

RANDOM GREEDY ALGORITHM-BASED HORIZONTAL FEDERATED GRADIENT BOOSTED TREE OPTIMIZATION METHOD

TECHNICAL FIELD

5 **[0001]** The present application relates to the technical field of federated learning, in particular to a horizontal federated Gradient Boosting Decision Tree optimization method based on a random greedy algorithm.

BACKGROUND

10 **[0002]** Federated learning is a machine learning framework, which can effectively help multiple organizations to model data usage and machine learning while meeting the requirements of user privacy protection, data security and government regulations, so that participants can jointly implement modeling on the basis of unshared data, which can technically break the data island and realize AI collaboration. Under this framework, the problem of collaboration among different data owners without exchanging data is solved by
15 designing virtual models. A virtual model is the best model for all parties to aggregate data together. Each region serves the local target according to the model. Federated learning requires that the modeling results should be infinitely close to the traditional model, that is, the data of multiple data owners are gathered in one place for modeling. Under the federated mechanism, each participant has the same identity and status, and a data sharing strategy can
20 be established. A greedy algorithm is a simpler and faster design technology for some optimal solutions. The characteristic of the greedy algorithm is that it is carried out step by step, often based on the current situation, the optimal selection is made according to an optimization measure, without considering all possible overall situations, which saves a lot of time that must be spent to exhaust all possibilities to find the optimal solution. The greedy algorithm adopts
25 the top-down and iterative method to make successive greedy choices. Every time a greedy choice is made, the required problem is simplified to a smaller sub-problem. Through every

greedy choice, an optimal solution of the problem can be obtained. Although the local optimal solution must be obtained in every step, the global solution generated therefrom is sometimes not necessarily optimal, so the greedy algorithm should not backtrack.

[0003] However, the existing horizontal federated Gradient Boosting Decision Tree
5 algorithm requires each participant and coordinator to frequently transmit histogram
information, which requires high network bandwidth of the coordinator, and the training
efficiency is easily affected by the network stability. Moreover, because the transmitted
histogram information contains user information, there is a risk of leaking user privacy. After
10 introducing privacy protection solutions such as multi-party secure computing, homomorphic
encryption and secret sharing, the possibility of user privacy leakage can be reduced, but the
local computing burden will be increased and the training efficiency will be reduced.

SUMMARY

[0004] The purpose of the present application is to provide a horizontal federated Gradient
Boosting Decision Tree optimization method based on a random greedy algorithm, which aims
15 to solve the problem that the existing horizontal federated Gradient Boosting Decision Tree
algorithm proposed in the above background technology requires all participants and
coordinators to frequently transmit histogram information, which has high requirements on the
network bandwidth of the coordinators, and the training efficiency is easily affected by the
network stability, and because the transmitted histogram information contains user information,
20 there is a risk of leaking user privacy. After introducing privacy protection solutions such as
multi-party secure computing, homomorphic encryption and secret sharing, the possibility of
user privacy leakage can be reduced, but the local computing burden will be increased and the
training efficiency will be reduced.

[0005] In order to achieve the above objectives, the present application provides the
25 following technical solution: a horizontal federated Gradient Boosting Decision Tree
optimization method based on a random greedy algorithm includes the following steps:

[0006] Step 1: a coordinator setting relevant parameters of a Gradient Boosting Decision
Tree model, including a maximum number of decision trees T , a maximum depth of trees L , an
initial predicted value base, etc., and sending the relevant parameters to respective participants

p_i ;

[0007] Step 2: letting a tree counter $t=1$;

[0008] Step 3: for each participant p_i , initializing a training target of a k^{th} tree $y_k = y_{k-1} - \hat{y}_{k-1}$; wherein $y_0 = y, \hat{y}_0 = \text{base}$;

5 **[0009]** Step 4: letting a tree layer counter $l=1$;

[0010] Step 5: letting a node counter of a current layer $n=1$;

[0011] Step 6: for each participant p_i , determining a segmentation point of a local current node n according to the data of the current node and an optimal segmentation point algorithm and sending the segmentation point information to the coordinator;

