SWARM INTELLIGENCE BASED FEDERATED LEARNING COLLABORATION METHOD AND SYSTEM FOR UNSTRUCTURED DATABASE

TECHNICAL FIELD

[0001] The present invention relates to a structured data technology, and specifically relates to a swarm intelligence based federated learning collaboration method and system for an unstructured database.

BACKGROUND

[0002] Unstructured databases, such as document databases and graph databases, have significant advantages in storage and management of complex data, and are widely used in various fields. With the continuous growth of data scale and the increasingly complex query demands, efficient query processing has become a key challenge. Traditional query optimization methods usually rely on predefined modes or indexes, making it difficult to adapt to the flexibility and diversity of unstructured data. Moreover, in a distributed environment, how to coordinate the resources of a plurality of database nodes to minimize query delay and maximize resource utilization is also a problem that is urgently needed to be solved.

[0003] Firstly, traditional query optimization methods struggle to effectively handle queries of the unstructured data. Due to the absence of a fixed mode for the unstructured data, traditional optimization strategies based on indexes or statistical information are difficult to apply, resulting in low query efficiency.

[0004] Secondly, existing distributed query processing methods usually lack a global resource coordination mechanism. Each database node independently executes query tasks, which can easily lead to resource competition and load imbalance, thus affecting overall query performance.

[0005] Finally, the traditional query optimization methods are often static and difficult to adapt to changes in query load. As data scale and query modes change, pre-established optimization strategies may no longer be effective, requiring dynamic adjustments to query execution plans to meet new demands.

SUMMARY OF THE INVENTION

[0006] The embodiments of the present invention provide a swarm intelligence based

federated learning collaboration method and system for an unstructured database, aiming to solve the problems in the related technology.

[0007] In a first aspect,

[0008] the swarm intelligence based federated learning collaboration method for the unstructured database provided by the embodiment of the present invention includes:

[0009] collecting query statement feature vectors, query resource consumption indexes, query response time indexes and query result accuracy indexes of a plurality of unstructured database nodes, and constructing the collected data into an initial training sample set; inputting the initial training sample set into a preset deep neural network, performing training through a federated learning framework to obtain a query mode identification sub-model, a path programming sub-model and a resource allocation sub-model through, and generating corresponding initial model parameters; establishing a swarm intelligence collaboration network based on the initial model parameters, and setting an initial path convergence time threshold and a resource utilization rate threshold as network assessment references;

[0010] transmitting a query request inputted by a user into the query mode identification sub-model for semantic parsing, extracting query semantic features, and generating a candidate query path by the path programming sub-model according to the initial path convergence time threshold; inputting the candidate query path into a hybrid swarm intelligence algorithm that fuses an ant colony path optimization mechanism and a particle swarm parameter optimization mechanism, and dynamically optimizing based on the resource utilization rate threshold to generate a query execution scheme; performing distributed resource scheduling by the resource allocation sub-model according to the query execution scheme to form a final query execution scheme; and executing the final query execution scheme, and recording actual path convergence time and resource occupation condition; and

[0011] performing differential analysis on the actual path convergence time and the initial path convergence time threshold, as well as the resource occupation condition and the resource utilization rate threshold to generate a performance optimization target; synchronously updating local model parameters of each node based on the performance optimization target, fusing the updated local model parameters through a security aggregation algorithm, and reconstructing to generate optimized global model parameters; and distributing the optimized global model parameters to the swarm intelligence collaboration network, and dynamically adjusting the initial path convergence time threshold and the resource utilization rate threshold to form a new assessment reference for

guiding the next round of query optimization.

[0012] The inputting the candidate query path into a hybrid swarm intelligence algorithm that fuses an ant colony path optimization mechanism and a particle swarm parameter optimization mechanism, and dynamically optimizing based on the resource utilization rate threshold to generate a query execution scheme includes:

[0013] constructing an initial candidate query path set, mapping an execution sequence of each path in the initial candidate query path set into a path code in an integer sequence form, mapping a resource allocation strategy for query operation into a parameter code in a real number vector form, and establishing a double-layer path representation model for path structure and resource allocation; computing a query cost ratio of each candidate path based on the double-layer path representation model, and generating an initial path assessment index;

[0014] treating the ratio of the initial path assessment index to a preset reference threshold as path heuristic information to compute a path transfer probability matrix in combination with initial pheromone concentration; inputting the path transfer probability matrix into a hybrid swarm intelligence optimization model, the hybrid swarm intelligence optimization model fusing the ant colony path optimization mechanism and the particle swarm parameter optimization mechanism; generating a path structure optimization sequence by an ant colony algorithm based on the path transfer probability matrix; and

[0015] treating an assessment score of the path structure optimization sequence as a pheromone increment, and transforming the pheromone increment into a fitness function of the particle swarm algorithm; guiding the particle swarm algorithm to perform iterative optimization on the resource allocation parameters based on the fitness function, and generating a parameter optimization vector; collecting system resource utilization rate data, and when the resource utilization rate data exceeds a preset reference threshold, adaptively adjusting a value space of the parameter optimization vector according to the exceeding degree; and combining the path structure optimization sequence with the parameter optimization vector to form a complete query execution scheme.

[0016] The treating the ratio of the initial path assessment index to a preset reference threshold as path heuristic information to compute a path transfer probability matrix in combination with initial pheromone concentration; inputting the path transfer probability matrix into a hybrid swarm intelligence optimization model, the hybrid swarm intelligence optimization model fusing the ant colony path optimization mechanism and the particle swarm parameter optimization mechanism; generating a path structure optimization sequence by an ant colony algorithm based on the path transfer probability matrix includes:

[0017] computing the ratio of the initial path assessment index to the preset reference threshold to obtain the path heuristic information, the initial path assessment index including a path length index, a path energy consumption index and a path time index, and the preset reference threshold including a reference path length threshold, a reference path energy consumption threshold and a reference path time threshold;

[0018] performing combination operation on the path heuristic information and the initial pheromone concentration to obtain the path transfer probability matrix, the initial pheromone concentration being a fixed initial value, and a computing result of the path transfer probability matrix including a pheromone concentration weight coefficient and a heuristic information weight coefficient;

[0019] inputting the path transfer probability matrix into the hybrid swarm intelligence optimization model, the hybrid swarm intelligence optimization model fusing the ant colony path optimization mechanism and the particle swarm parameter optimization mechanism, the ant colony path optimization mechanism being used for path structure search, and the particle swarm parameter optimization mechanism being used for optimizing the pheromone concentration weight coefficient and the heuristic information weight coefficient;

[0020] generating the candidate path by the ant colony algorithm based on the path transfer probability matrix, computing a path assessment score corresponding to the candidate path, determining a pheromone release amount according to the path assessment score, and updating the path transfer probability matrix based on the pheromone release amount and a preset pheromone volatilization coefficient;

[0021] treating the path assessment score as the fitness value of the particle swarm algorithm, updating an individual optimal solution and a global optimal solution of a particle swarm based on the fitness value, and computing a particle velocity update amount and position update amount according to the individual optimal solution and the global optimal solution; and

[0022] treating the updated particle position as a weight coefficient configuration, optimizing the path transfer probability matrix, and re-inputting the optimized path transfer probability matrix into the ant colony path optimization mechanism for iteration until the path structure optimization sequence is obtained by convergence.

[0023] The performing distributed resource scheduling by the resource allocation sub-model according to the query execution scheme to form a final query execution scheme; and executing the final query execution scheme, and recording actual path convergence time and resource occupation condition includes:

[0024] collecting processor performance indexes, internal memory capacity indexes, network bandwidth indexes and storage capacity indexes of each resource pool by the resource allocation sub-model to construct a resource feature matrix; computing resource pool scores according to the resource feature matrix, and performing initial resource allocation on the inputted query execution scheme based on the resource pool scores;

[0025] collecting real-time resource service data of each node by the resource allocation sub-model according to the initial resource allocation result, and computing node load states in combination with a resource type weight factor; when the node load state exceeds a preset load threshold, determining an optimal migration target node based on task migration cost and load balance income, and generating an optimized query execution scheme; and

[0026] performing distributed resource scheduling on the optimized query execution scheme by the resource allocation sub-model, and allocating execution resources based on the real-time computation capability, the storage capacity and the network state of each node, so as to form the final query execution scheme.

