity of the decision tree algorithm is positively correlated with the depth of the tree. The classification accuracy of the DDTA-DP algorithm increased first and then decreased with the increase of the depth of the tree. Therefore, further increasing the complexity of the model does not improve its performance.

5 Conclusion

Aiming at the low accuracy and poor privacy protection effect of traditional algorithms in data mining and classification in distributed scenarios, a frequent item set data mining method based on differential privacy was proposed. The research results showed that the DDFIM-DP algorithm had a maximum accuracy of 0.72, which was 0.04, 0.07, and 0.21 higher than other algorithms, respectively. Faced with datasets with longer average record lengths, the mining accuracy of the DDFIM-DP algorithm decreased even less, which was 0.11, 0.13, and 0.25 higher than other algorithms, respectively, indicating a stronger ability to remove noise interference. Under the same privacy budget, the relative error of DDFIM-DP algorithm was lower, with a minimum value of 0.1, which was 0.02 and 0.21 lower than other algorithms, respectively, because DDFIM-DP optimized support and reduced the impact of noise. The privacy leakage probability of the DDFIM-DP algorithm was close to the lowest value of 0.04% in the 1.0 budget, and the lowest leakage probabilities were 0.07%, 0.09%, and 0.10% lower than other algorithms, respectively. When performing data classification, the accuracy of the DDTA-DP algorithm increased with the increase of privacy budget, with a maximum accuracy of 0.82, which was 0.33 and 0.24 higher than other algorithms, respectively. The classification accuracy of the DDTA-DP algorithm increased first and then decreased with the maximum tree depth of the decision tree. When the tree depth was 4, the maximum classification accuracy of the DDTA-DP algorithm was 0.21 and 0.29 higher than other algorithms, respectively. The privacy leakage probability of the DDTA-DP algorithm was 0.04 and 0.09 lower than that of the logistic regression algorithm and PP-ID3 algorithm, respectively, with less privacy budget. There are still some issues in this study. Information gain cannot accurately measure the contribution of attributes to data classification. Future research will better measure relevant indicators.

References

[1] Zhao Y, Chen J. A survey on differential privacy for unstructured data content. ACM Computing Surveys (CSUR), 2022, 54(10):1-28. DOI: 10.1145/3490237.

[2] Wu H, Chen YF, Wu Z, et al. A decision procedure for string constraints with string/integer conversion and

flat regular constraints. Acta Informatica, 2024, 61(5): 23–52. DOI: 10.1007/s00236-023-00446-4.

[3] Gupta R, Saxena D, Gupta I, Singh AK. Differential and triphase adaptive learning-based privacy-preserving model for medical data in cloud environment. IEEE Networking Letters, 2022, 4(4):217-221. DOI: 10.1109/lnet.2022.3215248.

[4] Ho TT, Tran KD, Huang Y. FedSGDCOVID: Federated SGD COVID-19 detection under local differential privacy using chest X-ray images and symptom information. Sensors, 2022, 22(10):3728-3735. DOI: 10.3390/s22103728.

[5] Amoozad Mahdiraji H, Tavana M, Mahdiani P, Abbasi Kamardi AA. A multi-attribute data mining model for rule extraction and service operations benchmarking. Benchmarking: an international journal, 2022, 29(2):456-495. DOI: 10.1108/bij-03-2021-0127.

[6] Bhuyan HK, Kamila NK, Pani SK. Individual privacy in data mining using fuzzy optimization. Engineering Optimization, 2022, 54(8):1305-1323. DOI: 10.1080/0305215x.2021.1922897.

[7] Dhinakaran D, Prathap PJ. Protection of data privacy from vulnerability using two-fish technique with Apriori algorithm in data mining. The Journal of Supercomputing, 2022, 78(16):17559-17593. DOI: 10.1142/s0218126624501093.