10 **[0012]** Step 7: the coordinator counting the segmentation point information of all participants, and determining a segmentation feature f and a segmentation value v according to an epsilon- greedy algorithm;

[0013] Step 8, the coordinator sending the finally determined segmentation information, including the determined segmentation feature f and segmentation value v , to respective
15 participants;

[0014] Step 9: each participant segmenting a data set of the current node according to the segmentation feature f and the segmentation value v , and distributing new segmentation data to child nodes;

[0015] Step 10: letting $n=n+1$, and continuing with the Step 3 if n is less than or equal to a
20 maximum number of nodes in the current layer; otherwise, proceeding to a next step;

[0016] Step 11: resetting the node information of the current layer according to the child nodes of a node of a l^{th} layer, so that $l=l+1$, and continuing with the Step 5 if l is less than or equal to the maximum tree depth L ; otherwise, proceeding to a next step;

[0017] Step 12: letting $t=t+1$, and continuing with the Step 3 if t is greater than or equal to
25 the maximum number of decision trees T ; otherwise, ending.

[0018] Preferably, the optimal segmentation point algorithm in the Step 6:

[0019] I, determines a segmentation objective function, including but not limited to the following objective functions:

[0020] information gain: information gain is the most commonly used index to measure a
30 purity of a sample set; assuming that there are K types of samples in a node sample set D , in

which a proportion of a k^{th} type of samples is p_k , an information entropy of D is defined as:

[0021]
$$\text{Ent}(D) = -\sum_{k=1}^{|K|} p_k \log_2 p_k$$

[0022] assuming that the node is segmented into V possible values according to an attribute a , the information gain is defined as:

5 **[0023]**
$$\text{Gain}(D, a) = \text{Ent}(D) - \sum_{v=1}^V \frac{|D^v|}{|D|} \text{Ent}(D^v);$$

[0024] information gain rate:

[0025]
$$\text{Gain_ratio}(D, a) = \frac{\text{Gain}(D, a)}{\text{IV}(a)}$$

[0026] where

[0027]
$$\text{IV}(a) = -\sum_{v=1}^V \frac{|D^v|}{|D|} \log_2 \frac{|D^v|}{|D|}$$

10 **[0028]** a Gini coefficient:

[0029]
$$\text{Gini}(D) = \sum_{k=1}^{|K|} \sum_{k' \neq k} p_k p_{k'}$$

[0030]
$$\text{Gini_index}(D, a) = \sum_{v=1}^V \frac{|D^v|}{|D|} \text{Gini}(D^v)$$

[0031] a structural coefficient:

[0032]
$$\text{Gain} = \frac{1}{2} \left(\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{G^2}{H + \lambda} \right) - \gamma$$

15 **[0033]** where G_L is a sum of first-order gradients of the data set divided into a left node according to the segmentation point, H_L is a sum of second-order gradients of the data set of the left node, and G_R and H_R are sums of the gradient information of a corresponding right node, γ is a tree model complexity penalty term and λ is a second-order regular term;

[0034] II, determines a candidate list of segmentation values: determining the list of
20 segmentation values according to the data distribution of the current node, wherein the segmentation values comprise segmentation features and segmentation feature values; the list of segmentation value is determined according to the following method:

[0035] all values of all features in the data set;

[0036] determining a discrete segmentation point according to a value range of each feature in the data set; wherein the selection of the segmentation points can be evenly distributed within the value range according to the distribution of data, and the amount of data uniformly reflected between the segmentation points is approximately equal or a sum of the second-order gradients is approximately equal;

[0037] traversing the candidate list of segmentation values to find the segmentation point that makes the objective function optimal.

[0038] Preferably, the Epsilon greedy algorithm in the Step 7 includes: for the node n, each participant sending the node segmentation point information to the coordinator, including a segmentation feature f_i , a segmentation value v_i , a number of node samples N_i and a local objective function gain g_i , where i represents respective participants;

[0039] according to the segmentation information of each participant and based on a maximum number principle, the coordinator determining an optimal segmentation feature $f_{\max}$, where X is a random number evenly distributed among $[0,1]$ and randomly sampling X to obtain x ; if $x \leq \epsilon$, randomly selecting one of the segmentation features of each participant as a global segmentation feature; otherwise, selecting $f_{\max}$ as the global segmentation feature;

[0040] each participant recalculating the segmentation information according to the global segmentation feature and sending the segmentation information to the coordinator;

[0041] the coordinator determining a global segmentation value according to the following formula: if the total number of participants is P ,

$$\mathbf{v} = \sum_{i=1}^P \frac{N_i g_i}{\sum_{j=1}^P N_j g_j} v_i;$$

[0043] distributing the segmentation value to each participant to perform node segmentation.