[0027] The performing differential analysis on the actual path convergence time and the initial path convergence time threshold, as well as the resource occupation condition and the resource utilization rate threshold to generate a performance optimization target; synchronously updating local model parameters of each node based on the performance optimization target, fusing the updated local model parameters through a security aggregation algorithm, and reconstructing to generate optimized global model parameters includes:

[0028] constructing a path convergence time monitoring module and a resource occupation monitoring module, the path convergence time monitoring module collecting the actual path convergence time and comparing the actual path convergence time with a preset initial path convergence time threshold to generate a time dimension difference sequence, and the resource occupation monitoring module collecting resource occupation data of each node and comparing the resource occupation data with a preset resource utilization rate threshold to generate a resource dimension difference sequence;

[0029] performing dynamic time warping on the time dimension difference sequence, constructing a time dimension cost matrix containing a deviation value of each time point, and solving an optimal matching path through a dynamic programming algorithm to obtain a time difference score; computing weighted Euclidean distances of a processor occupation rate, an internal memory occupation rate and a network bandwidth occupation rate for the resource dimension difference sequence, and obtaining a resource difference score in

combination with a resource importance weight coefficient;

[0030] performing normalization processing on the time difference score and the resource difference score, and fusing based on a harmonic average method to generate a performance difference index; constructing a performance optimization target function containing stability constraints according to the performance difference index, and adaptively adjusting a weight coefficient of the performance optimization target function according to fluctuation degree of the time difference score and the resource difference score;

[0031] training local models in parallel at each node through the federated learning framework based on the performance optimization target function, updating the local model parameters by a stochastic gradient descent method with a momentum item, and dynamically adjusting the learning rate according to time difference scores in the training process; performing fragmentation processing on the updated local model parameters, and processing parameter fragments by a homomorphic encryption algorithm to generate encryption parameter blocks; and

[0032] allocating key fragments related to the resource difference score for each node, and combining the key fragments with stochastic mask information to generate a security aggregation key; performing distributed aggregation operation on the encryption parameter blocks through the security aggregation key to generate a global parameter aggregation matrix; computing node weight coefficients according to the time difference score and the resource difference score of each node, performing weighted averaging on the global parameter aggregation matrix based on the node weight coefficients, performing sparsification processing on the parameters subjected to weighted averaging, and reconstructing to generate the optimized global model parameters.

[0033] The performing dynamic time warping on the time dimension difference sequence, constructing a time dimension cost matrix containing a deviation value of each time point, and solving an optimal matching path through a dynamic programming algorithm to obtain a time difference score; computing weighted Euclidean distances of a processor occupation rate, an internal memory occupation rate and a network bandwidth occupation rate for the resource dimension difference sequence, and obtaining a resource difference score in combination with a resource importance weight coefficient includes:

[0034] constructing the time dimension cost matrix according to the time dimension difference sequence, each element of the time dimension cost matrix containing a time point difference distance item and a time jump penalty item, the time point difference distance item representing a deviation degree of a point corresponding to the time sequence,

and the time jump penalty item restraining nonlinear distortion of the time sequence through a preset penalty coefficient;

[0035] performing cumulative cost computation on the time dimension cost matrix to obtain a cumulative cost matrix, and searching a minimum cumulative cost path by the cumulative cost matrix through the dynamic programming algorithm; obtaining the optimal matching path by a backtracking mode based on the minimum cumulative cost path, and performing normalization processing on the cumulative cost at the optimal matching path to obtain a time dimension scoring sequence;

[0036] collecting the real-time resource occupation data of each node in the system, including the processor occupation rate, the internal memory occupation rate and the network bandwidth occupation rate, and computing a difference value between the real-time resource occupation data and the preset resource utilization rate threshold to obtain the resource dimension difference sequence; constructing a resource index judgment matrix by an analytic hierarchy process, and computing the feature vectors of the resource index judgment matrix to obtain the importance weight coefficient corresponding to each resource index; and

[0037] sampling the resource dimension difference sequence based on a sliding observation window with a preset size, and extracting the statistical features of the resource dimension difference sequence in the sliding observation window; and performing weighted computation on the resource dimension difference sequence and the importance weight coefficient, and measuring by the weighted Euclidean distance to obtain the resource difference score.

[0038] The distributing the optimized global model parameters to the swarm intelligence collaboration network, and dynamically adjusting the initial path convergence time threshold and the resource utilization rate threshold to form a new assessment reference for guiding the next round of query optimization includes:

[0039] constructing a topological structure of the swarm intelligence collaboration network, and computing an adjacency matrix based on a physical distance among the nodes and network link quality; dividing the adjacency matrix into a plurality of sub-network groups by a spectral clustering algorithm, and selecting a regional coordinator in each sub-network group based on the node computation capability and network connection stability;

[0040] collecting hardware configuration parameters and task load information of each node, and performing differential processing on the optimized global model parameters based on the hardware configuration parameters and the task load information

to generate a parameter deployment scheme adapted to different nodes; constructing a parameter integrity check matrix by a bloom filter, and distributing the parameter deployment scheme to each node in the swarm intelligence collaboration network by the regional coordinator;

[0041] deploying a performance collection agent in the swarm intelligence collaboration network, and collecting the path convergence time and the resource utilization rate after the nodes execute the parameter deployment scheme; computing a deviation value between the path convergence time and the initial path convergence time threshold and a difference value between the resource utilization rate and the resource utilization rate threshold based on the sliding time window;

[0042] inputting the deviation value and the difference value into an orthogonal test model, and analyzing the influence degree of the initial path convergence time threshold and the resource utilization rate threshold on the system performance; extracting a threshold combination by a principal component analysis method, and establishing a relationship between a mapping performance difference of a support vector regression model and a threshold adjustment amount;

[0043] constructing a dual Q learning network, treating the performance difference as a state input, and treating a threshold adjustment direction and a step length as action output; designing a reward function based on the performance improvement measurement, training a threshold dynamic adjustment strategy, and adaptively adjusting an exploration probability of the dual Q learning network according to the performance fluctuation degree;

[0044] performing exponential moving average processing on the performance data after the threshold dynamic adjustment strategy is executed, and generating a performance trend prediction value in combination with a time sequence prediction model; dynamically adjusting the initial path convergence time threshold and the resource utilization rate threshold based on the performance trend prediction value to form a new assessment reference; and

[0045] storing the performance data and the threshold dynamic adjustment strategy into an optimization knowledge base, and extracting an optimization rule to generate a scenario-based template; and taking the scenario-based template and the new assessment reference as a guidance basis for the next round of query optimization.

[0046] In a second aspect,

[0047] the swarm intelligence based federated learning collaboration system for the unstructured database provided by the embodiment of the present invention includes:

[0048] a first unit, configured to collect query statement feature vectors, query

resource consumption indexes, query response time indexes and query result accuracy indexes of a plurality of unstructured database nodes, and construct the collected data into an initial training sample set; input the initial training sample set into a preset deep neural network, perform training through a federated learning framework to obtain a query mode identification sub-model, a path programming sub-model and a resource allocation sub-model through, and generate corresponding initial model parameters; and establish a swarm intelligence collaboration network based on the initial model parameters, and set an initial path convergence time threshold and a resource utilization rate threshold as network assessment references;

[0049] a second unit, configured to transmit a query request inputted by a user into the query mode identification sub-model for semantic parsing, extract query semantic features, and generate a candidate query path by the path programming sub-model according to the initial path convergence time threshold; input the candidate query path into a hybrid swarm intelligence algorithm that fuses an ant colony path optimization mechanism and a particle swarm parameter optimization mechanism, and dynamically optimize to generate a query execution scheme based on the resource utilization rate threshold; perform distributed resource scheduling by the resource allocation sub-model according to the query execution scheme to form a final query execution scheme; and execute the final query execution scheme, and record actual path convergence time and resource occupation condition; and

[0050] a third unit, configured to perform differential analysis on the actual path convergence time and the initial path convergence time threshold, as well as the resource occupation condition and the resource utilization rate threshold to generate a performance optimization target; synchronously update local model parameters of each node based on the performance optimization target, fuse the updated local model parameters through a security aggregation algorithm, and reconstruct to generate optimized global model parameters; and distribute the optimized global model parameters to the swarm intelligence collaboration network, and dynamically adjust the initial path convergence time threshold and the resource utilization rate threshold to form a new assessment reference for guiding the next round of query optimization.

[0051] In a third aspect,

[0052] an electronic device provided by the embodiment of the present invention includes:

[0053] a processor; and

[0054] a memory configured to store processor executable instructions;

[0055] the processor is configured to call instructions stored in the memory to execute

the abovementioned method.

[0056] In a fourth aspect,

[0057] the embodiment of the present invention provides a computer-readable storage medium having computer program instructions stored therein, when executed by a processor, the computer program instructions execute the abovementioned method.

[0058] The present invention has the following beneficial effects:

[0059] 1. the query efficiency is improved: by the coordinative work of the query mode identification, path programming and resource allocation sub-models and the optimization of the hybrid swarm intelligence algorithm, the optimal query execution scheme can be dynamically generated, which can reduce the query response time, and improve the query efficiency.

[0060] 2. The resource utilization is optimized: dynamic resource scheduling is performed based on the resource utilization rate threshold, the differential analysis and model parameter updating are performed according to the actual resource occupation condition, therefore, resource waste can be effectively avoided, and the resource utilization rate is increased.

[0061] 3. Adaptive learning and optimization are performed: through the combination of the federated learning framework and the swarm intelligence collaboration network, the system can continuously learn and optimize the model parameters, the path convergence time threshold and the resource utilization rate threshold according to the query request of the user and the system running state, thus the adaptive query optimization is achieved, and the overall performance of the system is improved. The resource utilization rate threshold is adopted, so that adaptive query optimization is achieved, and the overall performance of the system is improved.

BRIEF DESCRIPTION OF THE DRAWINGS

[0062] FIG. 1 is a flowchart of a swarm intelligence based federated learning collaboration method for an unstructured database according to an embodiment of the present invention; and

[0063] FIG. 2 is a schematic structural diagram of a swarm intelligence based federated learning collaboration system for an unstructured database according to an embodiment of the present invention.