[8] Gai N, Xue K, Zhu B, Yang J, Liu J, He D. An efficient data aggregation scheme with local differential privacy in smart grid. Digital Communications and Networks, 2022, 8(3):333-342. DOI: 10.1109/msn50589.2020.00027.

[9] Zhao D, Zhao SY, Chen H, Liu RX, Li CP, Zhang XY. Hadamard Encoding Based Frequent Itemset Mining under Local Differential Privacy. Journal of Computer Science and Technology, 2023, 38(6):1403-1422. DOI: 10.1007/s11390-023-1346-7.

[10] Singh AK, Gupta R. A privacy-preserving model based on differential approach for sensitive data in cloud environment. Multimedia Tools and Applications, 2022, 81(23):33127-33150. DOI: 10.1007/s11042-021-11751-w.

[11] Lin W, Li B, Wang C. Towards private learning on decentralized graphs with local differential privacy. IEEE Transactions on Information Forensics and Security, 2022, 17(6):2936-2946. DOI: 10.1109/tifs.2022.3198283.

[12] Wang C, Wu X, Liu G, Deng T, Peng K, Wan S. Safeguarding cross-silo federated learning with local differential privacy. Digital Communications and Networks, 2022, 8(4):446-454. DOI: 10.1016/j.dcan.2021.11.006.

[13] Kumar S, Mohbey KK. A review on big data based parallel and distributed approaches of pattern mining. Journal of King Saud University-Computer and Information Sciences, 2022, 34(5):1639-1662. DOI: 10.1016/j.jksuci.2019.09.006.

[14] Sunhare P, Chowdhary RR, Chattopadhyay MK. Internet of things and data mining: An application oriented survey. Journal of King Saud University-Computer and Information Sciences, 2022, 34(6):3569-3590. DOI: 10.1016/j.jksuci.2020.07.002.

[15] Salem R B, Aimeur E, Hage H. A Multi-Party Agent for Privacy Preference Elicitation[C]//Artificial Intelligence and Applications. 2023, 1(2): 98-105. DOI: 10.47852/bonviewaia2202514.

[16] Hewage UH, Sinha R, Naeem MA. Privacy-preserving data (stream) mining techniques and their impact on data mining accuracy: a systematic literature review. Artificial Intelligence Review, 2023, 56(9):10427-10464. DOI: 10.1007/s10462-023-10425-3.

[17] Sarwar T, Seifollahi S, Chan J, Zhang X, Aksakalli V, Hudson I, Verspoor K, Cavedon L. The secondary use of electronic health records for data mining: Data characteristics and challenges. ACM Computing Surveys (CSUR), 2022, 55(2):1-40. DOI: 10.1145/3490234.

[18] G Sharma, A K Kapil. Intrusion Detection and Prevention Framework Using Data Mining Techniques for Financial Sector. Acta Informatica Malaysia. 2021; 5(2): 58-61. DOI: 10.26480/aim.02.2021.58.61.

[19] Chen C, Wu Y, Li J, et al. TBtools-II: A “one for all, all for one” bioinformatics platform for biological big-data mining. Molecular plant, 2023, 16(11):1733-1742. DOI: 10.1016/j.molp.2023.09.010.

[20] Chen C, Zhang Q, Kashani MH, Jun C, Bateni SM, Band SS, Dash SS, Chau KW. Forecast of rainfall distribution based on fixed sliding window long short-term memory. Engineering Applications of Computational Fluid Mechanics, 2022, 16(1):248-261. DOI: 10.1080/19942060.2021.2009374.

[21] Hong Y, Li J, Lin Y, Hu Q, Li X. Trajectory-aware privacy-preserving method with local differential privacy in crowdsourcing. EURASIP Journal on Information Security, 2024, 2024(1):28-36. DOI: 10.1186/s13635-024-00177-0.

[22] Dang TK, Tran-Truong PT. A Pragmatic Privacy-Preserving Deep Learning Framework Satisfying Differential Privacy. SN Computer Science, 2023, 5(1):130-142. DOI: 10.1007/s42979-023-02437-1.