[0044] Preferably, the horizontal federated learning is a distributed structure of federated learning, in which each distributed node has the same data feature and different sample spaces.

[0045] Preferably, the Gradient Boosting Decision Tree algorithm is an integrated model based on gradient boosting and decision tree.

[0046] Preferably, the decision tree is a basic model of a Gradient Boosting Decision Tree

model, and a prediction direction of a sample is judged at the node by given features based on a tree structure.

[0047] Preferably, the segmentation point is a segmentation position of non-leaf nodes in the decision tree for data segmentation.

5 **[0048]** Preferably, the histogram is statistical information representing the first-order gradient and the second-order gradient in node data.

[0049] Preferably, an input device can be one or more of data terminals such as computers and mobile phones or mobile terminals.

10 **[0050]** Preferably, the input device comprises a processor, and when executed by the processor, the algorithm of any one of steps 1 to 12 is implemented.

[0051] Compared with the prior art, the present application has the following beneficial effects: the horizontal federated Gradient Boosting Decision Tree optimization method based on a random greedy algorithm setting relevant parameters of a Gradient Boosting Decision Tree model, including but not limited to a maximum number of decision trees T , a maximum
15 depth of trees L , an initial predicted value base, etc., and sending the relevant parameters to respective participants p_i ; letting a tree counter $t=1$; for each participant p_i , letting a tree layer counter $l=1$; letting a node counter of a current layer $n=1$; for each participant p_i , determining a segmentation point of a local current node n according to the data of the current node and an optimal segmentation point algorithm and sending the segmentation point information to the
20 coordinator; the coordinator counting the segmentation point information of all participants, and determining a segmentation feature f and a segmentation value v according to an epsilon-greedy algorithm; the coordinator sending the finally determined segmentation information, including but not limited to the determined segmentation feature f and segmentation value v , to respective participants; each participant segmenting a data set of the current node according
25 to the segmentation feature f and the segmentation value v , and distributing new segmentation data to child nodes; letting $n=n+1$, and continuing with the Step 6 if n is less than or equal to a maximum number of nodes in the current layer; otherwise, proceeding to a next step; resetting the node information of the current layer according to the child nodes of a node of a l^{th} layer, so that $l=l+1$, and continuing with the Step 5 if l is less than or equal to the maximum tree depth
30 L ; otherwise, proceeding to a next step; letting $t=t+1$, and continuing with the Step 3 if t is

greater than or equal to the maximum number of decision trees T ; otherwise, ending. The supported horizontal federated learning includes participants and coordinators, wherein the participants have local data, the coordinators do not have any data, and the center for information aggregation of participants; participants calculate histograms separately and send them to the coordinators; after summarizing all histogram information, the coordinators find the optimal segmentation points according to the greedy algorithm, and then share them with respective participants to facilitate work with internal algorithms.

BRIEF DESCRIPTION OF DRAWINGS

[0052] In order to explain the technical solution in the embodiments of the present application more clearly, the drawings necessary for the description of the embodiments or the prior art the following will be briefly introduced. Obviously, the drawings in the following description are only some embodiments of the present application, and for those skilled in the art, other drawings can be obtained according to these drawings without creative efforts.

[0053] FIG. 1 is a schematic diagram of the horizontal federated Gradient Boosting Decision Tree optimization method based on a random greedy algorithm of the present application;

[0054] FIG. 2 is a schematic diagram of the steps of the horizontal federated Gradient Boosting Decision Tree optimization method based on a random greedy algorithm of the present application;

[0055] FIG. 3 is a schematic diagram for judging the horizontal federated Gradient Boosting Decision Tree optimization method based on a random greedy algorithm of the present application.