DETAILED DESCRIPTION

[0064] In order to make the purpose, technical solution and advantages of the

embodiment of the present invention clearer, the technical solutions in the embodiments of the present invention will be clearly and completely described in combination with the accompanying drawings in the embodiments of the present invention. Definitely, the described embodiments are only a part of the embodiments of the present invention, not all of them. Based on the embodiments in the present invention, all other embodiments obtained by those of ordinary skill in the art without carrying out creative labor fall within the scope of protection of the present invention.

[0065] The technical solutions of the present invention are described in detail by specific embodiments below. The following specific embodiments can be combined with each other, and the same or similar concepts or processes may not be repeated in some embodiments.

[0066] FIG. 1 is a flowchart of a swarm intelligence based federated learning collaboration method for an unstructured database according to an embodiment of the present invention, as shown in FIG. 1, the method includes:

[0067] S11: collecting query statement feature vectors, query resource consumption indexes, query response time indexes and query result accuracy indexes of a plurality of unstructured database nodes, and constructing the collected data into an initial training sample set; inputting the initial training sample set into a preset deep neural network, performing training through a federated learning framework to obtain a query mode identification sub-model, a path programming sub-model and a resource allocation sub-model through, and generating corresponding initial model parameters; establishing a swarm intelligence collaboration network based on the initial model parameters, and setting an initial path convergence time threshold and a resource utilization rate threshold as network assessment references;

[0068] S12: transmitting a query request inputted by a user into the query mode identification sub-model for semantic parsing, extracting query semantic features, and generating a candidate query path by the path programming sub-model according to the initial path convergence time threshold; inputting the candidate query path into a hybrid swarm intelligence algorithm that fuses an ant colony path optimization mechanism and a particle swarm parameter optimization mechanism, and dynamically optimizing based on the resource utilization rate threshold to generate a query execution scheme; performing distributed resource scheduling by the resource allocation sub-model according to the query execution scheme to form a final query execution scheme; and executing the final query execution scheme, and recording actual path convergence time and resource occupation condition; and

[0069] S13: performing differential analysis on the actual path convergence time and the initial path convergence time threshold, as well as the resource occupation condition and the resource utilization rate threshold to generate a performance optimization target; synchronously updating local model parameters of each node based on the performance optimization target, fusing the updated local model parameters through a security aggregation algorithm, and reconstructing to generate optimized global model parameters; and distributing the optimized global model parameters to the swarm intelligence collaboration network, and dynamically adjusting the initial path convergence time threshold and the resource utilization rate threshold to form a new assessment reference for guiding the next round of query optimization.

[0070] In an optional embodiment, the inputting the candidate query path into a hybrid swarm intelligence algorithm that fuses an ant colony path optimization mechanism and a particle swarm parameter optimization mechanism, and dynamically optimizing based on the resource utilization rate threshold to generate a query execution scheme includes:

[0071] constructing an initial candidate query path set, mapping an execution sequence of each path in the initial candidate query path set into a path code in an integer sequence form, mapping a resource allocation strategy for query operation into a parameter code in a real number vector form, and establishing a double-layer path representation model for path structure and resource allocation; computing a query cost ratio of each candidate path based on the double-layer path representation model, and generating an initial path assessment index;

[0072] treating the ratio of the initial path assessment index to a preset reference threshold as path heuristic information to compute a path transfer probability matrix in combination with initial pheromone concentration; inputting the path transfer probability matrix into a hybrid swarm intelligence optimization model, the hybrid swarm intelligence optimization model fusing the ant colony path optimization mechanism and the particle swarm parameter optimization mechanism; generating a path structure optimization sequence by an ant colony algorithm based on the path transfer probability matrix; and

[0073] treating an assessment score of the path structure optimization sequence as a pheromone increment, and transforming the pheromone increment into a fitness function of the particle swarm algorithm; guiding the particle swarm algorithm to perform iterative optimization on the resource allocation parameters based on the fitness function, and generating a parameter optimization vector; collecting system resource utilization rate data, and when the resource utilization rate data exceeds a preset reference threshold, adaptively adjusting a value space of the parameter optimization vector according to the exceeding

degree; and combining the path structure optimization sequence with the parameter optimization vector to form a complete query execution scheme.

[0074] A query execution scheme generation method based on the hybrid swarm intelligence algorithm that fuses the ant colony path optimization mechanism and the particle swarm parameter optimization mechanism can dynamically optimize the query path and the resource allocation strategy for the database query operations on the premise of meeting resource utilization rate constraints, so as to minimize the query cost.

[0075] Firstly, the initial candidate query path set is constructed. For example, for a query task containing three query operations (A, B, C), possible execution sequences include ABC, ACB, BAC, BCA, CAB, and CBA, which form the initial candidate query path set. The execution sequence of each path is mapped into the path code in the integer sequence form, for example, the ABC code is [1, 2, 3], the ACB code is [1, 3, 2], and so on. Moreover, the resource allocation strategy (for example, allocation proportion of resources such as a CPU, an internal memory, and IO) for the query operation are mapped into the parameter codes in the real number vector form, for example, [0.5, 0.3, 0.2] represents that 50% of CPU resources, 30% of internal memory resources and 20% of IO resources are allocated to the current operation. Therefore, the double-layer path representation model for the path structure and resource allocation is established.

[0076] Secondly, the query cost ratio of each candidate path is computed based on the double-layer path representation model, and thus the initial path assessment index is generated. The query cost ratio can be defined as a weighted averaging value of query execution time and resource consumption. For example, if the execution time of the path ABC is 10 sec, and the resource consumption is 5 units, the cost ratio after weighted averaging will be 0.8; and if the execution time of the path ACB is 8 sec, and the resource consumption is 6 units, the cost ratio after weighted averaging will be 0.7, and so on.

[0077] Thirdly, the ratio of the initial path assessment index to the preset reference threshold is treated as the path heuristic information. If the preset reference threshold is 0.6, the heuristic information of the path ABC will be 0.8/0.6 = 1.33, the heuristic information of the path ACB will be 0.7/0.6 = 1.17, and so on. The path transfer probability matrix is computed in combination with the initial pheromone concentration (for example, the initial pheromone concentration of all paths is 1). The path transfer probability matrix represents the probability of an ant selecting the next node at the current node, and the higher the probability value is, the higher the possibility that this path is selected is.

[0078] The path transfer probability matrix is inputted into the hybrid swarm intelligence optimization model. The model fuses the ant colony path optimization

mechanism and the particle swarm parameter optimization mechanism. Based on the path transfer probability matrix, the path structure optimization sequence is generated through the ant colony algorithm. For example, after several iterations, the ant colony algorithm may generate a path optimization sequence ACB-BAC-ABC, which indicates that the query efficiency of the three paths is high.

[0079] The assessment score of the path structure optimization sequence is treated as the pheromone increment. For example, if the assessment score of ACB is the highest, its pheromone increment will be the highest. The pheromone increment is transformed into the fitness function of the particle swarm algorithm. The particle swarm algorithm is guided to iteratively optimize the resource allocation parameters based on the fitness function, so as to generate the parameter optimization vector. For example, after optimization, the parameter optimization vector may be [0.6, 0.2, 0.2], which indicates that 60% of CPU resources, 20% of internal memory resources and 20% of IO resources are allocated to the current operation.

[0080] The system resource utilization rate data is collected. When the resource utilization rate data exceeds the preset reference threshold, the value space of the parameter optimization vector will be adaptively adjusted according to the exceeding degree. For example, when the CPU utilization rate exceeds 80%, the CPU resource allocation proportion will be reduced, and the allocation proportions of other CPU resources will be correspondingly adjusted.

[0081] Finally, the path structure optimization sequence and the parameter optimization vector are combined to form the complete query execution scheme. For example, the final query execution scheme may be that: the query is performed according to a path sequence of ACB, and the resource allocation strategy of [0.6, 0.2, 0.2] is adopted; and if the resource utilization rate exceeds the threshold, the parameter optimization vector will be dynamically adjusted.

[0082] Through the solution of the present invention:

[0083] the query efficiency can be improved: by optimizing the query path and the resource allocation strategy, the query execution time can be effectively reduced, and the query efficiency is improved. The resource consumption can be reduced: by dynamically adjusting the resource allocation strategy, excessive resource use can be avoided, and the resource consumption is reduced. The system stability can be enhanced: by monitoring the resource utilization rate and dynamically adjusting the parameters, system overload can be avoided, and the system stability is enhanced.

