IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 27, 2020, accepted May 8, 2020, date of publication May 18, 2020, date of current version June 1, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2995058

Research on an Ensemble Classification Algorithm Based on Differential Privacy

JUNJIE JIA AND WANYONG QIU
College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China

Corresponding author: Wanyong Qiu (365046920@qq.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61967013, and in part by the Gansu Province Colleges and Universities Innovation Capability Enhancement Project under Grant 2019A-006.

ABSTRACT In the field of information security, privacy protection based on machine learning is currently a hot topic. Combining differential privacy protection with AdaBoost, a machine learning ensemble classification algorithm, this paper proposes a scheme under differential privacy named CART-DPsAdaBoost (CART-Differential privacy structure of AdaBoost). In the process of boosting, the algorithm combines the idea of bagging, and uses a classification and regression tree (CART) stump as the base learner for ensemble learning. Applying feature perturbation, based on a random subspace algorithm, the exponential mechanism is used to select the splitting point for continuous attributes. We use the Gini index to find the optimal binary partitioning point for discrete attributes and add noise according to the Laplace mechanism. Throughout the process, a privacy budget is allocated in order to meet the appropriate differential privacy protection needs for the current application. Unlike similar algorithms, this method does not require discretization during preprocessing of the data. Experimental results with the Census Income, Digit Recognizer, and Adult Data Set show that while protecting private information, the scheme has little impact on classification accuracy and can effectively address large-scale and high-dimensional data classification problems.

INDEX TERMS Privacy protection, differential privacy, machine learning, AdaBoost.

I. INTRODUCTION

The disruptive development of the Internet has brought convenience and rapidity to communication and data sharing, and has promoted the arrival of the era of big data. We leave a lot of footprints on the Internet where various information systems collect and accumulate rich data, which constitutes an important foundation of big data. According to Schonberg, author of The Big Data Era: The Great Revolution in Life, Work, and Thinking, big data can utilize all data, not just a sample from it, and it can focus more on finding and analyzing the relevance of things. However, in bringing convenient services, such as personalized recommendations, the risk of privacy breaches has also increased rapidly. Hence, privacy issues are receiving more attention in machine learning research.

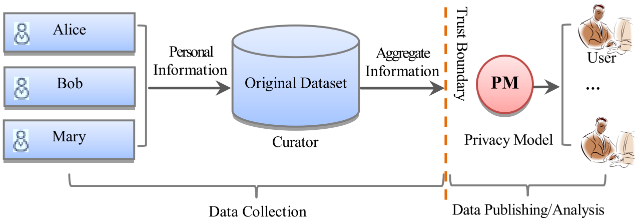


FIGURE 1. Privacy model.

Data sets often contain private or sensitive information, such as medical diagnoses and e-commerce transactions. The information contained within this data has the potential to bring significant benefits to society, research institutions, information consulting organizations, and government decision-making departments, as data has become a very important resource. This has greatly promoted interest in the publishing, sharing, and analysis of data. However, there is always a risk of inadvertently revealing personal information. Fig. 1 shows the process from information collection to privacy protection, which can be divided into data publishing, and data analysis, according to the purpose of the release. Data publishing aims to share data sets publicly or allow the public to run queries on the data. In the literature, this scenario

The associate editor coordinating the review of this manuscript and approving it for publication was Muhammad Asif.

is also called data sharing. Another scenario is data analysis, which provides the public with a data model associated with a particular algorithm, such as data mining and machine learning algorithms. In both cases, various privacy protection methods and privacy criteria are implemented and defined as a privacy model [42].

Privacy protection technology can effectively handle personal data to prevent the leakage of sensitive information during data publishing and data analysis. K anonymity [13] is a technique for protecting private information by generating an anonymized data set such that each unique record is hidden in an equivalence class of k records. This ensures that individual records cannot be uniquely identified from the data, and allows it to resist a linking attack, where an attacker attempts to identify an individual by connecting the anonymized private record with other publicly released data. Improved algorithms such as l-diversity, t-closeness, and (a, k)-anonymity make the private information in each equivalence class more difficult to isolate. However, anonymization technology is vulnerable if the attacker can utilize additional background knowledge and, as there is no clear definition of the assumptions of the attack model. Privacy cannot always be guaranteed.

Differential privacy addresses these issues. This model is based on a new definition of privacy proposed by Dwork *et al.* for statistical databases [5]–[7], [9]. Under this mechanism, whether a record is either in or out of the data set has little effect on the result of a calculation made on the data set. Therefore, the risk of privacy leakage caused by adding or deleting a record from the data set is reduced to a minimum acceptable range, which is determined by a prespecified privacy protection budget. This ensures that the attacker cannot obtain the individual information through the results of calculation. Differential privacy addresses two shortcomings of the traditional privacy protection model: 1) Differential privacy protection is independent of background knowledge, so that even if the attacker has significant background knowledge, it still provides good privacy protection. 2) Differential privacy is based on rigorous mathematical theory and provides a quantitative assessment of the level of privacy protection. The core concepts of differential privacy come from a range of fields such as machine learning, data mining, statistics, and learning theory. Differential privacy has become the de facto privacy standard [19], [25]. In June 2016, at the WWDC conference, Apple Inc. indicated that it implements differential privacy in its latest operating system and applications to protect each user's individual data.

For the privacy leakage problem in data model publishing and analysis, differential privacy is implemented in machine learning algorithms to protect private or sensitive information, while also maximizing the availability of the published data or algorithm [8], [27], [34], [35]. This paper focuses on research into ensemble classification algorithms which are applied to differential privacy. The machine learning ensemble classifier [26] combines multiple independently trained base classifiers to enable better prediction. Random forest and AdaBoost are the two most common algorithms in ensemble learning. The construction of the base classifier can adopt different classification algorithms, such as a decision tree, BP-neural network, etc. Therefore, the degree of flexibility in ensemble learning is very high, so it is widely used in classification problems and various competitions such as the international KDD Cup which is usually won by an ensemble method.

Based on the analysis of existing differential privacy decision trees, this paper improves the ensemble classification model DP-AdaBoost [31] under differential privacy constraints. Since the algorithm is based on ID3 decision trees that can only handle discrete attributes, it is necessary to preprocess continuous attributes in the data set before classification. Aiming at this problem, this paper proposes a CART-DPsAdaBoost algorithm, which does not require discrete preprocessing of data continuous attributes, eliminates the consumption of classification system performance. The algorithm model can handle both discrete feature data and continuous feature data. While maintaining a high classification accuracy, it also takes into account the privacy and availability of the classification model, and can effectively deal with large-scale, high-dimensional data classification problems.

The remainder of this article proceeds as follows. The related work in Section 2 gives an overview of the development of classification methods under differential privacy constraints, culminating with the DP-AdaBoost model. Section 3 gives the theoretical background of differential privacy and adaptive boosting. Next, Section 4 discusses the proposed CART-DPsAdaBoost algorithm in detail and describes how it will be evaluated. Section 5 provides a full examination and discussion of results from a range of experiments comparing the proposed CART-DPsAdaBoost with DP-AdaBoost and the original AdaBoost algorithms, on three standard data sets. The Conclusion summarizes the advantages and limitations of the proposed method, and looks forward to future research.

II. RELATED WORK

There has already been significant research on differential privacy in machine learning. For example, Sarwate and Chaudhuri [27] provided a general overview of the field, briefly discussing classification, regression, dimensionality reduction, time series, filtering, and other essential "building blocks" based on differential privacy. These include differences between the input, output, and target perturbations, the exponential mechanism, and differential privacy statistics. Ji *et al.* [15] focus more on specific algorithms under differential privacy and outline the operation of the naive Bayesian model, linear regression, linear SVM, logistic regression, kernel SVM, decision trees, k-means clustering, feature selection, PCA, and statistical estimation. Abadi *et al.* [1] applied target perturbation to deep learning. They adopted a small-batch stochastic gradient descent method to solve the problem of a non-convex loss function, and added noise at each step of the gradient descent. Shokri and Shmatikov [30] designed a distributed deep learning model that enables multiple parties to apply neural networks in combination.

In this paper, the ensemble classification model under the differential privacy constraint is studied using a decision tree as the base classifier. The decision tree is a commonly used data classification model, implemented in algorithms including ID3, C4.5, and CART [17]. The basic task of differential privacy data analysis is to extend existing non-private algorithms to differential privacy algorithms. Under a given interface mode, a decision tree based on differential privacy can only perform a limited number of queries, and each query will consume some of the privacy budget. Literature [3] proposed the differential privacy decision tree algorithm SuLQ-based on ID3 in an interface mode that uses the count value of Laplace noise to calculate the information gain of an attribute. However, since the privacy budget is consumed multiple times in each iteration, a lot of noise is generated, which leads to a significant reduction in the accuracy of the prediction. In addition, SuLQ cannot handle continuous attributes. In [18], the SuLQ algorithm is improved using Microsoft's Privacy Integrated Query (PINQ) platform. The partition operator is used to segment the data set into disjoint subsets, and then the ID3 decision tree is constructed. To deal with the disadvantages of high noise and only dealing with discrete attributes, literature [11] Friedman and Schuster proposed that the DiffP-ID3 algorithm based on the exponential mechanism evaluates all attributes simultaneously in one query, reducing noise and privacy budget waste. DiffP-C4.5 selects continuous attribute splitting points and achieves optimal attribute partitioning through two exponential mechanisms. The DiffGen algorithm proposed in literature [21] first uses a generalization technique, and then combines the exponential mechanism and information gain to segment the attributes. When the dimensionality of the dataset is low, the privacy protection effect is improved. Literature [43] improved the DiffGen algorithm and proposes the DT-diff algorithm. On this basis, a feature model selection strategy is proposed. By establishing a feature model to group samples, and by adding noise, the algorithm makes full use of the privacy budget and improves classification accuracy. In literature [23], Patil and Singh applied differential privacy based on a random forest, and proposed the DiffPRF algorithm, which uses ID3 as the base learner. Nevertheless, it can only deal with discrete attributes. In literature [14], Mu Hairong *et al* proposed a random forest algorithm, DiffPRFs, based on differential privacy. During the construction of each tree, the exponential mechanism is used to select the splitting point and splitting attribute. This method does not need to carry out discrete preprocessing of data. Literature [20], Mivule *et al*. proposed a framework that uses AdaBoost iterations to update data sets until the forest achieves an acceptable level of prediction accuracy. However, the framework lacks detail in that it does not give the specific content of the differential privacy technology, or how it allocates the privacy budget. The DP-AdaBoost algorithm [31], which is an AdaBoost algorithm with differential privacy protection, uses a single-layer ID3 decision tree as the base classifier for ensemble learning in order to reduce the complexity of the model. The algorithm no longer uses the counting function directly when adding noise, but instead considers the weight value of each record at simultaneously. Also, it does not need to introduce noise in attribute division, so the final result has better classification accuracy. However, this method requires discretization during preprocessing of the data set, which degrades the classification performance. Table 1 summarizes the related research based on decision tree classification methods under differential privacy constraints.