DESCRIPTION OF EMBODIMENTS

[0056] Next, the technical solutions in the embodiments of the present application will be clearly and completely described with reference to the drawings in the embodiments of the present application. Obviously, the described embodiments are only part of, not all of the embodiments of the present application. Based on the embodiments in the present application,

all other embodiments obtained by those skilled in the art without creative work are within the scope of the present application.

[0057] Referring to FIG. 1-3, the present application provides a technical solution: a horizontal federated Gradient Boosting Decision Tree optimization method based on a random greedy algorithm, which includes the following steps:

[0058] Step 1: a coordinator setting relevant parameters of a Gradient Boosting Decision Tree model, including a maximum number of decision trees T , a maximum depth of trees L , an initial predicted value base, etc., and sending the relevant parameters to respective participants p_i ;

[0059] Step 2: letting a tree counter $t=1$;

[0060] Step 3: for each participant p_i , initializing a training target of a k^{th} tree $y_k = y_{k-1} - \hat{y}_{k-1}$; wherein $y_0 = y$, $\hat{y}_0 = \text{base}$;

[0061] Step 4: letting a tree layer counter $l=1$;

[0062] Step 5: letting a node counter of a current layer $n=1$;

[0063] Step 6: for each participant p_i , determining a segmentation point of a local current node n according to the data of the current node and an optimal segmentation point algorithm and sending the segmentation point information to the coordinator;

[0064] Step 7: the coordinator counting the segmentation point information of all participants, and determining a segmentation feature f and a segmentation value v according to an epsilon- greedy algorithm;

[0065] Step 8, the coordinator sending the finally determined segmentation information, including the determined segmentation feature f and segmentation value v , to respective participants;

[0066] Step 9: each participant segmenting a data set of the current node according to the segmentation feature f and the segmentation value v , and distributing new segmentation data to child nodes;

[0067] Step 10: letting $n=n+1$, and continuing with the Step 3 if n is less than or equal to a maximum number of nodes in the current layer; otherwise, proceeding to a next step;

[0068] Step 11: resetting the node information of the current layer according to the child nodes of a node of a l^{th} layer, so that $l=l+1$, and continuing with the Step 5 if l is less than or

equal to the maximum tree depth L ; otherwise, proceeding to a next step;

[0069] Step 12: letting $t=t+1$, and continuing with the Step 3 if t is greater than or equal to the maximum number of decision trees T ; otherwise, ending.

[0070] Furthermore, the optimal segmentation point algorithm in the Step 6:

5 **[0071]** I, determines a segmentation objective function, including but not limited to the following objective functions:

[0072] information gain: information gain is the most commonly used index to measure a purity of a sample set; assuming that there are K types of samples in a node sample set D , in which a proportion of a k^{th} type of samples is p_k , an information entropy of D is defined as:

10 **[0073]**
$$\text{Ent}(D) = -\sum_{k=1}^{|K|} p_k \log_2 p_k$$

[0074] assuming that the node is segmented into V possible values according to an attribute a , the information gain is defined as:

[0075]
$$\text{Gain}(D, a) = \text{Ent}(D) - \sum_{v=1}^V \frac{|D^v|}{|D|} \text{Ent}(D^v);$$

[0076] information gain rate:

15 **[0077]**
$$\text{Gain_ratio}(D, a) = \frac{\text{Gain}(D, a)}{\text{IV}(a)}$$

[0078] where

[0079]
$$\text{IV}(a) = -\sum_{v=1}^V \frac{|D^v|}{|D|} \log_2 \frac{|D^v|}{|D|}$$

[0080] a Gini coefficient:

[0081]
$$\text{Gini}(D) = \sum_{k=1}^{|K|} \sum_{k' \neq k} p_k p_{k'}$$

20 **[0082]**
$$\text{Gini_index}(D, a) = \sum_{v=1}^V \frac{|D^v|}{|D|} \text{Gini}(D^v)$$

[0083] a structural coefficient:

[0084]
$$\text{Gain} = \frac{1}{2} \left(\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{G^2}{H + \lambda} \right) - \gamma$$