[0084] In an optional embodiment, the treating the ratio of the initial path assessment

index to a preset reference threshold as path heuristic information to compute a path transfer probability matrix in combination with initial pheromone concentration; inputting the path transfer probability matrix into a hybrid swarm intelligence optimization model, the hybrid swarm intelligence optimization model fusing the ant colony path optimization mechanism and the particle swarm parameter optimization mechanism; generating a path structure optimization sequence by an ant colony algorithm based on the path transfer probability matrix includes:

[0085] computing the ratio of the initial path assessment index to the preset reference threshold to obtain the path heuristic information, the initial path assessment index including a path length index, a path energy consumption index and a path time index, and the preset reference threshold including a reference path length threshold, a reference path energy consumption threshold and a reference path time threshold;

[0086] performing combination operation on the path heuristic information and the initial pheromone concentration to obtain the path transfer probability matrix, the initial pheromone concentration being a fixed initial value, and a computing result of the path transfer probability matrix including a pheromone concentration weight coefficient and a heuristic information weight coefficient;

[0087] inputting the path transfer probability matrix into the hybrid swarm intelligence optimization model, the hybrid swarm intelligence optimization model fusing the ant colony path optimization mechanism and the particle swarm parameter optimization mechanism, the ant colony path optimization mechanism being used for path structure search, and the particle swarm parameter optimization mechanism being used for optimizing the pheromone concentration weight coefficient and the heuristic information weight coefficient;

[0088] generating the candidate path by the ant colony algorithm based on the path transfer probability matrix, computing a path assessment score corresponding to the candidate path, determining a pheromone release amount according to the path assessment score, and updating the path transfer probability matrix based on the pheromone release amount and a preset pheromone volatilization coefficient;

[0089] treating the path assessment score as the fitness value of the particle swarm algorithm, updating an individual optimal solution and a global optimal solution of a particle swarm based on the fitness value, and computing a particle velocity update amount and position update amount according to the individual optimal solution and the global optimal solution; and

[0090] treating the updated particle position as a weight coefficient configuration,

optimizing the path transfer probability matrix, and re-inputting the optimized path transfer probability matrix into the ant colony path optimization mechanism for iteration until the path structure optimization sequence is obtained by convergence.

[0091] Firstly, the path information is initialized. A network structure for to-be-optimized paths is constructed, and the network structure contains a plurality of nodes and paths that connect the nodes. The initial path assessment indexes are set for each path, including the path length, path energy consumption and path time. The preset reference thresholds are set, including the reference path length threshold, the reference path energy consumption threshold and the reference path time threshold, for example, they are set to be 10 km, 20 kwh and 30 min respectively.

[0092] Secondly, the path heuristic information is computed. The length, energy consumption and time indexes of each path are compared with the corresponding reference thresholds respectively, and accordingly the ratio is computed. For example, if the length of one path is 12 km, the heuristic information of the length index will be 12/10 = 1.2. Similarly, the heuristic information of the energy consumption index and the time index is computed.

[0093] Thirdly, the initial path transfer probability matrix is constructed. The initial pheromone concentration is set, for example, it is set to be 1. The combination operation is performed on the heuristic information of each path and the initial pheromone concentration to obtain the path transfer probability matrix. The combination operation can be performed by multiplying the pheromone concentration and the heuristic information by corresponding weight coefficients respectively and then adding. The initial weight coefficients can be set to be equal values, such as 0.5.

[0094] Fourthly, the path transfer probability matrix is inputted into the hybrid swarm intelligence optimization model. This model fuses the ant colony algorithm and the particle swarm algorithm. The ant colony algorithm is used for path structure search, and it simulates the movement of ants in the path network according to the path transfer probability matrix and gradually constructs the candidate paths. The particle swarm algorithm is used for optimizing the pheromone concentration weight coefficient and the heuristic information weight coefficient, so as to improve the path search efficiency.

[0095] The candidate paths are generated by the ant colony algorithm based on the path transfer probability matrix. The path assessment score of each candidate path is computed, and the score can be a weighted sum of the path length, the energy consumption and the time. For example, if the length of one candidate path is 11 km, the energy consumption is 18 kwh, and the time is 28 min, its path assessment score can be computed

as 0.5 * 11/10 + 0.25 * 18/20 + 0.25 * 28/30 = 1.0083.

[0096] The pheromone release amount is determined according to the assessment score of the candidate paths. The higher the score is, the more the released pheromone is. For example, the score can be directly treated as the pheromone release amount. Then, the path transfer probability matrix is updated according to the pheromone release amount and the preset pheromone volatilization coefficient. The pheromone volatilization coefficient can be set to be 0.1, which indicates that the pheromone concentration is reduced by 10% after each iteration.

[0097] The path assessment score is treated as the fitness value of the particle swarm algorithm. The individual optimal solution and the global optimal solution of the particle swarm are updated according to the fitness value. Then, the particle velocity update amount and position updating amount are computed according to the individual optimal solution and the global optimal solution, and the particle positions, namely the pheromone concentration weight coefficient and the heuristic information weight coefficient, are updated.

[0098] The path transfer probability matrix is optimized by taking the updated particle position as the weight coefficient. The optimized path transfer probability matrix is re-inputted into the ant colony algorithm for iteration, and the above steps are repeated until the algorithm converges, so as to obtain the final path structure optimization sequence.

[0099] Through the solution of the present invention:

[0100] the path search efficiency can be improved. By fusing the ant colony algorithm and the particle swarm algorithm and in combination with the path heuristic information and pheromone concentration, a high-quality path scheme can be found more quickly. The comprehensive optimization of path assessment indexes can be realized. By comprehensively considering a plurality of indexes such as the path length, energy consumption and time, a path scheme more meeting actual requirements can be obtained. The adaptability can be improved. According to the method, the parameters such as the preset reference threshold, the weight coefficient and the pheromone volatilization coefficient can be adjusted according to different application scenarios and requirements, so that the optimal optimization effect is obtained.

[0101] In an optional embodiment, the performing distributed resource scheduling by the resource allocation sub-model according to the query execution scheme to form a final query execution scheme; and executing the final query execution scheme, and recording actual path convergence time and resource occupation condition includes:

[0102] collecting processor performance indexes, internal memory capacity indexes,

network bandwidth indexes and storage capacity indexes of each resource pool by the resource allocation sub-model to construct a resource feature matrix; computing resource pool scores according to the resource feature matrix, and performing initial resource allocation on the inputted query execution scheme based on the resource pool scores;

[0103] collecting real-time resource service data of each node by the resource allocation sub-model according to the initial resource allocation result, and computing node load states in combination with a resource type weight factor; when the node load state exceeds a preset load threshold, determining an optimal migration target node based on task migration cost and load balance income, and generating an optimized query execution scheme; and

[0104] performing distributed resource scheduling on the optimized query execution scheme by the resource allocation sub-model, and allocating execution resources based on the real-time computation capability, the storage capacity and the network state of each node, so as to form the final query execution scheme.

[0105] Firstly, the resource feature matrix is constructed. The processor performance indexes of each resource pool are collected, such as CPU dominant frequency, and core number; the internal memory capacity indexes are collected, such as total internal memory size, and available internal memory size; the network bandwidth indexes are collected, such as upload bandwidth, and download bandwidth; and the storage capacity indexes are collected, such as total disk capacity, and available disk capacity. These collected indexes form a resource feature matrix, each row in the resource feature matrix represents one resource pool, and each column represents one resource index. For example, if the CPU dominant frequency of a resource pool 1 is 3 GHz, the core number is 8, the internal memory capacity is 32 GB, the network bandwidth is 10 Gbps, and the storage capacity is 1 TB, the feature vector corresponding to the resource pool 1 will be [3, 8, 32, 10, 1024].

[0106] Secondly, the resource pool scores are computed. The comprehensive score of each resource pool is computed according to the resource feature matrix. For example, the score can be computed in a weighted summation mode, and each resource index is endowed with different weights, for example, the CPU weight is 0.4, the internal memory weight is 0.3, the network bandwidth weight is 0.2, and the storage capacity weight is 0.1. Then, each index of each resource pool is multiplied by the corresponding weight and summated to obtain the score of this resource pool. If the score of each index of the resource pool 1 is: 0.8 points of CPU, 0.9 points of internal memory, 0.7 points of network bandwidth, and 0.6 points of storage capacity, the comprehensive score of the resource pool 1 will be $0.4 * 0.8 + 0.3 * 0.9 + 0.2 * 0.7 + 0.1 * 0.6 = 0.77$.

[0107] Thirdly, the initial resource allocation is performed. According to the computed resource pool scores, the initial resource allocation is performed on the inputted query execution scheme. For example, a query scheme contains three sub-tasks, which need different computing resources and storage resources respectively. According to the scores of the resource pools, the sub-tasks are allocated to different resource pools for execution, and the resource pool with higher score is preferentially selected. If the score of the resource pool 1 is the highest, the compute-intensive sub-task 1 will be allocated to the resource pool 1 for execution.

[0108] Fourthly, the real-time resource service data of the nodes is collected. After initial resource allocation, the real-time resource service data of each node is collected, such as CPU utilization rate, internal memory utilization rate, network bandwidth utilization rate and disk IO rate. For example, the CPU utilization rate of the resource pool 1 is 80%, the internal memory usage rate is 70%, the network bandwidth utilization rate is 60%, and the disk IO rate is 50%.

[0109] Fifthly, the node load state is computed. In combination with the resource type weight factor, such as 0.5 of weight of CPU, 0.3 of weight of internal memory, 0.1 of weight of network bandwidth, 0.1 of weight of disk IO, the node load state is computed. For example, the load state of resource pool 1 is 0.5 * 80% + 0.3 * 70% + 0.1 * 60% + 0.1 * 50% = 62%.