Based on the analysis of existing research, this paper proposes the CART-Differential Privacy structure of AdaBoost (CART-DPsAdaBoost) for classification problems under privacy protection requirements: an AdaBoost classification algorithm based on differential privacy protection. Under the requirement of differential privacy protection, the algorithm maximizes the advantages of AdaBoost ensemble learning. Compared with the similar method of DP-AdaBoost, the method in this paper is less complex, the algorithm's efficiency is improved, while also maintaining its accuracy of classification.

Theoretical analysis and experiments are conducted to demonstrate that the advantages of this algorithm are as follows:

- In the boosting process, the algorithm combines the idea of bagging with a random subspace algorithm for attribute perturbation. While increasing the diversity of the base classifier, the probability that each tuple is selected during the iterative process is determined by its weight. Compared with methods based on sequence sampling, the proposed approach is more efficient at classifying large, high-dimensional data sets.
- During the construction of the CART tree, the continuous feature split point is selected by the exponential mechanism, and the best split feature is selected by the Gini index. Using the improved CART decision stump as the base learner for integrated learning, the complexity of the model is low while avoiding the influence of the depth of the tree on the level-sharing strategy in privacy budget allocation.
- As the number of leaf nodes decreases greatly, more privacy budget will be allocated, the added Laplace noise will be reduced, and the resulting ensemble model maintains a high classification accuracy.

III. DEFINITION AND THEORETICAL BASIS

A. THE BASIS OF DIFFERENTIAL PRIVACY THEORY

1) DIFFERENTIAL PRIVACY

Definition 1 : (ε- *Differential Privacy* [5], [6], [9] Given adjacent data set D and D', there is at most one difference between the records, *i.e.* ($D\Delta D' \leq 1$). A randomized mechanism M, and arbitrary output O ($O \in Range\,(M)$) satisfy the following

TABLE 1. **Comparison of decision tree classification methods based on differential privacy constraints.**

Algorithm	Mechanism	Specific method	Noise	Frame	Data type	Characteristic
SuLQ-based ID3	Laplace	Calculate attribute information gain using count values with Laplace noise	High	Interface mode	Discrete	High noise and reduced data availability
PinQ-based ID3	Laplace	The Partition operator is used to divide the data set and then implement ID3 algorithm	High	Interface mode	Discrete	Increased privacy budget utilization but fails to reduce noise
DiffP-ID3	Laplace Exponential	Dividing attributes by exponential mechanism	Low	Interface mode	Discrete	A split only consumes a privacy budget, reducing noise
DiffP-C4.5	Laplace Exponential	Extend the exponential mechanism to continuous attributes	Low	Interface mode	Discrete Continuous	Excessive privacy budget with two exponential mechanisms
DiffGen	Laplace Exponential	Use generalization techniques and combine exponential mechanisms with information gain	Low	Full access	Discrete Continuous	Better privacy protection when there are fewer attribute types
DT-Diff	Laplace Exponential	Build a feature model to group samples and add noise	Low	Full access	Discrete Continuous	Make full use of privacy budgets to improve accuracy
DiffPRF	Laplace Exponential	Random forest is constructed by ID3 decision tree algorithm	Low	Interface mode	Discrete	Random forests solve dimensional problems, but need to pre-discretize continuous attributes
DiffPRFs	Laplace Exponential	Improved DiffPRF algorithm and extended the exponential mechanism to continuous attributes	Low	Interface mode	Discrete Continuous	Eliminate multi-dimensional big data discretization preprocessing problem
DP-AdaBoost	Laplace	AdaBoost algorithm is constructed by using a single layer ID3 decision tree	Low	Full access	Discrete Continuous	Reduced model complexity and improved classification accuracy

In the interface mode, data mining workers are considered untrustworthy, and the data manager does not publish the original data set, but simply provides an access interface. The data digger can only obtain the results of the counting function after differential privacy protection. In this mode, the privacy protection is fully provided by the interface.

In the full access mode, data mining workers are considered to be credible, and can directly access the data set and modify the execution algorithm to meet the requirements of differential privacy protection, so as to ensure that the final published model will not leak the privacy information in the data set.

inequality, so M satisfies ε-differential privacy.

$$\Pr\left[M\left(D_1\right) \in O\right] \leq e^{\varepsilon} \Pr\left[M\left(D_2\right) \in O\right] \tag{1}$$

where $Pr[\cdot]$ represents the risk of privacy disclosure. The privacy budget ε reflects the level of privacy protection that M can provide. The higher the data security requirement, the smaller ε has to be, and the more similar the probability distribution of query results returned by the differential privacy algorithm on adjacent data sets is, and the more difficult it is for the attacker to distinguish the adjacent data sets. The value of ε should be combined with operational requirements to achieve a balance between security and availability of output.

Fig. 2 shows a more intuitive description of the nature of differential privacy [4], [24]. The selection of the random function M is independent of the background knowledge of the attacker. The query function is f and the adjacent data sets are D and D'. Differential privacy maps the result of the query function $f(\cdot)$ to a randomized value field and feeds results back the user with a certain probability distribution. The degree of approximation of the probability distribution on the adjacent data set is controlled by the parameter ε, so that the output results are almost identical, thereby achieving the purpose of protecting the individual information in the data set.

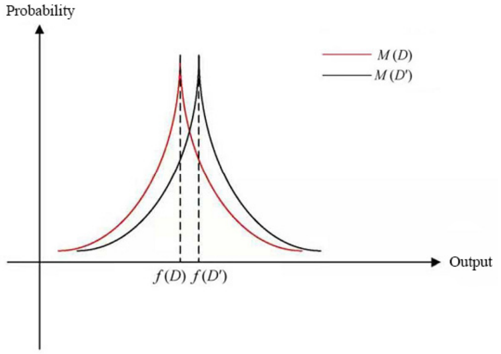


FIGURE 2. **The output probability of differential privacy on adjacent data sets.**

Definition 2 (*Global Sensitivity* [5], [6], [9]): For adjacent data sets D and D' that satisfy $\left|D\Delta D' = 1\right|$, given the query function $f : \mathrm{D} \rightarrow \mathrm{R}^d$, the sensitivity of the function f is:

$$\Delta f = \max_{D_1, D_2} \left\| f\left(D_1\right) - f\left(D_2\right) \right\| \tag{2}$$

where R represents the mapped real space, d represents the query dimension of function f, and $\left\| f\left(D_1\right) - f\left(D_2\right) \right\|$ represents the first normal form (1NF) distance between $f\left(D_1\right)$ and $f\left(D_2\right)$.

Global sensitivity reflects the maximum degree of change in the results of a query function for a pair of adjacent data sets. For most query functions f, the value of Δf is relatively small, such as for the counting function, the sensitivity $\Delta f = 1$. It is worth mentioning that sensitivity is only one of the properties of function f, that is independent of the data set D.

2) IMPLEMENTATION MECHANISM

The main implementation mechanism of differential privacy is the noise mechanism. Laplace mechanism and exponential mechanism are the two most commonly used mechanisms. The noise mechanism is constrained by the global sensitivity and privacy protection budget. Too much noise will affect the usability of the results, while too little will not provide sufficient security.

Definition 3 (*Laplace Mechanism* [12], [16], [19]): Given an arbitrary function $f : D \rightarrow R^d$, if the output of the expression $K(D)$ satisfies the following equation, then $K(D)$ satisfies the requirements of ε - differential privacy.

$$K(D) = f(D) + \left(Lapace\left(\frac{\Delta f}{\varepsilon} \right) \right)^d \tag{3}$$

where $Laplace\left(\Delta f / \varepsilon\right)$ is the Laplace distribution obeying the scale parameter $\Delta f / \varepsilon$, and the amount of noise is related to the values of Δf and ε.

The Laplace mechanism uses the noise generated by the Laplace distribution to disturb the true output value to achieve differential privacy protection. Since the Laplace mechanism is only suitable for the protection of numerical results, for non-numeric data, such as entity objects, McSherry and Talwar [44] proposed an exponential mechanism.

Definition 4 (*Exponential Mechanism* [12], [16], [19]): Given a utility function $q : (D \times O) \rightarrow r\ (r \in Range)$, if the Function $F(D, q)$ satisfies the following equation, then $F(D, q)$ satisfies ε- differential privacy.

$$F(D, q) = \{r : Pr[r \in O]\} \propto \exp\left(\frac{\varepsilon q(D, r)}{2\Delta q} \right) \tag{4}$$

where D is the input data set, the output is an entity object r, and Δq is the global sensitivity of the utility function $q(D, r)$. The function F selects and outputs r from *Range* in proportion to the probability of $\exp\left(\varepsilon q(D, r) / 2\Delta q\right)$.

3) RELATED LEMMA

Usually a complex privacy protection problem has to be solved by applying differential privacy protection algorithms multiple times. In this case, in order to ensure that the privacy protection of the entire process is controlled within a given budget ε, it is necessary to reasonably allocate all privacy budgets to the various steps of the entire algorithm. At this point, two supporting lemmas can be used.

Lemma 1 (*Sequential Composition* [19], [24]): Suppose, in a set of mechanisms $M_1, M_2, \ldots, M_n$, if M_i satisfies ε_i-differential privacy, then for the same data set D, the combining mechanism $M(M_1(D), M_2(D), \ldots, M_n(D))$ will provide $\sum_{i=1}^{n} \varepsilon_i$- differential privacy.