[0085] where G_L is a sum of first-order gradients of the data set divided into a left node according to the segmentation point, H_L is a sum of second-order gradients of the data set of

the left node, and G_R and H_R are sums of the gradient information of a corresponding right node, γ is a tree model complexity penalty term and λ is a second-order regular term;

[0086] II, determines a candidate list of segmentation values: determining the list of segmentation values according to the data distribution of the current node, wherein the segmentation values comprise segmentation features and segmentation feature values; the list of segmentation value is determined according to the following method:

[0087] all values of all features in the data set;

[0088] determining a discrete segmentation point according to a value range of each feature in the data set; wherein the selection of the segmentation points can be evenly distributed within the value range according to the distribution of data, and the amount of data uniformly reflected between the segmentation points is approximately equal or a sum of the second-order gradients is approximately equal;

[0089] traversing the candidate list of segmentation values to find the segmentation point that makes the objective function optimal.

[0090] Furthermore, the Epsilon greedy algorithm in the Step 7 includes: for the node n ,

[0091] each participant sending the node segmentation point information to the coordinator, including a segmentation feature f_i , a segmentation value v_i , a number of node samples N_i and a local objective function gain g_i , where i represents respective participants;

[0092] according to the segmentation information of each participant and based on a maximum number principle, the coordinator determining an optimal segmentation feature $f_{\max}$,

[0093] where X is a random number evenly distributed among $[0,1]$ and randomly sampling X to obtain x ; if $x \leq \epsilon$, randomly selecting one of the segmentation features of each participant as a global segmentation feature; otherwise, selecting $f_{\max}$ as the global segmentation feature;

[0094] each participant recalculating the segmentation information according to the global segmentation feature and sending the segmentation information to the coordinator;

[0095] the coordinator determining a global segmentation value according to the following formula: if the total number of participants is P ,

[0096]
$$v = \sum_{i=1}^P \frac{N_i g_i}{\sum_{j=1}^P N_j g_j} v_i ;$$

[0097] distributing the segmentation value to each participant to perform node segmentation.

[0098] Furthermore, the horizontal federated learning is a distributed structure of federated learning, in which each distributed node has the same data feature and different sample spaces, which can facilitate comparison work.

[0099] Furthermore, the Gradient Boosting Decision Tree algorithm is an integrated model based on gradient boosting and decision tree, which can facilitate work.

[00100] Furthermore, the decision tree is a basic model of a Gradient Boosting Decision Tree model, and a prediction direction of a sample is judged at the node by given features based on a tree structure, which can facilitate prediction.

[00101] Furthermore, the segmentation point is a segmentation position of non-leaf nodes in the decision tree for data segmentation, which can facilitate segmentation.

[00102] Furthermore, the histogram is statistical information representing the first-order gradient and the second-order gradient in node data, which can facilitate more intuitive representation.

[00103] Furthermore, an input device can be one or more of data terminals such as computers and mobile phones or mobile terminals, which can facilitate data input.

[00104] Furthermore, the input device comprises a processor, and when executed by the processor, the algorithm of any one of steps 1 to 12 is implemented.

[00105]

[00106] The working principle is as below: Step 1: a coordinator setting relevant parameters of a Gradient Boosting Decision Tree model, including a maximum number of decision trees T , a maximum depth of trees L , an initial predicted value base, etc., and sending the relevant parameters to respective participants p_i ; Step 2: letting a tree counter $t=1$; Step 3: for each participant p_i , initializing a training target of a k^{th} tree $y_k = y_{k-1} - \hat{y}_{k-1}$; wherein $y_0 = y$, $\hat{y}_0 = \text{base}$; Step 4: letting a tree layer counter $l=1$; Step 5: letting a node counter of a current layer $n=1$; Step 6: for each participant p_i , determining a segmentation point of a local current node n according to the data of the current node and an optimal segmentation point algorithm

and sending the segmentation point information to the coordinator; I. I, determines a segmentation objective function, including but not limited to the following objective functions:

[00107] information gain: information gain is the most commonly used index to measure a purity of a sample set; assuming that there are K types of samples in a node sample set D, in