[0110] Sixthly, the optimal migration target node is determined. When the load state of the node exceeds the preset load threshold, such as 70%, it is needed to perform task migration. Based on the task migration cost such as the migration data size and the network transmission speed, and the load balance income such as the reduction degree of the load state of the node after migration, the optimal migration target node can be determined. For example, part of tasks on the resource pool 1 are migrated to a resource pool 2 with a low load, thus the load state of the resource pool 1 can be reduced, and the overall query execution efficiency is improved.

[0111] Seventhly, the distributed resource scheduling is performed. The distributed resource scheduling is performed on the optimized query execution scheme, and execution resources are allocated based on the real-time computation capability, the storage capacity and the network state of each node, so as to form the final query execution scheme.

[0112] Eighthly, the final query execution scheme is performed and related information is recorded. The finally determined query execution scheme is performed, and the actual path convergence time and the resource occupation condition are recorded, including the CPU utilization rate, the internal memory utilization rate, the network

bandwidth utilization rate and the like. Such information can be used in subsequent resource scheduling optimization.

[0113] Through the solution of the present invention:

[0114] the resource utilization rate can be increased: by dynamically adjusting the resource allocation scheme, the computation resources and storage resources of each resource pool can be fully utilized, thus resource waste is avoided, and the overall resource utilization rate is increased. The query delay can be reduced: the tasks are allocated to nodes with low loads to be executed, thus the task queuing waiting time can be reduced, the query delay is reduced, and query efficiency is improved. The system stability can be enhanced: by monitoring the load state of the nodes in real time and dynamically adjusting, system crash caused by excessive load of a single node can be avoided, and the system stability is enhanced.

[0115] In an optional embodiment, the performing differential analysis on the actual path convergence time and the initial path convergence time threshold, as well as the resource occupation condition and the resource utilization rate threshold to generate a performance optimization target; synchronously updating local model parameters of each node based on the performance optimization target, fusing the updated local model parameters through a security aggregation algorithm, and reconstructing to generate optimized global model parameters includes:

[0116] constructing a path convergence time monitoring module and a resource occupation monitoring module, the path convergence time monitoring module collecting the actual path convergence time and comparing the actual path convergence time with a preset initial path convergence time threshold to generate a time dimension difference sequence, and the resource occupation monitoring module collecting resource occupation data of each node and comparing the resource occupation data with a preset resource utilization rate threshold to generate a resource dimension difference sequence;

[0117] performing dynamic time warping on the time dimension difference sequence, constructing a time dimension cost matrix containing a deviation value of each time point, and solving an optimal matching path through a dynamic programming algorithm to obtain a time difference score; computing weighted Euclidean distances of a processor occupation rate, an internal memory occupation rate and a network bandwidth occupation rate for the resource dimension difference sequence, and obtaining a resource difference score in combination with a resource importance weight coefficient;

[0118] performing normalization processing on the time difference score and the resource difference score, and fusing based on a harmonic average method to generate a

performance difference index; constructing a performance optimization target function containing stability constraints according to the performance difference index, and adaptively adjusting a weight coefficient of the performance optimization target function according to fluctuation degree of the time difference score and the resource difference score;

[0119] training local models in parallel at each node through the federated learning framework based on the performance optimization target function, updating the local model parameters by a stochastic gradient descent method with a momentum item, and dynamically adjusting the learning rate according to time difference scores in the training process; performing fragmentation processing on the updated local model parameters, and processing parameter fragments by a homomorphic encryption algorithm to generate encryption parameter blocks; and

[0120] allocating key fragments related to the resource difference score for each node, and combining the key fragments with stochastic mask information to generate a security aggregation key; performing distributed aggregation operation on the encryption parameter blocks through the security aggregation key to generate a global parameter aggregation matrix; computing node weight coefficients according to the time difference score and the resource difference score of each node, performing weighted averaging on the global parameter aggregation matrix based on the node weight coefficients, performing sparsification processing on the parameters subjected to weighted averaging, and reconstructing to generate the optimized global model parameters.

[0121] Firstly, the path convergence time monitoring module and the resource occupation monitoring module are established. The path convergence time monitoring module is configured to collect the time spent by actual path convergence in the model training process, compare the time with the preset initial path convergence time threshold, and generate a sequence which reflects the time dimension difference, for example, if the preset threshold is 10 min, and the actual time is 9 min, 11 min, and 8 min respectively, the generated time dimension difference sequences will be -1, 1, and -2. The resource occupation monitoring module is configured to collect the resource occupation data of each node, including the processor occupation rate, the internal memory occupation rate, the network bandwidth occupation rate and the like, compare the collected data with the preset resource utilization rate threshold, and generate the sequence reflecting the resource dimension difference. For example, if the processor occupation rate threshold is 70%, and the actual occupation rate is 65%, 75%, and 60% respectively, the generated resource dimension difference sequences will be -5, 5, and -10.

[0122] Secondly, analysis processing is performed on the generated time dimension difference sequence and the resource dimension difference sequence. The dynamic time warping is performed on the time dimension difference sequence to construct the time dimension cost matrix containing the deviation value of each point, and then the optimal matching path is solved through the dynamic programming algorithm, thus obtaining the time difference score. For example, the time difference score computed by dynamic time warping and the dynamic programming algorithm is 0.8. For the resource dimension difference sequence, the weighted Euclidean distances of the processor occupation rate, the memory occupation rate and the network bandwidth occupation rate are computed respectively, and in combination with the resource importance weight coefficients (for example, the weight coefficients of the processor, the internal memory and the network bandwidth are 0.5, 0.3, and 0.2 respectively), the resource difference score can be obtained. For example, the computed resource difference score is 1.2.

[0123] Thirdly, the normalization processing is performed on the time difference score and the resource difference score to ensure that their values are within the same numerical range, such as 0 to 1. The two scores are fused based on a harmonic average method to generate the performance difference index. For example, if the normalized time difference score is 0.7 and the normalized resource difference score is 0.8, the performance difference index computed according to the harmonic average method will be 0.75. The performance optimization target function containing the stability constraints is established according to the performance difference index. The weight coefficient of the performance optimization target function is adaptively adjusted according to the fluctuation degree of the time difference score and the resource difference score, and the larger the fluctuation is, the smaller the weight coefficient is.

[0124] The local model is trained in parallel at each node by the federated learning framework according to the performance optimization target function. The local model parameters are updated by the random gradient descent method with the momentum item, and the learning rate is dynamically adjusted according to the time difference score in the training process. The larger the time difference score is, the smaller the learning rate is. The fragmentation processing is performed on the updated local model parameters, and the homomorphic encryption algorithm is performed on parameter fragments to generate the encryption parameter blocks. For example, the model parameters are divided into 10 fragments and encrypted respectively.

[0125] The key fragments related to the resource difference score are allocated for each node. The higher the resource difference score is, the more important the distributed

key fragments are. The key fragments and the stochastic mask information are combined to generate the security aggregation key. The security aggregation key is utilized to perform distributed aggregation operation on the encryption parameter blocks, so as to generate the global parameter aggregation matrix. The node weight coefficient is computed according to the time difference score and the resource difference score of each node. The lower the time difference score and the resource difference score are, the higher the node weight coefficient is. The weighted averaging is performed on the global parameter aggregation matrix based on the node weight coefficient, the sparsification processing is performed on parameters subjected to weighted averaging, the parameters with small weights are removed, and thus the optimized global model parameters are generated by reconstruction.

[0126] Through the solution of the present invention:

[0127] the model training efficiency can be improved: by dynamically adjusting the learning rate and optimizing the model parameters, the model convergence speed is increased, and the training time is reduced. The resource utilization can be optimized: by monitoring the resource occupation condition and adjusting the resource allocation strategy, the resource utilization rate is increased, and the training cost is reduced. The model robustness can be enhanced: through the federated learning framework and the security aggregation algorithm, the generalization ability and robustness of the model can be improved, and the practical value of the model is enhanced.

[0128] In an optional embodiment, the performing dynamic time warping on the time dimension difference sequence, constructing a time dimension cost matrix containing a deviation value of each time point, and solving an optimal matching path through a dynamic programming algorithm to obtain a time difference score; computing weighted Euclidean distances of a processor occupation rate, an internal memory occupation rate and a network bandwidth occupation rate for the resource dimension difference sequence, and obtaining a resource difference score in combination with a resource importance weight coefficient includes:

[0129] constructing the time dimension cost matrix according to the time dimension difference sequence, each element of the time dimension cost matrix containing a time point difference distance item and a time jump penalty item, the time point difference distance item representing a deviation degree of a point corresponding to the time sequence, and the time jump penalty item restraining nonlinear distortion of the time sequence through a preset penalty coefficient;

[0130] performing cumulative cost computation on the time dimension cost matrix to obtain a cumulative cost matrix, and searching a minimum cumulative cost path by the

cumulative cost matrix through the dynamic programming algorithm; obtaining the optimal matching path by a backtracking mode based on the minimum cumulative cost path, and performing normalization processing on the cumulative cost at the optimal matching path to obtain a time dimension scoring sequence;

[0131] collecting the real-time resource occupation data of each node in the system, including the processor occupation rate, the internal memory occupation rate and the network bandwidth occupation rate, and computing a difference value between the real-time resource occupation data and the preset resource utilization rate threshold to obtain the resource dimension difference sequence; constructing a resource index judgment matrix by an analytic hierarchy process, and computing the feature vectors of the resource index judgment matrix to obtain the importance weight coefficient corresponding to each resource index; and

[0132] sampling the resource dimension difference sequence based on a sliding observation window with a preset size, and extracting the statistical features of the resource dimension difference sequence in the sliding observation window; and performing weighted computation on the resource dimension difference sequence and the importance weight coefficient, and measuring by the weighted Euclidean distance to obtain the resource difference score.