Lemma 2 (*Parallel Composition* [19], [24]): Suppose, in a set of mechanism $M_1, M_2, \ldots, M_n$, if M_i satisfies ε_i -differential privacy, then for disjoint data sets $D_1, D_2, \ldots, D_n$, the combining mechanism $M(M_1(D), M_2(D), \ldots, M_n(D))$ will provide $(\max \varepsilon_i)$-differential privacy.

B. CLASSIFICATION AND REGRESSION TREE

1) IMPROVED CART

A Classification and Regression Tree (CART) is a popular binary decision tree that can be used for classification or regression analysis [10], [33], [41]. This paper considers to reduce the complexity of the model while increasing the diversity of the model to maintain a certain degree of accuracy of the base classifier. To this end, two types of randomness schemes, sample perturbation and feature perturbation, are introduced in the process of building the base classifier, that is, the bootstrap sampling scheme is introduced during the iteration of the ensemble algorithm, and the random subspace algorithm is used in feature selection.

Literature [14] Mu Hairong *et al.* proposed DiffPRFs, a random forest algorithm based on differential privacy protection. This algorithm uses an exponential mechanism to select split points and split features during tree construction, and adds noise according to the Laplace mechanism. Although the algorithm does not require discrete preprocessing of the data, the exponential mechanism is called twice per iteration, which consumes a large amount of privacy budget, resulting in a low utilization rate of the privacy protection budget.

In this paper, based on the analysis of existing research, in the construction of the base classifier CART tree, an exponen tial mechanism is used to select the split point for continuous attributes, and the Gini index is used to select the best split fe ature, which ultimately ensures the utilization of privacy prot ection budget. Formally, we define the following parameters: Sample set D_i, sample size N_i, attribute set $\Lambda\ \left(\Lambda' \in \Lambda\right)$, R_i is the size of the set of spaces evaluated by the utility function q, number of class labels k, and entity object r. We record the probability P_{r_k} of belonging to category $k\ (k \in K)$. The gini value of the probability distribution is $Gini(\Pr) = \sum_{k=1}^{K} \Pr_k (1 - \Pr_k)$
$= 1 - \sum_{k=1}^{K} \Pr_k^2$. Then the CART classification tree in this paper is a binary tree that satisfies the following conditions:

i) Measure the Gini index value of sample set D_i: $Gini(D_i) = 1 - \sum_{k=1}^{K} \left(\frac{N_i}{N}\right)^2$

ii) Consider each possible binary partition for each attribute;

For continuous attributes, the exponential mechanism is used to select the splitting points:

$$\frac{\exp\left(\frac{\varepsilon}{2\Delta q} q\left(D_k, \Lambda'\right)\right) |R_i|}{\sum_i \exp\left(\frac{\varepsilon}{2\Delta \mathrm{q}} q\left(D_k, \Lambda'\right)\right) |R_i|} \tag{5}$$

For discrete attributes, calculate all subsets to get the minimum Gini index:

$$Gini\left(D, \Lambda'\right) = \frac{N_1}{N} Gini\left(D_1\right) + \frac{N_2}{N} Gini\left(D_2\right) \tag{6}$$

iii) Determine whether the termination condition has been reached;

iv) Recursive call i) -iii);

v) Return improved CART decision tree.

Algorithm 1 *Build_DPTree* $(D(i), \Lambda, \varepsilon_c, d)$

```
Input: $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$; $x_i \in \Lambda$; $y_i \in \{-1, 1\}$; CART tree depth, $d$; Privacy budget, $\varepsilon_c$.
Output: Base classifier $g_t(x)$.
1: repeat
2:   $\bar{\varepsilon} = \frac{\varepsilon_c}{d+1}$
3:   Select $\bar{\Lambda} = \{\Lambda_1, \cdots, \Lambda_k\}$ from $\Lambda$ by random
       subspace algorithm;
4:   Gini index minimization principle, choose the optimal
       $\Lambda'$ binary division from $\bar{\Lambda}$;
5:   if Node satisfies the termination condition: then
6:       Classification leaf node
         $D_y = Partition\left(D(i), \forall y \in [Y] : r_y = y\right)$;
7:       Return leaf nodes to mark:
         $\max_y\left(N_y = Count\left(\left|D_y\right|\right)\right)$;
8:   else if $\bar{\Lambda}$ contains $n$ Continuous attributes, perform
       step 9 then
9:       $\bar{\bar{\varepsilon}} = \frac{\bar{\varepsilon}}{2(n+1)}$
10:      Select continuous attribute split point
         $$\frac{\exp\left(\frac{\bar{\bar{\varepsilon}}}{2\Delta q} q\left(D_y, \Lambda'\right)\right) |R_i|}{\sum_i \exp\left(\frac{\bar{\bar{\varepsilon}}}{2\Delta q} q\left(D_y, \Lambda'\right)\right) |R_i|}$$
         where $q\left(D_y, \Lambda'\right)$ is availability function, $\Delta q$
         is sensitivity, $|R_i|$ is the size of interval set;
11:      Divide data set $D$ into $D_l$ and $D_r$;
12:      Build left and right subtrees:
           $t_l = Build\_DPTree\left(D_l(i), \Lambda, \bar{\varepsilon}, d+1\right)$;
           $t_r = Build\_DPTree\left(D_r(i), \Lambda, \bar{\varepsilon}, d+1\right)$;
13:  end if
14: until Node label consistent, Maximum depth $d$, privacy
       budget exhausted.
15: return $g_t(x)$
```

The improved CART decision tree reduces the complexity of the model by controlling the depth of the tree. When the tree is built, the data is divided into two and entered into the left and right subtrees respectively. The algorithm can handle both discrete features and continuous features.Compared with the ID3 and C4.5 decision tree algorithms, the Gini index is used to select the best split feature, which avoids the preference impact of information gain and information gain rate on the number of feature values.The CART classification tree in this paper has a better advantage as a base classifier for ensemble learning.

C. ENSEMBLE LEARNING ADABOOST

1) WEAKLY LEARNABLE THEOREM

AdaBoost originated from an important theoretical issue discussed in [Kearns & Valiant, STOC '89]: "weakly learnable"?= "strongly learnable?" For the Probably Approximately Correct (PAC) learning theory in machine learning. The PAC Model is defined as follows [28], [29]:

Definition 5 (*Learnable or strongly learnable*): Suppose we have a polynomial algorithm that has a high probability of obtaining a learning model with a small error, which is called learnable or strongly learnable. The formal definition for 0<δ, $\varepsilon \leq 0.5$ is as follows:

$$P_r\left(E_{x \sim D}\left[\text{II}\left[h(x) \neq f(x)\right]\right] < \varepsilon\right) \geq 1 - \delta \tag{7}$$

Definition 6 (*Weakly learnable*): Suppose we have a polynomial algorithm that outputs a learning model with an error of 0.5-$1/p$(p is the parameter of the polynomial). Then the process is called weakly learnable.

In 1990 Schapire and Freund [29] proved the equivalence of weakly learnable and strongly learnable algorithms through constructive methods. A boosting algorithm was then proposed, which can combine several weak learning algorithms that are slightly better than random guessing into a high-precision strong learning algorithm as an alternative to searching directly for a strong learning algorithm that is difficult to find under normal circumstances. This is the famous weakly learnable theorem.

2) ADABOOST (ADAPTIVE BOOSTING)

Ensemble learning can achieve better generalization performance than a single learner by combining multiple weak learners. At present, ensemble learning uses bagging and boosting methods. A random forest is a popular type of bagging method. In the boosting algorithm family, AdaBoost is currently the most successful.

AdaBoost [28] is an iterative process that adaptively changes the distribution of training samples, allowing subsequent base classifiers to focus on previously intractable samples. The core idea is: given the training set, initialize the sample distribution, call the weak learning algorithm to obtain the base classifier, and then adjust the weight of the training samples according to the error rate of the classifier, so as to reduce the weight of the correctly classified samples and increase the weight of the incorrectly classified samples. Based on the new sample distribution, after a number of iterations, a set of complementary base classifiers are obtained and linearly combined into a strong classifier to improve the accuracy and stability of the ensemble classifier. The algorithm classification model is shown in Figure 3, and the specific pseudo-code is shown in algorithm 2.

IV. PROPOSED ALGORITHM

The AdaBoost algorithm continuously generates weak classifiers through a series of iterations, but does not include privacy protection. If there are two data sets D_1 and D_2, and their data differ by only one record, the difference in classification

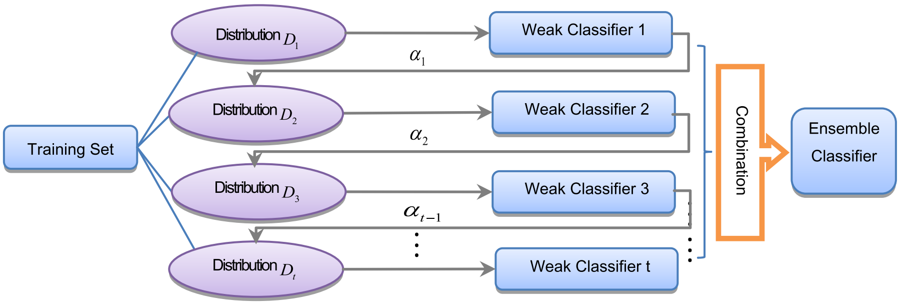


FIGURE 3. AdaBoost classification model. D_t represents the sample distribution of the data set after each iteration, and α_t represents the weight coefficient of the base classifier.

Algorithm 2 AdaBoost

```
Input: $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$; base learning algorithm, $\mathfrak{L}$; number of iterations, $T$.
Initialize weight distribution: $D_1(x) = \frac{1}{n}$;
 for $t = 1$ to $T$ do
  Call algorithm $\mathfrak{L}$ to generate base classifier:
     $g_t = \mathfrak{L}(D, D_t)$;
  Estimate the error of $g_t$: $\tau_t = P_{x \sim D_t}(g_t(x) \neq y_i)$;
  if $\tau_t > 0.5$ then
     break;
  end if
  Weight of $g_t$: $\alpha_t = 1/2 \ln\left(1-\tau_t/\tau_t\right)$;
  Update sample distribution:
     $D_{t+1}(x) = \frac{D_t(x)}{Z_t} \times \begin{cases} \exp(-\alpha_t), \text{if } g_t(x) = y_i \\ \exp(\alpha_t), \text{if } g_t(x) \neq y_i \end{cases} = \frac{D_t(x) \exp(-\alpha_t y_i g_t(x))}{Z_t}$
     where $Z_t$ is the normalization factor to ensure that $D_{t+1}$ is a distribution.
 end for
Output: $G(x) = sign\left(\sum_{t=1}^{T} \alpha_t g_t(x)\right)$
```

results may be reflected on each weak classifier. Therefore, privacy protection of the entire AdaBoost algorithm results in the need to add differential privacy protection noise for each weak classifier.