5 which a proportion of a k^{th} type of samples is p_k , an information entropy of D is defined as:

[00108]
$$\text{Ent}(D) = -\sum_{k=1}^{|K|} p_k \log_2 p_k$$

[00109] assuming that the node is segmented into V possible values according to an attribute a, the information gain is defined as:

[00110]
$$\text{Gain}(D, a) = \text{Ent}(D) - \sum_{v=1}^V \frac{|D^v|}{|D|} \text{Ent}(D^v);$$

10 **[00111]** information gain rate:

[00112]
$$\text{Gain_ratio}(D, a) = \frac{\text{Gain}(D, a)}{\text{IV}(a)}$$

[00113] where

[00114]
$$\text{IV}(a) = -\sum_{v=1}^V \frac{|D^v|}{|D|} \log_2 \frac{|D^v|}{|D|}$$

[00115] a Gini coefficient:

15 **[00116]**
$$\text{Gini}(D) = \sum_{k=1}^{|K|} \sum_{k' \neq k} p_k p_{k'}$$

$$\text{Gini_index}(D, a) = \sum_{v=1}^V \frac{|D^v|}{|D|} \text{Gini}(D^v)$$

[00117]

[00118] a structural coefficient:

[00119]
$$\text{Gain} = \frac{1}{2} \left(\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{G^2}{H + \lambda} \right) - \gamma$$

[00120] where G_L is a sum of first-order gradients of the data set divided into a left node

20 according to the segmentation point, H_L is a sum of second-order gradients of the data set of the left node, and G_R and H_R are sums of the gradient information of a corresponding right node, γ is a tree model complexity penalty term and λ is a second-order regular term;

[00121] II, determines a candidate list of segmentation values: determining the list of segmentation values according to the data distribution of the current node, wherein the segmentation values comprise segmentation features and segmentation feature values; the list of segmentation value is determined according to the following method:

5 **[00122]** all values of all features in the data set;

[00123] determining a discrete segmentation point according to a value range of each feature in the data set; wherein the selection of the segmentation points can be evenly distributed within the value range according to the distribution of data, and the amount of data uniformly reflected between the segmentation points is approximately equal or a sum of the second-order gradients is approximately equal;

[00124] traversing the candidate list of segmentation values to find the segmentation point that makes the objective function optimal; Step 7: the coordinator counting the segmentation point information of all participants, and determining a segmentation feature f and a segmentation value v according to an epsilon- greedy algorithm; for the node n ,

15 **[00125]** each participant sending the node segmentation point information to the coordinator, including a segmentation feature f_i , a segmentation value v_i , a number of node samples N_i and a local objective function gain g_i , where i represents respective participants;

[00126] according to the segmentation information of each participant and based on a maximum number principle, the coordinator determining an optimal segmentation feature $f_{\max}$,

[00127] where X is a random number evenly distributed among $[0,1]$ and randomly sampling X to obtain x ; if $x \leq \epsilon$, randomly selecting one of the segmentation features of each participant as a global segmentation feature; otherwise, selecting $f_{\max}$ as the global segmentation feature;

25 **[00128]** each participant recalculating the segmentation information according to the global segmentation feature and sending the segmentation information to the coordinator;

[00129] the coordinator determining a global segmentation value according to the following formula: if the total number of participants is P ,

$$\mathbf{[00130]} \quad v = \frac{\sum_{i=1}^P N_i g_i}{\sum_{j=1}^P N_j g_j} v_i ;$$