[0133] Firstly, the time dimension cost matrix is constructed. The time sequence data of a to-be-analyzed system is collected, such as a response time sequence of a certain service. If the collected time sequence is T1 = [10, 12, 11, 15, 13], the reference time sequence will be T2= [9, 11, 12, 14, 16]. The deviation degree of corresponding time points in a time sequence T1 and a time sequence T2 is computed, for example, the deviation of a first time point is |10-9|=1. Moreover, in order to restrain the nonlinear distortion of the time sequence, the time jump penalty item is added. For example, if it is jumped from the first time point of the T1 to a second time point of the T2, one penalty coefficient will be added, and if the penalty coefficient is 0.5, the jump penalty item will be 0.5. A time point difference distance item and a time jump penalty item are added to form an element of the time dimension cost matrix. For example, if the cost of the first time point of the T1 and the first time point of the T2 is 1+0=1, the cost of the first time point of the T1 and the second time point of the T2 will be |10-11|+0.5=1.5. By parity of reasoning, the complete time dimension cost matrix can be constructed.

[0134] Thirdly, the cumulative cost computation is performed on the time dimension cost matrix. It starts from the upper left element in the cost matrix to compute cumulative cost of the elements one by one. The computation mode of the cumulative cost is as follows:

the cost of the current element is added with the minimum cumulative cost of the three adjacent elements on the left side, the upper portion and the upper left of the current element. For example, the cumulative cost of the upper left element in the cost matrix is its own cost. The cumulative cost of the second element is its own cost added with the cumulative cost of the upper left element. By parity of reasoning, the complete cumulative matrix can be computed.

[0135] Fourthly, the optimal matching path is obtained by the backtracking mode based on the accumulated cost matrix. It starts from the lower right element in the accumulated cost matrix to search the element with the minimum accumulated cost in the three adjacent elements on the left side, the upper portion and the upper left of the lower right element, and the position of this element is recorded. This process is repeated until the upper left element is traced back, and finally a path from the upper left to the lower right can be obtained, namely the optimal matching path.

[0136] Fifthly, normalization processing is performed on the accumulated cost at the optimal matching path to obtain the time dimension scoring sequence. The accumulated cost of each element at the optimal matching path is divided by the path length to obtain normalized cost, thus forming the time dimension scoring sequence.

[0137] Sixthly, the real-time resource occupation data of each node in the system is collected, including the processor occupation rate, the internal memory occupation rate and the network bandwidth occupation rate. It is assumed that the collected processor occupation rate is 70%, the collected memory occupation rate is 80%, and the collected network bandwidth occupation rate is 60%. The preset resource utilization rate thresholds are 80%, 90% and 70% respectively. The resource dimension difference sequences can be |70%-80%|=10%, |80%-90%|=10%, and |60%-70%|=10%, respectively.

[0138] Seventhly, the resource index judgment matrix is constructed by the analytic hierarchy process, and the feature vectors of the resource index judgment matrix are computed to obtain the importance weight coefficient corresponding to each resource index. It is assumed that the importance weight coefficients of the processor, the internal memory and the network bandwidth are 0.4, 0.3 and 0.3 respectively.

[0139] Eighthly, the resource dimension difference sequence is sampled based on the sliding observation window with the preset size. It is assumed that the size of the sliding observation window is 5, the resource dimension difference sequence is sample, and the statistical features in the window are extracted, such as an average value.

[0140] Ninthly, the resource dimension difference sequence and the importance weight coefficient are subjected to weighted computation, and the measurement is carried

out through the weighted Euclidean distances to obtain the resource difference score. Each element of the resource dimension difference sequence is multiplied by the corresponding weight coefficient, and then the three results are added to obtain the final resource difference score. For example, the resource difference score is 0.4 * 10% + 0.3 *10% + 0.3 * 10% = 10%.

[0141] Through the solution of the present invention:

[0142] the accuracy of anomaly detection can be improved: in comprehensive consideration of the time and resource dimension difference, the system anomalies can be identified more accurately. The false alarm rate can be reduced: by methods such as dynamic time warping and weighted Euclidean distances, the false alarm rate is effectively reduced. The robustness of anomaly detection can be enhanced: by the technologies such as sliding observation window and statistical feature extraction, the robustness of anomaly detection is enhanced.

[0143] In an optional embodiment, the distributing the optimized global model parameters to the swarm intelligence collaboration network, and dynamically adjusting the initial path convergence time threshold and the resource utilization rate threshold to form a new assessment reference for guiding the next round of query optimization includes:

[0144] constructing a topological structure of the swarm intelligence collaboration network, and computing an adjacency matrix based on a physical distance among the nodes and network link quality; dividing the adjacency matrix into a plurality of sub-network groups by a spectral clustering algorithm, and selecting a regional coordinator in each sub-network group based on the node computation capability and network connection stability;

[0145] collecting hardware configuration parameters and task load information of each node, and performing differential processing on the optimized global model parameters based on the hardware configuration parameters and the task load information to generate a parameter deployment scheme adapted to different nodes; constructing a parameter integrity check matrix by a bloom filter, and distributing the parameter deployment scheme to each node in the swarm intelligence collaboration network by the regional coordinator;

[0146] deploying a performance collection agent in the swarm intelligence collaboration network, and collecting the path convergence time and the resource utilization rate after the nodes execute the parameter deployment scheme; computing a deviation value between the path convergence time and the initial path convergence time threshold and a difference value between the resource utilization rate and the resource

utilization rate threshold based on the sliding time window;

[0147] inputting the deviation value and the difference value into an orthogonal test model, and analyzing the influence degree of the initial path convergence time threshold and the resource utilization rate threshold on the system performance; extracting a threshold combination by a principal component analysis method, and establishing a relationship between a mapping performance difference of a support vector regression model and a threshold adjustment amount;

[0148] constructing a dual Q learning network, treating the performance difference as a state input, and treating a threshold adjustment direction and a step length as action output; designing a reward function based on the performance improvement measurement, training a threshold dynamic adjustment strategy, and adaptively adjusting an exploration probability of the dual Q learning network according to the performance fluctuation degree;

[0149] performing exponential moving average processing on the performance data after the threshold dynamic adjustment strategy is executed, and generating a performance trend prediction value in combination with a time sequence prediction model; dynamically adjusting the initial path convergence time threshold and the resource utilization rate threshold based on the performance trend prediction value to form a new assessment reference; and

[0150] storing the performance data and the threshold dynamic adjustment strategy into an optimization knowledge base, and extracting an optimization rule to generate a scenario-based template; and taking the scenario-based template and the new assessment reference as a guidance basis for the next round of query optimization.

[0151] The swarm intelligence collaboration query optimization method aims to improve the query processing efficiency and the resource utilization rate in a distributed environment. According to the method, the path convergence time threshold and the resource utilization rate threshold are dynamically adjusted to optimize the query execution strategy in combination with the swarm intelligence collaboration network.

[0152] Firstly, the swarm intelligence collaboration network is constructed. The adjacency matrix is computed according to the physical distance among the nodes and the network link quality. For example, if the physical distance between a node A and a node B is 10 km, and the network link quality rating is 5 (ranges from 1 to 5, and 5 represents the best), the element values corresponding to the node A and the node B in the adjacency matrix will be 5/10= 0.5. Then the adjacency matrix is divided into a plurality of sub-network groups by the spectral clustering algorithm. It is assumed that the spectral clustering algorithm is utilized to divide the network into 3 groups. In each sub-network

group, the regional coordinator is selected according to the computing power of the node and the network connection stability. For example, in a group 1, the computing power of a node C is 1000 MIPS, and the network connection stability is 99%, and the node C is selected as the regional coordinator.

[0153] Secondly, the hardware configuration parameters and the task load information of each node are collected. For example, the hardware configuration parameters of a node D include 2.5 GHz of dominant frequency of the CPU, 8 GB of the internal memory, 500 G of the capacity of the hard disk, and 50% of the current task load. Based on such information, differential processing is performed on the optimized global model parameters to generate the parameter deployment scheme adapted to different nodes. For example, for nodes with low computing power, a smaller model parameter subset will be allocated. The bloom filter is utilized to construct the parameter integrity check matrix to ensure the consistency and integrity of parameter distribution. For example, Hash values of the model parameters are stored in the bloom filter, and after the nodes receive the parameters, the integrity of the parameters is judged by checking the Hash values. Finally, the parameter deployment scheme is distributed to each node in the swarm intelligence collaboration network by the regional coordinator.

[0154] The performance collection agent is deployed in the swarm intelligence collaboration network. The performance acquisition agent collects the path convergence time and the resource utilization rate after the nodes execute the parameter deployment scheme. For example, after a node E executes the query task, the path convergence time is 10 msec, the CPU utilization rate is 70%, and the memory utilization rate is 60%. Based on the sliding time window (for example, the window size is set to be 10 sec), the deviation value between the path convergence time and the initial path convergence time threshold, and the difference value between the resource utilization rate and the resource utilization rate threshold are computed. If the initial path convergence time threshold is 15 msec and the resource utilization rate threshold is 80%, the path convergence time deviation value of the node E will be -5 msec, and the resource utilization rate difference value will be -10%.