Assuming that the data provider can perform a series of classification analysis on the data and has full access to the entire data set, it can add noise in the classifier generation process. At this time, in the interface mode, data mining workers can only obtain the counting function results after differential privacy protection, and construct their own classifiers. Therefore, it is necessary to add differential privacy noise in the construction of weak classifiers. However, for the AdaBoost algorithm, the differential privacy noise is added to the weak classifier generation process, which will bring the noise to the subsequent weak classifier construction, causing the noise to be continuously superimposed, and ultimately affecting the quality of the generated results.

Therefore, in order not to cause large deviations in the final classification results, the algorithm in this paper adopts full access mode to protect the data set. Add the corresponding noise after the base classifier is constructed, and calculate the corresponding maximum data set weight value when generating each base classifier. After obtaining the base classifier, it is necessary to add differential privacy noise in combination with the weight value to obtain the classification result with differential privacy.

A. ALGORITHM FRAMEWORK

1) ALGORITHM MODEL

Given the number of iterations of the algorithm is T and the total privacy budget is ε_p, each iteration generates a new base classifier. The construction process of the algorithm model is as follows. The specific pseudo-code is shown in algorithm 3.

Step 1: Initialize the weight distribution of data samples.

Step 2: Recursively use the bootstrap sampling strategy and call Algorithm 1 to generate the base classifier.

Step 3: Calculate the proportion of misclassified data, and calculate the corresponding weight coefficient of the base classifier in the overall classifier according to the error rate.

Step 4: Update the weight distribution of the data samples according to the error rate.

Step 5: According to the weight distribution, find the maximum weight value as the sensitivity, and calculate the noise value required by each leaf node according to the noise formula, add it into the base classifier, and get the base classifier that satisfies the differential privacy protection.

Step 6: Determine whether the current number of iterations has satisfied the given value, and terminate if it is satisfied, otherwise perform the next step.

Step 7: Return to Step 2 and continue to build a new base classifier.

Step 8: The obtained base classifiers are linearly combined to obtain a final integrated classifier that meets differential privacy protection.

The strong classifier generated through the above steps is an integrated classification algorithm model with differential privacy protection, which can be directly distributed to data mining workers without worrying about privacy leakage.

Assume that the initial training set is $D = (x_2, y_2)\{(x_1, y_1), \ldots, (x_n, x_n)\}$, with attribute collection, Λ, and class label set, Y. The size of the data set is n. The initial sample weight distribution $D_1 = \left(\omega_{1,1}, \ldots, \omega_{1,i}, \ldots, \omega_{1,n}\right), \omega_{1,i} = 1/n$, $i = 1, \ldots, n$.

Algorithm 3 CART-DPsAdaBoost

```
Input: $D = \{(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)\}$; $x_i \in \Lambda$; $y_i \in \{-1, 1\}$; number of iterations, $T$; Privacy budget, $\varepsilon_p$.
Output: $\varepsilon_p$-differential privacy classifier:
$G(x) = \left\{\hat{g}_1(x), \hat{g}_2(x), \cdots, \hat{g}_t(x)\right\}$;
Initialize weights: $D_1(i) = {}^1/_n, i \in [1, n]$;
Error function: $E(g(x), y, i) = exp(-y_i g(x_i))$;
$\varepsilon_c = \frac{\varepsilon_p}{T} - log(\alpha_t)$;
for $t = 1$ to $T$ do
    Select $D_t$ of size $|D|$ from $D$ by bootstrap sampling;
    Call algorithm 3 to generate base classifier $g_t(x)$
        $g_t(x) = Build\_DPTree(D(i), \Lambda, \varepsilon_c, d)$
    Calculate the error rate of $g_t(x)$:
        $\tau_t = \sum \omega_t(i) I[g_t(x_i) \neq y_i], i \in [1, n]$;
    if $\tau_t > 0.5$ then
        break
end if
Weight coefficient of $g_t(x)$: $\alpha_t = 1/2 \ln\left(1 - \tau_t/\tau_t\right)$;
Update weights distribution:
        $D_{t+1}(i) = \frac{D_t(i) \exp(-\alpha_t y_i g_t(x_i))}{Z_t}$
    where $Z_t$ is the normalization factor
        $Z_t = \sum_{i=1}^{n} D_t(i) \exp(-\alpha_t y_i g_t(x_i))$
Add Laplace differential privacy noise to $g_t(x)$
    $\hat{g}_t(x) = g_t(x) + Laplace\left(\frac{\max \omega_{t,i}(T \log \alpha_t)}{\varepsilon_c}\right)$
end for
return $G(x) = sign\left(\sum_{t=1}^{T} \alpha_t \hat{g}_t(x)\right)$
```

The algorithm protects private information through the Laplace noise mechanism. The amount of noise is controlled by the privacy budget parameter ε. The reasonable allocation of privacy budgets allows the added noise to protect user privacy without reducing the effectiveness of the data due to excessive noise. Since each prediction is obtained from a leaf node, Laplace noise is added to the leaf node after the weak classifier is generated. When calculating the level of noise, the weight value ω_i of each record needs to be considered so that the function sensitivity dynamically changes in the differential privacy noise calculation formula. According to the sensitivity formula, $S(f) = \max(\omega_i)$ is obtained, and the Laplace noise formula is converted into $K(D) = f(x) + \left(Laplace\left(max(\omega_i)/\varepsilon\right)\right)^d$, and finally the class label and the total number of records with noise of each leaf node are obtained [31], [33]. The utility function of the exponential mechanism in the algorithm uses the Gini index, which is the division criterion in the CART algorithm.

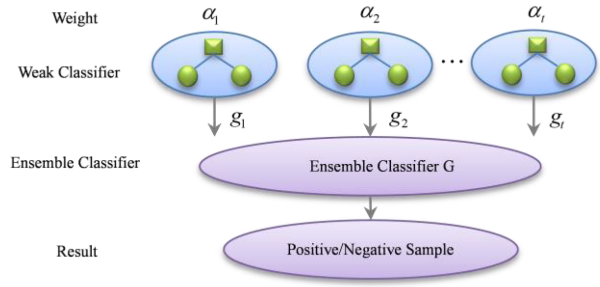


FIGURE 4. Improved algorithm model.

$$\Delta Gini(\Lambda) = Gini(D) - Gini_{\Lambda}(D)$$

$$q_{Gini}(R, \Lambda) = -\sum_{i\in\Lambda} R_i^{\Lambda}\left(1 - \sum_{y\in Y}\left(\frac{R_{i,y}^{\Lambda}}{R_i^{\Lambda}}\right)^2\right) \qquad (8)$$

where R_y means that a record belongs to label y, $R_{i,y}^{\Lambda}$ means that a record values Λ_i on feature Λ, and its label is y.

Minimizing the Gini index is equivalent to maximizing $Gini_{\Lambda}(D)$ with a sensitivity of $\Delta q_{Gini} = 2$. Compared with an information gain sensitivity of $\log(|D| + 1) + 1/\ln 2$, and the lower sensitivity of the Gini index as the scoring function, this improves the efficiency of the exponential mechanism [2], [36]. In this paper, the number of iterations is specified in advance, and the weight value α_t of each weak classifier is considered in the privacy budget allocation, that is, the budget assigned to each weak classifier is $\varepsilon_p/T - \log(\alpha_t)$.

2) CLASSIFICATION

In machine learning model training, the complex and accurate weak classifier is easy to overfit after integration, which leads to the decrease of the accuracy of prediction.Considering the influence of tree depth on privacy budget allocation and time efficiency, the weak classifier with low model complexity and certain accuracy will have better classification results and time cost after integration. Adaboost does not need such a precise decision tree. The simplest way is to set the depth of the tree to 1, that is, each tree is composed of a split attribute and two leaf nodes. The algorithm's classifier model is shown in Figure 4.

The above algorithm generates an ensemble classifier $G(x)$ under ε_p-differential privacy, and the process of classifying the test set is shown in Algorithm 4.

For each record in the new sample set, apply each base classifier in the final ensemble model to classify and predict it. The classification result obtained by each base classifier is multiplied by the corresponding weight, and then a linear combination is performed to obtain the final classification result. The classification results of all records are then output. The algorithm model performs well on large data sets, can process high-dimensional data, and has fast training

Algorithm 4 Classify New Samples

```
Input: Test set, $\hat{M}$; $\varepsilon_p$-differential privacy classifier $G(x)$.
Output: Classification result $Y\left(\hat{M}\right)$.
repeat
  For each sample to be classified
    for $i = 1$ to $m$ do
      From the current root node to the leaf node;
       Calculate the product of the current tree
       prediction result and the tree weight;
      Linear combination: $sign\left(\sum_{t=1}^{T} \alpha_t \hat{g}_t(x)\right)$
  end for
until Arrival leaf node
return Output result $Y\left(\hat{M}\right)$ of sample set $\hat{M}$
```

speed, which realizes effective prediction of large-scale, high-dimensional data classification.