[00131] distributing the segmentation value to each participant to perform node segmentation; Step 8, the coordinator sending the finally determined segmentation information, including the determined segmentation feature f and segmentation value v , to respective participants; Step 9: each participant segmenting a data set of the current node according to the segmentation feature f and the segmentation value v , and distributing new segmentation data to child nodes; Step 10: letting $n=n+1$, and continuing with the Step 3 if n is less than or equal to a maximum number of nodes in the current layer; otherwise, proceeding to a next step; Step 11: resetting the node information of the current layer according to the child nodes of a node of a l^{th} layer, so that $l=l+1$, and continuing with the Step 5 if l is less than or equal to the maximum tree depth L ; otherwise, proceeding to a next step; Step 12: letting $t=t+1$, and continuing with the Step 3 if t is greater than or equal to the maximum number of decision trees T ; otherwise, ending. The coordinator sets relevant parameters of a Gradient Boosting Decision Tree model, including but not limited to a maximum number of decision trees, a maximum depth of trees, an initial predicted value, etc., and sending the relevant parameters to respective participants; the coordinator sends the finally determined segmentation information, including but not limited to the determined segmentation feature and segmentation value, to all participants, and each participant segments the data set of the current node according to the segmentation feature and segmentation value. The supported horizontal federated learning includes participants and coordinators, wherein the participants have local data, the coordinators do not have any data, and the center for information aggregation of participants; participants calculate histograms separately and send them to the coordinators; after summarizing all histogram information, the coordinators find the optimal segmentation points according to the greedy algorithm, and then share them with respective participants to facilitate work with internal algorithms.

Although the embodiments of the present application have been shown and described, it will be understood by those skilled in the art that many changes, modifications, substitutions and variations can be made to these embodiments without departing from the principles and spirit of the present application, the scope of which is defined by the appended claims and their equivalents.

CLAIMS

1. A horizontal federated Gradient Boosting Decision Tree optimization method based on a random greedy algorithm, comprising the following steps:

Step 1: a coordinator setting relevant parameters of a Gradient Boosting Decision Tree model, including a maximum number of decision trees T , a maximum depth of trees L , an initial predicted value base, etc., and sending the relevant parameters to respective participants p_i ;

Step 2: letting a tree counter $t=1$;

Step 3: for each participant p_i , initializing a training target of a k^{th} tree $y_k = y_{k-1} - \hat{y}_{k-1}$; wherein $y_0 = y, \hat{y}_0 = \text{base}$;

Step 4: letting a tree layer counter $l=1$;

Step 5: letting a node counter of a current layer $n=1$;

Step 6: for each participant p_i , determining a segmentation point of a local current node n according to the data of the current node and an optimal segmentation point algorithm and sending the segmentation point information to the coordinator;

Step 7: the coordinator counting the segmentation point information of all participants, and determining a segmentation feature f and a segmentation value v according to an epsilon- greedy algorithm;

Step 8, the coordinator sending the finally determined segmentation information, including the determined segmentation feature f and segmentation value v , to respective participants;

Step 9: each participant segmenting a data set of the current node according to the segmentation feature f and the segmentation value v , and distributing new segmentation data to child nodes;

Step 10: letting $n=n+1$, and continuing with the Step 3 if n is less than or equal to a maximum number of nodes in the current layer; otherwise, proceeding to a next step;

Step 11: resetting the node information of the current layer according to the child nodes of a node of a l^{th} layer, so that $l=l+1$, and continuing with the Step 5 if l is less than or equal to the maximum tree depth L ; otherwise, proceeding to a next step;

Step 12: letting $t=t+1$, and continuing with the Step 3 if t is greater than or equal to the maximum number of decision trees T ; otherwise, ending.

2. The horizontal federated Gradient Boosting Decision Tree optimization method based on a random greedy algorithm according to claim 1, wherein the optimal segmentation point algorithm in the Step 3:

determines a segmentation objective function, including an objective function,

5 information gain: information gain is the most commonly used index to measure a purity of a sample set; assuming that there are K types of samples in a node sample set D, in which a proportion of a k^{th} type of samples is p_k , an information entropy of D is defined as:

$$\text{Ent}(D) = - \sum_{k=1}^{|K|} p_k \log_2 p_k$$

10 assuming that the node is segmented into V possible values according to an attribute a, the information gain is defined as:

$$\text{Gain}(D, a) = \text{Ent}(D) - \sum_{v=1}^V \frac{|D^v|}{|D|} \text{Ent}(D^v);$$

information gain rate:

$$\text{Gain_ratio}(D, a) = \frac{\text{Gain}(D, a)}{\text{IV}(a)}$$

where

$$15 \quad \text{IV}(a) = - \sum_{v=1}^V \frac{|D^v|}{|D|} \log_2 \frac{|D^v|}{|D|}$$