[0155] The deviation value and the difference value are inputted into the orthogonal test model, and the influence degree of the initial path convergence time threshold and the resource utilization rate threshold on the system performance is analyzed. For example, it is found through an orthogonal test that the system performance can be remarkably improved by reducing the path convergence time threshold, and the influence of increasing the resource utilization rate threshold on the system performance is small. The threshold combination is extracted by the principal component analysis method, and the relationship

between the mapping performance difference of the support vector regression model and the threshold adjustment amount is established. For example, prediction can be performed by the support vector regression model to obtain that if the path convergence time threshold is reduced by 2 msec, the system performance can be improved by 5%.

[0156] The dual Q learning network is constructed, the performance difference serves as the state input, and the threshold adjustment direction and the step length serve as the action output. For example, if the state is “path convergence time deviation value is -5 msec and resource utilization rate difference value is -10%”, possible actions include “reduce the path convergence time threshold by 2 msec, increase the resource utilization rate threshold by 5%”, etc. The reward function is designed based on the performance improvement measurement, for example, the larger the performance improvement is, the higher the reward value is. The threshold dynamic adjustment strategy is trained, and the exploration probability of the dual Q learning network is adaptively adjusted according to the performance fluctuation degree. For example, when the performance fluctuation is large, the exploration probability can be increased to search a better threshold adjustment strategy.

[0157] The exponential moving average processing is performed on the performance data after the threshold dynamic adjustment strategy is executed, and the performance trend prediction value is generated in combination with the time sequence prediction model. For example, it is predicted that the system performance will be improved by 10% within 5 min in the future. The initial path convergence time threshold and the resource utilization rate threshold are dynamically adjusted based on the performance trend prediction value to form the new assessment benchmark. For example, according to the prediction result, the path convergence time threshold is adjusted to be 13 msec and the resource utilization rate threshold is adjusted to be 75%.

[0158] The performance data and the threshold dynamic adjustment strategy are stored into the optimization knowledge base, and the optimization rule is extracted to generate the scenario-based template. For example, a corresponding threshold adjustment template is generated for a high-concurrency query scenario. The scenario-based template and the new assessment reference are treated as the guidance basis for the next round of query optimization.

[0159] Through the solution of the present invention:

[0160] the query efficiency can be improved: by dynamically adjusting the threshold and optimizing the query execution strategy, the query path convergence time is reduced, and thus the query efficiency is improved. The resource utilization can be optimized: in the

method, differential parameter deployment can be performed according to the hardware configuration and the task load conditions of the nodes, and the resource utilization rate threshold is dynamically adjusted, thereby optimizing the resource utilization and avoiding resource waste. The system adaptability can be enhanced: by the dual Q learning network and performance trend prediction, in the method, it can dynamically adjust parameters according to the system running state, thus the system adaptability is enhanced, and the system can better cope with different query loads and environment changes.

[0161] FIG. 2 is a schematic structural diagram of a swarm intelligence based federated learning collaboration system for an unstructured database according to an embodiment of the present invention, as shown in FIG. 2, the system includes:

[0162] a first unit, configured to collect query statement feature vectors, query resource consumption indexes, query response time indexes and query result accuracy indexes of a plurality of unstructured database nodes, and construct the collected data into an initial training sample set; input the initial training sample set into a preset deep neural network, perform training through a federated learning framework to obtain a query mode identification sub-model, a path programming sub-model and a resource allocation sub-model through, and generate corresponding initial model parameters; and establish a swarm intelligence collaboration network based on the initial model parameters, and set an initial path convergence time threshold and a resource utilization rate threshold as network assessment references;

[0163] a second unit, configured to transmit a query request inputted by a user into the query mode identification sub-model for semantic parsing, extract query semantic features, and generate a candidate query path by the path programming sub-model according to the initial path convergence time threshold; input the candidate query path into a hybrid swarm intelligence algorithm that fuses an ant colony path optimization mechanism and a particle swarm parameter optimization mechanism, and dynamically optimize to generate a query execution scheme based on the resource utilization rate threshold; perform distributed resource scheduling by the resource allocation sub-model according to the query execution scheme to form a final query execution scheme; and execute the final query execution scheme, and record actual path convergence time and resource occupation condition; and

[0164] a third unit, configured to perform differential analysis on the actual path convergence time and the initial path convergence time threshold, as well as the resource occupation condition and the resource utilization rate threshold to generate a performance optimization target; synchronously update local model parameters of each node based on the performance optimization target, fuse the updated local model parameters through a

security aggregation algorithm, and reconstruct to generate optimized global model parameters; and distribute the optimized global model parameters to the swarm intelligence collaboration network, and dynamically adjust the initial path convergence time threshold and the resource utilization rate threshold to form a new assessment reference for guiding the next round of query optimization.

[0165] In a third aspect,

[0166] an electronic device provided by the embodiment of the present invention includes:

[0167] a processor; and

[0168] a memory configured to store processor executable instructions;

[0169] the processor is configured to call instructions stored in the memory to execute the abovementioned method.

[0170] In a fourth aspect,

[0171] the embodiment of the present invention provides a computer-readable storage medium having computer program instructions stored therein, when executed by a processor, the computer program instructions execute the abovementioned method.

[0172] The present invention can be a method, an apparatus, a system, and/or a computer program product. The computer program product can include a computer-readable storage medium that contains computer-readable program instructions for executing the present invention in various aspects.

[0173] Finally, it is to be noted that: the above embodiments are only used for illustrating the technical solution of the present invention, not to limit it. Although the present invention is described in detail with reference to the aforementioned embodiments, those of ordinary skill in the art should understand that it may still modify the technical solutions recorded in the aforementioned embodiments, or replace some or all of the technical features thereof. These modifications or substitutions do not make the essence of the corresponding technical solution out of the scope of the technical solutions of each embodiment of the present invention.

CLAIMS

We claim:

1. A swarm intelligence based federated learning collaboration method for an unstructured database, comprising:

collecting query statement feature vectors, query resource consumption indexes, query response time indexes and query result accuracy indexes of a plurality of unstructured database nodes, and constructing the collected data into an initial training sample set; inputting the initial training sample set into a preset deep neural network, performing training through a federated learning framework to obtain a query mode identification sub-model, a path programming sub-model and a resource allocation sub-model through, and generating corresponding initial model parameters; establishing a swarm intelligence collaboration network based on the initial model parameters, and setting an initial path convergence time threshold and a resource utilization rate threshold as network assessment references;

transmitting a query request inputted by a user into the query mode identification sub-model for semantic parsing, extracting query semantic features, and generating a candidate query path by the path programming sub-model according to the initial path convergence time threshold; inputting the candidate query path into a hybrid swarm intelligence algorithm that fuses an ant colony path optimization mechanism and a particle swarm parameter optimization mechanism, and dynamically optimizing based on the resource utilization rate threshold to generate a query execution scheme; performing distributed resource scheduling by the resource allocation sub-model according to the query execution scheme to form a final query execution scheme; and executing the final query execution scheme, and recording actual path convergence time and resource occupation condition; and

performing differential analysis on the actual path convergence time and the initial path convergence time threshold, as well as the resource occupation condition and the resource utilization rate threshold to generate a performance optimization target; synchronously updating local model parameters of each node based on the performance optimization target, fusing the updated local model parameters through a security aggregation algorithm, and reconstructing to generate optimized global model parameters; and distributing the optimized global model parameters to the swarm intelligence collaboration network, and dynamically adjusting the initial path convergence time threshold and the resource utilization rate threshold to form a new assessment reference for

guiding the next round of query optimization.

2. The method according to claim 1, wherein the inputting the candidate query path into a hybrid swarm intelligence algorithm that fuses an ant colony path optimization mechanism and a particle swarm parameter optimization mechanism, and dynamically optimizing based on the resource utilization rate threshold to generate a query execution scheme comprises:

constructing an initial candidate query path set, mapping an execution sequence of each path in the initial candidate query path set into a path code in an integer sequence form, mapping a resource allocation strategy for query operation into a parameter code in a real number vector form, and establishing a double-layer path representation model for path structure and resource allocation; computing a query cost ratio of each candidate path based on the double-layer path representation model, and generating an initial path assessment index;

treating the ratio of the initial path assessment index to a preset reference threshold as path heuristic information to compute a path transfer probability matrix in combination with initial pheromone concentration; inputting the path transfer probability matrix into a hybrid swarm intelligence optimization model, the hybrid swarm intelligence optimization model fusing the ant colony path optimization mechanism and the particle swarm parameter optimization mechanism; generating a path structure optimization sequence by an ant colony algorithm based on the path transfer probability matrix; and

treating an assessment score of the path structure optimization sequence as a pheromone increment, and transforming the pheromone increment into a fitness function of the particle swarm algorithm; guiding the particle swarm algorithm to perform iterative optimization on the resource allocation parameters based on the fitness function, and generating a parameter optimization vector; collecting system resource utilization rate data, and when the resource utilization rate data exceeds a preset reference threshold, adaptively adjusting a value space of the parameter optimization vector according to the exceeding degree; and combining the path structure optimization sequence with the parameter optimization vector to form a complete query execution scheme.