B. ALGORITHM PERFORMANCE

1) PRIVACY

In this paper, in the final classifier generation process, the number of iterations of the algorithm is defined in advance, so the CART-DPsAdaBoost algorithm adopts a hierarchical equalization method. Firstly, the total privacy budget ε_p is evenly distributed to each tree $\varepsilon_c = \varepsilon_p/T - \log(\alpha_t)$. The sample set of each tree based on bootstrap sampling has an intersection. According to differential privacy Lemma 1, the total privacy budget is a superposition of the budget consumed by each tree. For each tree, the privacy budget is evenly distributed to each layer $\bar{\varepsilon} = \varepsilon_c/(d+1)$, and the samples of each node in each layer are disjoint subsets. According to differential privacy Lemma 2, the budget of each layer is not superimposed, that is, the budget of each node is $\bar{\varepsilon} = \varepsilon_c/(d+1)$. We use half $\bar{\varepsilon}/2$ of each node budget to estimate the number of node instances and decide whether the node meets the termination condition. If so, the node is designated as a leaf node and we use the other half of the budget to determine the leaf node class count. If the current node has n continuous attributes, the budget of the other half is divided into $n+1$, which is used to select the splitting point of each continuous attribute. The budget consumed by the exponential mechanism is $\bar{\bar{\varepsilon}} = \bar{\varepsilon}/2(n+1)$ each time. According to Lemma 1, the budget consumed by the multiple exponential mechanisms is a superposition of each time. The total privacy budget consumed by the algorithm is not greater than ε_p, satisfying ε- differential privacy protection. The proof is as follows:

Proof: Assume adjacent data sets D and D', $\left|D\Delta D' = 1\right|$, $M(D)$ and $M\left(D'\right)$ respectively represent the output of the random algorithm, the total privacy budget is ε_p, the weight of the base classifier $g_t(x)$ is α_t, and the number of base classifiers is T. For each continuous attribute, there are r_i division methods, and the probability that r_i is selected by the exponential mechanism is:

$$p(r_i) = \frac{E(D, r_i)}{\sum_{r_i \in Range} E(D, r_i)} \tag{9}$$

where $E(D, r_i)$ represents $\exp\left(\hat{\varepsilon} q(D, r_i)/2\Delta q\right)$ and $p(r_i)$ is a weight, then the continuous attribute partitioning scheme r_i participates in the global selection with a probability proportional to $p(r_i) E(D, r_i)$. The Gini index is used to divide the discrete attributes. If the attribute Λ' has v values, there are $(2^v - 2)/2$ divided subsets, and the calculated Gini index values for attribute Λ' are:

$$Gini\left(D, \Lambda'\right) = \frac{N_1}{N} Gini(D_1) + \frac{N_2}{N} Gini(D_2) \tag{10}$$

where N_1 and N_2 are disjoint subsets, and $\hat{\varepsilon} = \varepsilon_c/\left((d+1)\left|(2^v-2)/2\right|\right)$ is obtained according to differential privacy Lemma 1 The differential privacy budget for the Gini index can be converted according to Lemma 2:

$$\begin{aligned}
&\frac{prob\,(M(D) = r_i)}{prob\,(M(D') = r_i)} \\
&= \frac{\prod_{i=1}^{\left|\frac{2^v-2}{2}\right|} p(r_i) E(D, r_i)}{\prod_{i=1}^{\left|\frac{2^v-2}{2}\right|} p(r_i) E(D', r_i)} \\
&= \frac{\prod_{i=1}^{\left|\frac{2^v-2}{2}\right|} \frac{\exp\left(\frac{\hat{\varepsilon} q(D,r_i)}{2\Delta q}\right)}{\sum_{i \in Range} \exp\left(\frac{\hat{\varepsilon} q(D,r_i)}{2\Delta q}\right)} \exp\left(\frac{\hat{\varepsilon} q(D,r_i)}{2\Delta q}\right)}{\prod_{i=1}^{\left|\frac{2^v-2}{2}\right|} \frac{\exp\left(\frac{\hat{\varepsilon} q(D',r_i)}{2\Delta q}\right)}{\sum_{r_i \in Range} \exp\left(\frac{\hat{\varepsilon} q(D',r_i)}{2\Delta q}\right)} \exp\left(\frac{\hat{\varepsilon} q(D',r_i)}{2\Delta q}\right)} \\
&= \prod_{i=1}^{\left|\frac{2^v-2}{2}\right|} \frac{\left(\exp\left(\frac{\hat{\varepsilon} q(D,r_i)}{2\Delta q}\right)\right)^2}{\left(\exp\left(\frac{\hat{\varepsilon} q(D',r_i)}{2\Delta q}\right)\right)^2} \cdot \frac{\sum\limits_{r_i \in Range} \exp\left(\frac{\hat{\varepsilon} q(D',r_i)}{2\Delta q}\right)}{\sum\limits_{r_i \in Range} \exp\left(\frac{\hat{\varepsilon} q(D,r_i)}{2\Delta q}\right)} \\
&= \prod_{i=1}^{\left|\frac{2^v-2}{2}\right|} \left(\exp\left(\frac{\hat{\varepsilon}\left(q(D, r_i) - q\left(D', r_i\right)\right)}{2\Delta q}\right)\right)^2 \\
&\quad \times \frac{\sum\limits_{r_i \in Range} \exp\left(\frac{-\hat{\varepsilon}\left(q(D,r_i)-q\left(D',r_i\right)\right)}{2\Delta q}\right) \exp\left(\frac{\hat{\varepsilon} q(D,r_i)}{2\Delta q}\right)}{\sum\limits_{r_i \in Range} \exp\left(\frac{\hat{\varepsilon} q(D,r_i)}{2\Delta q}\right)} \\
&\leq \prod_{i=1}^{\left|\frac{2^v-2}{2}\right|} \left(e^{\frac{\hat{\varepsilon}}{2}}\right)^2 \frac{e^{\frac{-\hat{\varepsilon}}{2}} \sum\limits_{r_i \in Range} \exp\left(\frac{\hat{\varepsilon} q(D,r_i)}{2\Delta q}\right)}{\sum\limits_{r_i \in Range} \exp\left(\frac{\hat{\varepsilon} q(D,r_i)}{2\Delta q}\right)} \\
&= \prod_{i=1}^{\left|\frac{2^v-2}{2}\right|} e^{\frac{\hat{\varepsilon}}{2}} \\
&= \prod_{i=1}^{\left|\frac{2^v-2}{2}\right|} e^{\frac{\varepsilon_c}{2(d+1)\left|\frac{2^v-2}{2}\right|}} \\
&= e^{\frac{\varepsilon_c}{d+1}}
\end{aligned} \tag{11}$$

According to Lemma 2, the degree of differential privacy protection for each tree is:

$$\prod_{i=1}^{|d+1|} e^{\frac{\varepsilon_c}{d+1}} = e^{\varepsilon_c} \tag{12}$$

Therefore, ε_c-differential privacy protection is provided.

For the ensemble classification model $G(x)$, the total differential privacy budget satisfies:

$$\begin{aligned}
&\frac{Prob\left(M\left(D\right)=R\right)}{Prob\left(M\left(D'\right)=R\right)}\\
&\leq \frac{\prod_{i=1}^{T}\alpha_t \exp\left(\frac{q(D,r_i)}{2\Delta q}\right)\cdot\left(\frac{\varepsilon_p}{T}-\log\alpha_t\right)}{\prod_{i=1}^{T}\alpha_t \exp\left(\frac{q(D',r_i)}{2\Delta q}\right)\cdot\left(\frac{\varepsilon_p}{T}-\log\alpha_t\right)}\\
&=\prod_{i=1}^{T}\left(\alpha_t \exp\left(\frac{\varepsilon_p}{T}-\log\alpha_t\right)\cdot\left(\left|q\left(D,r_i\right)-q\left(D',r_i\right)\right|\right)\right)\\
&=e^{\sum_{i=1}^{T}\left(\log\alpha_t+\frac{\varepsilon_p}{T}-\log\alpha_t\right)}\\
&=e^{\varepsilon_p}
\end{aligned} \tag{13}$$

Therefore, $G(x)$ provides ε_p- differential privacy.

2) APPLICABILITY

The CART decision tree can handle both continuous and discrete attributes, and can eliminate the influence of the number of feature values. In this paper, the exponential mechanism is used to deal with continuous attributes, and the Gini index deals with discrete attributes, so that the algorithm can process both continuous attribute data and discrete attribute data, and ensures the reasonable use of privacy budget.

The introduction of two types of randomness schemes makes the ensemble model under differential privacy protection still have good performance, and to a certain extent solves the problems caused by large-scale, high-dimensional data classification. This is because in the training process of the algorithm, the training samples of each tree are obtained through bootstrap sampling. As the training sets are different, the decision tree models generated are different. The random subspace algorithm is introduced when selecting the best split feature, so that the decision tree model generated by the same training set may also be different. The diversity of the final integrated model comes not only from the sample perturbation in the training set, but also from the feature perturbation, which makes the generalization performance of the model can be further improved by the difference between the base classifiers, and the applicability of the algorithm is better.

3) PERFORMANCE EVALUATION METHOD

F-Measure is a commonly used index used to evaluate the performance of the classifier. It comprehensively measures the accuracy and recall rate of the model, and the value range is [0, 1]. Consider a two-category problem where the outcome can be either positive or negative [32], [37]. If the prediction is positive and the real outcome is positive, this is a true positive (TP). If the prediction is positive and the observed result is negative, this is a false positive (FP). A false negative (FN) occurs when the prediction is negative but the result is positive. The F-measure provides a measurement that combines both precision and recall. The evaluation indicators are defined as follows:

TABLE 2. Experimental data set.

Data set	Number of samples	Number of features	Task
Adult	32561	14	classification
Census Income	199523	41	classification
Digit Recognizer	42000	784	classification

$$Precision = \frac{TP}{TP+FP} \tag{14}$$

$$Recall = \frac{TP}{TP+FN} \tag{15}$$

$$F-Measure = \frac{\left(1+\beta^2\right)\cdot \Pr ecision\cdot Recall}{\beta^2\cdot \Pr ecision+Recall} \tag{16}$$

where β is a coefficient that adjusts the relative importance of *Precision* and *Recall*. When the value of β is 1, *F-Measure* is defined as *F1Score*, and the larger the value of *F1Score*, the better the algorithm availability.