a Gini coefficient:

$$\begin{aligned} \text{Gini}(D) &= \sum_{k=1}^{|K|} \sum_{k' \neq k} p_k p_{k'} \\ \text{Gini_index}(D, a) &= \sum_{v=1}^V \frac{|D^v|}{|D|} \text{Gini}(D^v) \end{aligned}$$

a structural coefficient:

$$20 \quad \text{Gain} = \frac{1}{2} \left(\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{G^2}{H + \lambda} \right) - \gamma$$

where G_L is a sum of first-order gradients of the data set divided into a left node according to the segmentation point, H_L is a sum of second-order gradients of the data set of the left node, and G_R and H_R are sums of the gradient information of a corresponding right node, γ is a tree model complexity penalty term and λ is a second-order regular term;

25 determines a candidate list of segmentation values: determining the list of segmentation values according to the data distribution of the current node, wherein the segmentation values

comprise segmentation features and segmentation feature values; the list of segmentation value is determined according to the following method:

all values of all features in the data set;

determining a discrete segmentation point according to a value range of each feature in the

5 data set;

wherein the selection of the segmentation points can be evenly distributed within the value range according to the distribution of data, and the amount of data uniformly reflected between the segmentation points is approximately equal or a sum of the second-order gradients is approximately equal;

10 traversing the candidate list of segmentation values to find the segmentation point that makes the objective function optimal.

3. The horizontal federated Gradient Boosting Decision Tree optimization method based on a random greedy algorithm according to claim 1, wherein the Epsilon greedy algorithm in
15 the Step 7 comprises:

for the node n, each participant sending the node segmentation point information to the coordinator, including a segmentation feature f_i , a segmentation value v_i , a number of node samples N_i and a local objective function gain g_i , where i represents respective participants;

20 according to the segmentation information of each participant and based on a maximum number principle, the coordinator determining an optimal segmentation feature $f_{\max}$, where X is a random number evenly distributed among $[0,1]$ and randomly sampling X to obtain x ; if $x \leq \epsilon$, randomly selecting one of the segmentation features of each participant as a global segmentation feature; otherwise, selecting $f_{\max}$ as the global segmentation feature;

25 each participant recalculating the segmentation information according to the global segmentation feature and sending the segmentation information to the coordinator;

the coordinator determining a global segmentation value according to the following formula: if the total number of participants is P ,

$$v = \sum_{i=1}^P \frac{N_i g_i}{\sum_{j=1}^P N_j g_j} v_i;$$

distributing the segmentation value to each participant to perform node segmentation.

30

4. The horizontal federated Gradient Boosting Decision Tree optimization method based on a random greedy algorithm according to claim 1, wherein the horizontal federated learning is a distributed structure of federated learning, in which each distributed node has the same data

feature and different sample spaces.

5 5. The horizontal federated Gradient Boosting Decision Tree optimization method based on a random greedy algorithm according to claim 1, wherein the Gradient Boosting Decision Tree algorithm is an integrated model based on gradient boosting and decision tree.

10 6. The horizontal federated Gradient Boosting Decision Tree optimization method based on a random greedy algorithm according to claim 1, wherein the decision tree is a basic model of a Gradient Boosting Decision Tree model, and a prediction direction of a sample is judged at the node by given features based on a tree structure.

15 7. The horizontal federated Gradient Boosting Decision Tree optimization method based on a random greedy algorithm according to claim 1, wherein the segmentation point is a segmentation position of non-leaf nodes in the decision tree for data segmentation.

20 8. The horizontal federated Gradient Boosting Decision Tree optimization method based on a random greedy algorithm according to claim 1, wherein the histogram is statistical information representing the first-order gradient and the second-order gradient in node data.

25 9. The horizontal federated Gradient Boosting Decision Tree optimization method based on a random greedy algorithm according to claim 1, wherein an input device can be one or more of data terminals such as computers and mobile phones or mobile terminals.

30 10. The horizontal federated Gradient Boosting Decision Tree optimization method based on a random greedy algorithm according to claim 1, wherein the input device comprises a processor, and when executed by the processor, the algorithm of any one of steps 1 to 12 is implemented.