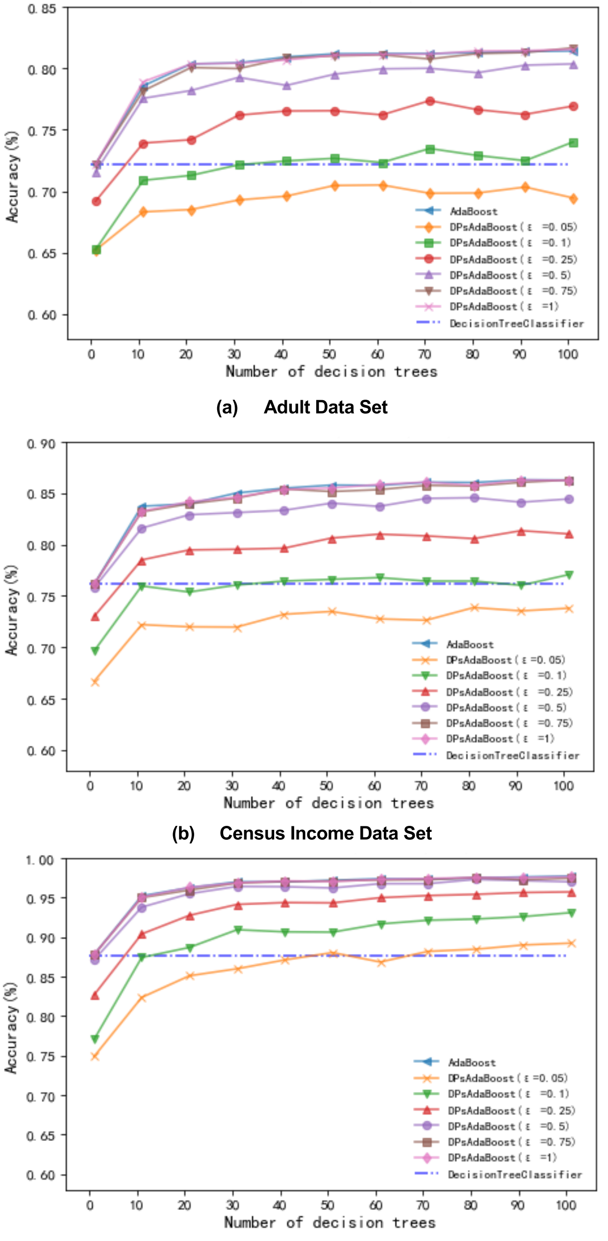
In the experiment, the Receiver Operating Characteristic (ROC) Area Under the Curve (AUC) is also used to evaluate the generalization performance of the classifier [35]. The ROC curve is drawn from the ratio between the TP and FP calculated from the confusion matrix. The value of the AUC is in the range [0, 1]. Higher values of the AUC indicate better generalization performance of the classifier.

C. SOME OPTIMIZATION OF THE ALGORITHM

Differential privacy noise is related to the privacy budget and function sensitivity. For adjacent data sets D and D', the sensitivity of the counting function is always 1. However, considering the change in the weight value, the sensitivity becomes a statistical function of the weight, and the maximum weight value must be evaluated in each iteration. In the AdaBoost algorithm, if a record is classified incorrectly in the current iteration, its weight value is increased in the next iteration. For noise records (also called outliers), the weight will become exceptionally high after several iterations, which makes the added differential privacy noise excessive, and reduces the classification performance. Therefore, it is necessary to perform some form of outlier detection. This is done by means of a threshold parameter θ. When the record weight is greater than θ. in the current iteration, the sample weight is set to 0. In step 12 of Algorithm 2, the formula for calculating the weight becomes:

$$D_{t+1}(i)=\begin{cases}\dfrac{D_t(i)\exp\left(-\alpha_t y_i g_t(x_i)\right)}{z_t}, & \text{if } \mathrm{D}_t(i)\leq\theta\\ 0, & if\, \mathrm{D}_t(i)>\theta\end{cases} \tag{17}$$

2) For an unbalanced data set, the model is trained by changing the distribution of the unbalanced data to obtain a new sample set with a more balanced ratio [22], [39], [40].

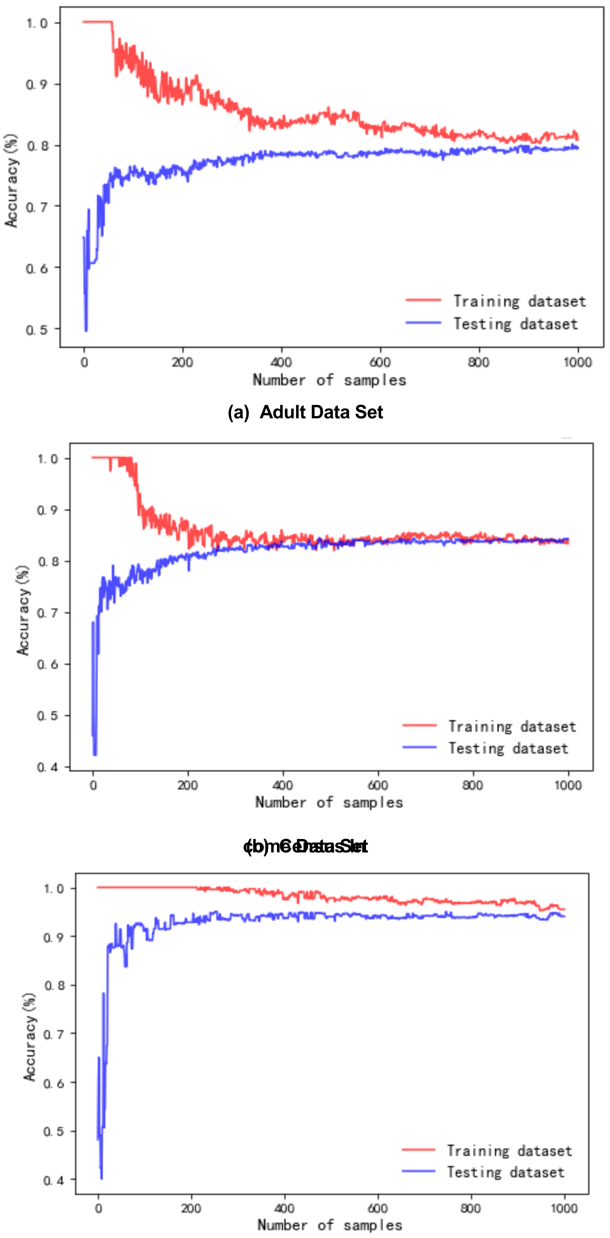


(a) Adult Data Set

(b) Census Income Data Set

(c) zeDigitRSeetogni

FIGURE 5. Classification accuracy of CART-DPsAdaBoost under different conditions in 3 data sets.



(a) Adult Data Set

(b) mCeDatasSet

(c) Digit Recognizer Data Set

FIGURE 6. The relationship between sample size and model accuracy.

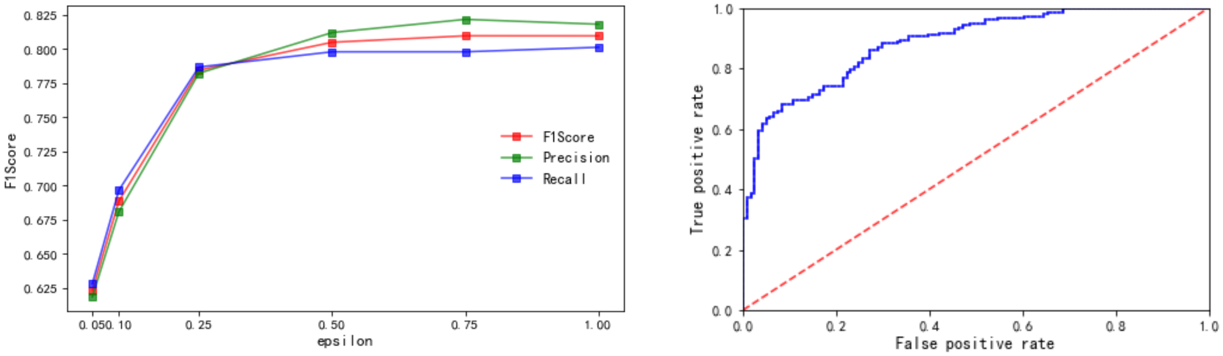
Literature [38] proposed a new Random Balance sampling-AdaBoost (RBS-AdaBoost) algorithm, which can randomly change the unbalanced rate and distribution of unbalanced data to conduct classification and learning on the generated data sets.

V. EXPERIMENTAL SCHEME AND RESULT ANALYSIS

A. EXPERIMENTAL DATASET

The classifier training, testing, and data preprocessing in this article are all implemented in Pycharm based on Python 3.7. The Adult Data Set from the UCI Machine Learning Repository was used to help design the algorithm control experiments. The validity of the algorithm was verified using the Census Income large-scale data set. Finally, Kaggle Data's Digit Recognizer dataset was used to further validate the effectiveness of the algorithm in dealing with large-scale, high-dimensional data classification problems. Table 2 shows the format of the three data sets.

The Adult Data Set contains 6 continuous attributes and 8 attributes. The category attribute is income level, and the category values are "<= 50K" ">50K". There are 32,561 tuples (no missing values) in the data set, 70% of which were used as the training set and 30% as the test set. Census Income has 41 attributes, including



(a) *Precision/Recall/F1Score* changes with ε on the Adult Data Set

(b) ROC curve on Adult Data Set

FIGURE 7. The algorithm model evaluates part of the diagram.

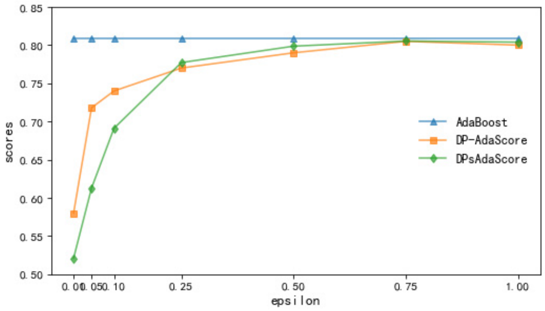


FIGURE 8. Comparison of CART-DPsAdaBoost with AdaBoost and DP-AdaBoost on the adult data set.

8 numerical attributes, 33 discrete attributes, and a total of 199,523 records. Digit Recognizer's 42,000 records form a large-scale, high-dimensional dataset concerned with identifying handwritten numbers. It contains 784 attributes with 10 label categories as shown in Table 3 below. Through each iteration, one of the label classes is selected as a positive sample, and the rest are merged into a negative sample. A down-sampling strategy is adopted, and the average value is used as the final result.

B. EXPERIMENTAL DESIGN

To test the validity and performance of the CART-DPsAdaBoost algorithm, we conducted several sets of experiments: 1.) The difference between added noise and no added noise; 2.) The impact of the privacy budget on the accuracy of the classifier; 3.) The relationship between the size of the training set and the accuracy of the classification; 4.) The evaluation of the overall performance of the algorithm; 5.) A comparison against the DP-AdaBoost algorithm [31]. Apart from the second experiment, the total privacy budget is $\varepsilon = 1$, and the number of features randomly selected by nodes in the random subspace algorithm is $k = 5$.

In order to evaluate the impact of the privacy protection strategy on the data classification performance of the algorithm, a Gini index-based single-layer decision tree algorithm is used on the data set, and its level of accuracy is taken as a baseline. Then multiple sets of experiments are run. Since the algorithm introduces two kinds of randomness and the probability of adding noise is done by the Laplace mechanism, the values of n Laplace noise levels are averaged to ensure differential privacy ($n = 100$ in this experiment). Finally, each group of experiments was performed 5 times, with the average value taken as the final result.

C. ANALYSIS OF EXPERIMENTAL

1) EXPERIMENT

As shown in Fig. 5(a-c), we set $\varepsilon = 0.05$, 0.1, 0.25, 0.5, 0.75, 1, and took the accuracy of the CART stump classifier as the baseline to compare the results of the classification models with and without noise on the three data sets. When the value of ε is small, the accuracy of the ensemble classifier is low. As the value of ε increases, the classification accuracy rate gradually increases. This is because the larger the degree privacy budget ε, the less noise is added, and so the smaller the impact on the availability of data, but the lower the of privacy protection. In Fig. 5(a), it can be seen from the first inflection point on the curve that about 10 base classifiers are sufficient to essentially reach the optimal model, and the accuracy rate is stable after about 30 base classifiers, at the second inflection point. It can be seen from the experiment that when $\varepsilon = 1$, $T = 51$, the accuracy of the model on the

Adult Data Set reaches a peak of 0.8156953503050705. In the Census Income large-scale dataset, the model stabilized after about 40 base classifiers. The peak parameter values are $\varepsilon = 1$, $T = 91$, when the model accuracy rate is 0.8629021667760998. For the Digit Recognizer dataset, due to the larger amount of data and higher dimensionality, more privacy budgets are allocated. While protecting data privacy, the algorithm performs better on large-scale, high-dimensional data sets, indicating that the algorithm has good data scalability.

Fig. 6(a-c) show the relationship between dataset size and model accuracy. The accuracy of the CART-DPsAdaBoost algorithm with differential privacy increases with the size of the training set, and rises from 50% to 81% in the Adult

TABLE 3. Category label for digit recognizer datasets.

Category label	0	1	2	3	4	5	6	7	8	9
Number of samples	4132	4684	4177	4351	4072	3795	4137	4401	4063	4188

TABLE 4. How *Precision*, *Recall,* and *F1Scores* vary with ε.

Value of ε	Adult			Census Income			Digit Recognizer		
	Precision	Recall	F1Score	Precision	Recall	F1Score	Precision	Recall	F1Score
0.05	0.62938	0.62628	0.62783	0.62609	0.65019	0.63791	0.65387	0.69932	0.67583
0.1	0.69987	0.71540	0.70755	0.71432	0.74330	0.72852	0.80197	0.81742	0.80962
0.25	0.79039	0.76344	0.77668	0.80251	0.83068	0.81635	0.91271	0.91959	0.91614
0.5	0.82904	0.78069	0.80414	0.83594	0.84681	0.84134	0.92452	0.94388	0.93410
0.75	0.83610	0.78357	0.80898	0.84145	0.84473	0.84308	0.92232	0.94472	0.93338
1	0.84182	0.78685	0.81341	0.84571	0.84395	0.84483	0.92981	0.94304	0.93638

TABLE 5. AUC performance comparison.

Data Set	$\varepsilon = 0.05$	$\varepsilon = 0.1$	$\varepsilon = 0.25$	$\varepsilon = 0.5$	$\varepsilon = 0.75$	$\varepsilon = 1$
Adult	0.64267	0.72996	0.84343	0.86289	0.87287	0.89058
Census Income	0.68524	0.79946	0.89804	0.91971	0.92414	0.92514
Digit Recognizer	0.79509	0.93989	0.99033	0.99468	0.99567	0.99595

Data Set. The larger the privacy budget, the closer the accuracy of the model is to the accuracy of the unnoised model, and the 800 sample sets have reached the upper limit of the model. For the large-scale dataset of Census Income, this rises from 35% to about 85%. Due to the increase in the number of features, the model finds it easier to learn, and by about 400 samples it has reached the limit of the model. On the Digit Recognizer high-dimensional dataset, the T = 10 accuracy rate is 0.9440154440015444, and the model size stabilizes when the sample size reaches about 400. This experiment again validates the classification effectiveness of the algorithm for large-scale, high-dimensional data sets. With experimental settings of $\varepsilon = 0.05, 0.1, 0.25, 0.5, 0.75, 1$, $T = 10$, the proposed algorithm is compared with the DP-AdaBoost algorithm, where the data under the same conditions of the DP-AdaBoost algorithm are provided by literature [31]. Under different privacy budgets, the experimental results of the two models are shown in Fig. 8.

2) EVALUATION

Table 4 and Table 5 show the performance of the algorithm at different data protection levels across the three data sets. The model was evaluated by the value of the *F1Score* and the value of the AUC of the ROC curve. In Table 4, as the privacy budget increases, the availability of the algorithm improves. Of course, the limitation of the classification and regression tree height and the probability of adding noise lead to noise mean deviation, and there is a phenomenon that the *F1Score* fluctuates when the privacy protection level is raised. In the Adult Data Set in Table 4, after $\varepsilon = 0.25$, the *F1Score* stabilizes, and Fig. 7(a) shows the curve of *precision*, *recall*, and *F1Score* as a function of ε. In the Census Income and Digit Recognizer datasets, after $\varepsilon = 0.5$, the *F1Score* tends to stabilize. This experiment shows that the algorithm adds less noise in the high-dimensional dataset to make the model performance stable and the algorithm has better usability. As shown in Table 5, as the level of privacy protection is higher, the generalization performance of the model generally shows better performance. The experimental results of the Adult Data Set show that the AUC change is the most unstable, as shown in Fig. 7(b) for the ROC at $\varepsilon = 1$. As the number of features in the Adult Data Set is small, adjusting the privacy budget causes a greater disturbance to the AUC value. The *F1Score* results in Table 4 also show that the lower the feature dimension, the greater the impact on the accuracy of the model. Hence, the algorithm performs better on the other two data sets that have a large sample size and high feature dimensionality.

3) COMPARISONS

Experimental setting $\varepsilon = 0.05, 0.1, 0.25, 0.5, 0.75, 1$, $T = 10$, compare the proposed algorithm with DP-Adaboost algorithm, the data under the same conditions of DP-Adaboost algorithm is provided by literature [31].

Compared with the original AdaBoost, and DP-AdaBoost algorithms, it can be seen that the proposed algorithm maintains a good classification accuracy under the differential privacy protection strategy. In the construction of the decision tree, part of the privacy budget is consumed in processing the continuous attributes, but there is no need to discretize or preprocess these attributes, which improves the efficiency of the classification task. The accuracy of the algorithm fluctuates within an acceptable range after the introduction of the differential privacy policy. For the Adult Data Set, it can be seen from Fig. 8 that when the privacy budget ε is between 0.05 and 0.25, the classification accuracy of the proposed algorithm is lower than the DP-AdaBoost algorithm. This is because the smaller the value of ε at this time, the larger the amount of differential privacy noise that is added to the leaf node instances and classes by the AdaBoost algorithm after the weak classifier is generated. Also, during the iterative process, the quality of the generated results is finally affected.

After $\varepsilon = 0.25$, with the increase of the privacy budget, the introduction of two types of randomness schemes, and the use of CART boosting allows the advantages of the algorithm to become more apparent. When $\varepsilon \geq 0.5$ the algorithm maintains a better classification accuracy. And the algorithm reduces the complexity of the model while avoiding the effect of the depth of the tree and the privacy budget allocation on the hierarchical equalization strategy. The reduction in the number of nodes improves the allocation of the privacy budget, the required noise is sharply reduced, and has little impact on the accuracy of the final classification model.

VI. CONCLUSION

Based on existing decision tree algorithms, differential privacy is applied to an ensemble learning process. The AdaBoost classification algorithm under differential privacy protection was studied with the aim of enabling the protection of private information and improving the classification accuracy of the model. The algorithm used an improved CART classification tree to introduce two types of random schemes: sample perturbation and feature perturbation, in the process of constructing the base learner. In addition, our experiments showed that when using the exponential mechanism and the Gini index to deal with continuous attributes, and using discrete attributes to construct the CART decision tree, the classification accuracy rate is not significantly reduced. Also, the utilization rate of the privacy protection budget and the efficiency of algorithm execution are improved, so that it can effectively deal with largescale and high-dimensional data classification problems. In addition, the proposed approach is well suited, due to its scalability, to being applied in a big data environment.

The methods proposed in this paper achieved good results. However, there are still some shortcomings: 1.) The privacy protection level is based on a quantitative analysis of the privacy budget ε, but there is no agreed standard for setting its level in practical algorithms and applications. 2.) Differential privacy usually assumes that the records in the data set are independent of each other, but in practice, the records may be related, which increases the risk of privacy leakage. In future work, the algorithm will be further improved to optimize generalization performance. At the same time, the AdaBoost classification model will be generated by each sample set and integrated under differential privacy constraints. The classification accuracy and generalization performance are studied in the final fusion mode. In addition, a single privacy policy cannot meet the needs of personalized differential privacy. How to evaluate the level of privacy protection for different users is another task for future work.

Building upon the research described in this paper, we will design a variety of machine learning algorithms based on differential privacy, and then propose corresponding machine learning privacy protection data release mutual feed- back mechanisms to solve some important theoretical issues in this area. Also, we plan to develop a set of analysis and research methods for privacy protection data model publishing, applicable to specific machine learning algorithms. We hope that these results will provide important theoretical guidance and technical support for the engineering and implementation of privacy protection in data distribution systems.

REFERENCES

[1] M. Abadi, A. Chu, I. Goodfellow, H. B. Mcmahan, I. Mironov, K. Talwar, and L. Zhang, “Deep learning with differential privacy,” in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2016, pp. 308–318.

[2] R. Bassily, A. Smith, T. Steinke, and J. Ullman, “More general queries and less generalization error in adaptive data analysis,” 2015, *arXiv:1503.04843*. [Online]. Available: http://arxiv.org/abs/1503.04843

[3] A. Blum, C. Dwork, F. McSherry, and K. Nissim, “Practical privacy: The SuLQ framework,” in *Proc. 24th ACM SIGMOD-SIGACT-SIGART Symp. Princ. Database Syst.*, 2005, pp. 128–138.

[4] P. Cuff and L. Yu, “Differential privacy as a mutual information constraint,” in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2016, pp. 43–54.

[5] C. Dwork, *Differential Privacy* (Lecture Notes in Computer Science), vol. 4052. Berlin, Germany: Springer-Verlag, 2006, pp. 1–12.

[6] C. Dwork, ”Differential privacy: A survey of results,” in *Theory and Applications of Models of Computation* (Lecture Notes in Computer Science), vol. 4978. Berlin, Germany: Springer-Verlag, 2008, pp. 1–19.

[7] C. Dwork, V. Feldman, M. Hardt, T. Pitassi, O. Reingold, and A. Roth, “The reusable holdout: Preserving validity in adaptive data analysis,” in *Proc. Annu. ACM Symp. Theory Comput.*, 2015, pp. 117–126.

[8] M. Shoaran, A. Thomo, and J. H. Weber, *Differential Privacy in Practice* (Lecture Notes in Computer Science), vol. 7482. Springer-Verlag, 2012, pp. 14–24.

[9] C. Dwork and A. Roth, “The algorithmic foundations of differential privacy,” *Found. Trends Theor. Comput. Sci.*, vol. 9, nos. 3–4, pp. 211–407, 2013.

[10] S. Fletcher and M. Z. Islam, “Decision tree classification with differential privacy: A survey,” *ACM Comput. Surv.*, vol. 52, no. 4, pp. 1–33, Aug. 2019.

[11] A. Friedman and A. Schuster, “Data mining with differential privacy,” in *Proc. 16th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2010, pp. 493–502.

[12] Q. Geng, W. Ding, R. Guo, and S. Kumar, “Optimal noise-adding mechanism in additive differential privacy,” *IEEE Trans. Inf. Theory.*, vol. 62, no. 2, pp. 925–951, Feb. 2016.

[13] P. Goswami and S. Madan, “Privacy preserving data publishing and data anonymization approaches: A review,” in *Proc. Int. Conf. Comput., Commun. Autom. (ICCCA)*, May 2017, pp. 139–142.

[14] M. Hairong, D. Liping, S. Yuning, and L. Guoqing, "Diffprfs: A differential privacy protection algorithm for random forests," *J. Commun., China*, vol. 37, no. 9, pp. 175–182, 2016.

[15] Z. Ji, Z. C. Lipton, and C. Elkan, "Differential privacy and machine learning: A survey and review," 2014, *arXiv:1412.7584*. [Online]. Available: http://arxiv.org/abs/1412.7584

[16] P. Kairouz, S. Oh, and P. Viswanath, "The composition theorem for differential privacy," in *Proc. Int. Conf. Mach. Learn. (ICML)*, vol. 2, Jul. 2015, pp. 1376–1385.

[17] W. Y. Loh, *Classification and Regression Tree Methods*. Hoboken, NJ, USA: Wiley, 2008, doi: 10.1002/9780470061572.eqr492.

[18] F. D. McSherry, "Privacy integrated queries: An extensible platform for privacy-preserving data analysis," *Commun ACM.*, vol. 53, no. 9, pp. 89–97, Sep. 2010.

[19] I. Mironov, "Rényi differential privacy," in *Proc. IEEE 30th Comput. Secur. Found. Symp. (CSF)*, Aug. 2017, pp. 263–275.

[20] K. Mivule, C. Turner, and S. Y. Ji, "Towards a differential privacy and utility preserving machine learning classifier," in *Proc. Conf. Complex Adapt. Syst. (CAS)*, vol. 12, 2012, pp. 176–181.

[21] N. Mohammed, R. Chen, B. C. M. Fung, and P. S. Yu, "Differentially private data release for data mining," in *Proc. 17th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2011, pp. 493–501.

[22] A. S. More and D. P. Rana, "Review of random forest classification techniques to resolve data imbalance," in *Proc. 1st Int. Conf. Intell. Syst. Inf. Manage. (ICISIM)*, Oct. 2017, pp. 72–78.

[23] A. Patil and S. Singh, "Differential private random forest," in *Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI)*, Sep. 2014, pp. 2623–2630.

[24] X. Ping, Z. Tianqing, and W. Xiaofeng, "Differential privacy protection and its application," *J. Comput. Sci., China*, vol. 37, no. 1, pp. 101–122, 2014.

[25] N. Polatidis, C. K. Georgiadis, E. Pimenidis, and H. Mouratidis, "Privacy-preserving collaborative recommendations based on random perturbations," *Expert Syst. Appl.*, vol. 71, pp. 18–25, Apr. 2017.

[26] Y. Ren, L. Zhang, and P. N. Suganthan, "Ensemble classification and regression-recent developments, applications and future directions [review article]," *IEEE Comput. Intell. Mag.*, vol. 11, no. 1, pp. 41–53, Feb. 2016.

[27] A. D. Sarwate and K. Chaudhuri, "Signal processing and machine learning with differential privacy: Algorithms and challenges for continuous data," *IEEE Signal Process. Mag.*, vol. 30, no. 5, pp. 86–94, Sep. 2013.

[28] R. E. Schapire, "Explaining adaboost," in *Empir. Inference: Festschrift in Honor of Vladimir N. Vapnik*. Berlin, Germany: Springer-Verlag, Jan. 2013, pp. 37–52.

[29] R. E. Schapire and Y. Freund, *Boosting: Foundations and Algorithms*. Cambridge, MA, USA: MIT Press, 2013, doi: 10.1108/03684921311295547.

[30] R. Shokri and V. Shmatikov, "Privacy-preserving deep learning," in *Proc. 53rd Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, Sep. 2015, pp. 1310–1321.

[31] S. Siqian, "Research on classification algorithm of differential privacy protection," M.S. thesis, Nanjing Aerosp. Univ., Nanjing, China, 2017.

[32] J. Soria-Comas, J. Domingo-Ferrer, D. Sanchez, and D. Megias, "Individual differential privacy: A utility-preserving formulation of differential privacy guarantees," *IEEE Trans. Inf. Forensics Security*, vol. 12, no. 6, pp. 1418–1429, Jun. 2017.

[33] S. Truex, L. Liu, M. E. Gursoy, and L. Yu, "Privacy-preserving inductive learning with decision trees," in *Proc. IEEE Int. Congr. Big Data (BigData Congress)*, Jun. 2017, pp. 57–64.

[34] A. Wang, C. Wang, M. Bi, and J. Xu, *A Review of Privacy-Preserving Machine Learning Classification* (Lecture Notes in Computer Science), vol. 11066. Cham, Switzerland: Springer-Verlag, 2018, pp. 671–682.

[35] X. Wang, P. Li, H. Xu, Z. Xu, and Y. Zhang, "Analysis and research based on differential privacy protection related algorithms," in *Proc. 9th Int. Symp. Parallel Architectures, Algorithms Program. (PAAP)*, Dec. 2018, pp. 126–133.

[36] A. Waseda and R. Nojima, *Analyzing Randomized Response Mechanisms Under Differential Privacy* (Lecture Notes in Computer Science), vol. 9866. Cham, Switzerland: Springer-Verlag, 2016, pp. 271–282.

[37] T. Xiang, Y. Li, X. Li, S. Zhong, and S. Yu, "Collaborative ensemble learning under differential privacy," *Web Intell.*, vol. 16, no. 1, pp. 73–87, Mar. 2018.

[38] L. Yuan and M. Y. Ji, "Research on unbalanced data set classification algorithm based on random balanced sampling," *Natural Sci. J. Hainan Univ., China*, vol. 5, no. 3, pp. 33–38, 2017.

[39] J. Yan and S. Han, "Classifying imbalanced data sets by a novel RE-sample and cost-sensitive stacked generalization method," *Math. Problems Eng.*, vol. 2018, Jan. 2018, Art. no. 5036710.

[40] Z. Yuan, D. Bao, Z. Chen, and M. Liu, "Integrated transfer learning algorithm using multi-source TrAdaBoost for unbalanced samples classification," in *Proc. Int. Conf. Comput. Intell. Inf. Syst. (CIIS)*, Apr. 2017, pp. 188–195.

[41] J. Zhang, X. Xiao, and X. Xie, "Privtree: A differentially private algorithm for hierarchical decompositions," in *Proc. Int. Conf. Manage. Data*, Jun. 2016, pp. 155–170.

[42] T. Zhu, G. Li, W. Zhou, and P. S. Yu, "Differentially private data publishing and analysis: A survey," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 8, pp. 1619–1638, Aug. 2017.

[43] T. Zhu, P. Xiong, Y. Xiang, and W. Zhou, "An effective deferentially private data releasing algorithm for decision tree," in *Proc. 12th IEEE Int. Conf. Trust, Secur. Privacy Comput. Commun.*, Jul. 2013, pp. 388–395.

[44] F. McSherry and K. Talwar, "Mechanism design via differential privacy," in *Proc. 48th Annu. IEEE Symp. Found. Comput. Sci. (FOCS)*, Oct. 2007, pp. 94–103.

JUNJIE JIA was born in Lanzhou, China, in 1974. He received the M.S. degree in computer application technology from Northwest Normal University, Lanzhou, in 2005, and the Ph.D. degree from Chang'an University, Xi'an, China, in 2009. He is currently an Associate Professor with Northwest Normal University. His current research interests include data security and privacy protection.

WANYONG QIU was born in Qingyang, China, in 1993. He received the B.S. degree from the School of Electronics and Information Science, Lanzhou City University, China, in 2017. He is currently pursuing the M.S. degree with the School of Computer Science and Engineering, Northwest Normal University, Lanzhou, China. His main research interests include data mining, machine learning, and privacy protection.