3. The method according to claim 2, wherein the treating the ratio of the initial path assessment index to a preset reference threshold as path heuristic information to compute a path transfer probability matrix in combination with initial pheromone concentration; inputting the path transfer probability matrix into a hybrid swarm intelligence optimization model, the hybrid swarm intelligence optimization model fusing the ant colony path

optimization mechanism and the particle swarm parameter optimization mechanism; generating a path structure optimization sequence by an ant colony algorithm based on the path transfer probability matrix comprises:

computing the ratio of the initial path assessment index to the preset reference threshold to obtain the path heuristic information, the initial path assessment index comprising a path length index, a path energy consumption index and a path time index, and the preset reference threshold comprising a reference path length threshold, a reference path energy consumption threshold and a reference path time threshold;

performing combination operation on the path heuristic information and the initial pheromone concentration to obtain the path transfer probability matrix, the initial pheromone concentration being a fixed initial value, and a computing result of the path transfer probability matrix comprising a pheromone concentration weight coefficient and a heuristic information weight coefficient;

inputting the path transfer probability matrix into the hybrid swarm intelligence optimization model, the hybrid swarm intelligence optimization model fusing the ant colony path optimization mechanism and the particle swarm parameter optimization mechanism, the ant colony path optimization mechanism being used for path structure search, and the particle swarm parameter optimization mechanism being used for optimizing the pheromone concentration weight coefficient and the heuristic information weight coefficient;

generating the candidate path by the ant colony algorithm based on the path transfer probability matrix, computing a path assessment score corresponding to the candidate path, determining a pheromone release amount according to the path assessment score, and updating the path transfer probability matrix based on the pheromone release amount and a preset pheromone volatilization coefficient;

treating the path assessment score as the fitness value of the particle swarm algorithm, updating an individual optimal solution and a global optimal solution of a particle swarm based on the fitness value, and computing a particle velocity update amount and position update amount according to the individual optimal solution and the global optimal solution; and

treating the updated particle position as a weight coefficient configuration, optimizing the path transfer probability matrix, and re-inputting the optimized path transfer probability matrix into the ant colony path optimization mechanism for iteration until the path structure optimization sequence is obtained by convergence.

4. The method according to claim 1, wherein the performing distributed resource scheduling by the resource allocation sub-model according to the query execution scheme to form a final query execution scheme; and executing the final query execution scheme, and recording actual path convergence time and resource occupation condition comprises:

collecting processor performance indexes, internal memory capacity indexes, network bandwidth indexes and storage capacity indexes of each resource pool by the resource allocation sub-model to construct a resource feature matrix; computing resource pool scores according to the resource feature matrix, and performing initial resource allocation on the inputted query execution scheme based on the resource pool scores;

collecting real-time resource service data of each node by the resource allocation sub-model according to the initial resource allocation result, and computing node load states in combination with a resource type weight factor; when the node load state exceeds a preset load threshold, determining an optimal migration target node based on task migration cost and load balance income, and generating an optimized query execution scheme; and

performing distributed resource scheduling on the optimized query execution scheme by the resource allocation sub-model, and allocating execution resources based on the real-time computation capability, the storage capacity and the network state of each node, so as to form the final query execution scheme.

5. The method according to claim 1, wherein the performing differential analysis on the actual path convergence time and the initial path convergence time threshold, as well as the resource occupation condition and the resource utilization rate threshold to generate a performance optimization target; synchronously updating local model parameters of each node based on the performance optimization target, fusing the updated local model parameters through a security aggregation algorithm, and reconstructing to generate optimized global model parameters comprises:

constructing a path convergence time monitoring module and a resource occupation monitoring module, the path convergence time monitoring module collecting the actual path convergence time and comparing the actual path convergence time with a preset initial path convergence time threshold to generate a time dimension difference sequence, and the resource occupation monitoring module collecting resource occupation data of each node and comparing the resource occupation data with a preset resource utilization rate threshold to generate a resource dimension difference sequence;

performing dynamic time warping on the time dimension difference sequence,

constructing a time dimension cost matrix containing a deviation value of each time point, and solving an optimal matching path through a dynamic programming algorithm to obtain a time difference score; computing weighted Euclidean distances of a processor occupation rate, an internal memory occupation rate and a network bandwidth occupation rate for the resource dimension difference sequence, and obtaining a resource difference score in combination with a resource importance weight coefficient;

performing normalization processing on the time difference score and the resource difference score, and fusing based on a harmonic average method to generate a performance difference index; constructing a performance optimization target function containing stability constraints according to the performance difference index, and adaptively adjusting a weight coefficient of the performance optimization target function according to fluctuation degree of the time difference score and the resource difference score;

training local models in parallel at each node through the federated learning framework based on the performance optimization target function, updating the local model parameters by a stochastic gradient descent method with a momentum item, and dynamically adjusting the learning rate according to time difference scores in the training process; performing fragmentation processing on the updated local model parameters, and processing parameter fragments by a homomorphic encryption algorithm to generate encryption parameter blocks; and

allocating key fragments related to the resource difference score for each node, and combining the key fragments with stochastic mask information to generate a security aggregation key; performing distributed aggregation operation on the encryption parameter blocks through the security aggregation key to generate a global parameter aggregation matrix; computing node weight coefficients according to the time difference score and the resource difference score of each node, performing weighted averaging on the global parameter aggregation matrix based on the node weight coefficients, performing sparsification processing on the parameters subjected to weighted averaging, and reconstructing to generate the optimized global model parameters.

6. The method according to claim 5, wherein the performing dynamic time warping on the time dimension difference sequence, constructing a time dimension cost matrix containing a deviation value of each time point, and solving an optimal matching path through a dynamic programming algorithm to obtain a time difference score; computing weighted Euclidean distances of a processor occupation rate, an internal memory

occupation rate and a network bandwidth occupation rate for the resource dimension difference sequence, and obtaining a resource difference score in combination with a resource importance weight coefficient comprises:

constructing the time dimension cost matrix according to the time dimension difference sequence, each element of the time dimension cost matrix containing a time point difference distance item and a time jump penalty item, the time point difference distance item representing a deviation degree of a point corresponding to the time sequence, and the time jump penalty item restraining nonlinear distortion of the time sequence through a preset penalty coefficient;

performing cumulative cost computation on the time dimension cost matrix to obtain a cumulative cost matrix, and searching a minimum cumulative cost path by the cumulative cost matrix through the dynamic programming algorithm; obtaining the optimal matching path by a backtracking mode based on the minimum cumulative cost path, and performing normalization processing on the cumulative cost at the optimal matching path to obtain a time dimension scoring sequence;

collecting the real-time resource occupation data of each node in the system, comprising the processor occupation rate, the internal memory occupation rate and the network bandwidth occupation rate, and computing a difference value between the real-time resource occupation data and the preset resource utilization rate threshold to obtain the resource dimension difference sequence; constructing a resource index judgment matrix by an analytic hierarchy process, and computing the feature vectors of the resource index judgment matrix to obtain the importance weight coefficient corresponding to each resource index; and

sampling the resource dimension difference sequence based on a sliding observation window with a preset size, and extracting the statistical features of the resource dimension difference sequence in the sliding observation window; and performing weighted computation on the resource dimension difference sequence and the importance weight coefficient, and measuring by the weighted Euclidean distance to obtain the resource difference score.

7. The method according to claim 1, wherein the distributing the optimized global model parameters to the swarm intelligence collaboration network, and dynamically adjusting the initial path convergence time threshold and the resource utilization rate threshold to form a new assessment reference for guiding the next round of query optimization comprises:

constructing a topological structure of the swarm intelligence collaboration network, and computing an adjacency matrix based on a physical distance among the nodes and network link quality; dividing the adjacency matrix into a plurality of sub-network groups by a spectral clustering algorithm, and selecting a regional coordinator in each sub-network group based on the node computation capability and network connection stability;

collecting hardware configuration parameters and task load information of each node, and performing differential processing on the optimized global model parameters based on the hardware configuration parameters and the task load information to generate a parameter deployment scheme adapted to different nodes; constructing a parameter integrity check matrix by a bloom filter, and distributing the parameter deployment scheme to each node in the swarm intelligence collaboration network by the regional coordinator;

deploying a performance collection agent in the swarm intelligence collaboration network, and collecting the path convergence time and the resource utilization rate after the nodes execute the parameter deployment scheme; computing a deviation value between the path convergence time and the initial path convergence time threshold and a difference value between the resource utilization rate and the resource utilization rate threshold based on the sliding time window;

inputting the deviation value and the difference value into an orthogonal test model, and analyzing the influence degree of the initial path convergence time threshold and the resource utilization rate threshold on the system performance; extracting a threshold combination by a principal component analysis method, and establishing a relationship between a mapping performance difference of a support vector regression model and a threshold adjustment amount;

constructing a dual Q learning network, treating the performance difference as a state input, and treating a threshold adjustment direction and a step length as action output; designing a reward function based on the performance improvement measurement, training a threshold dynamic adjustment strategy, and adaptively adjusting an exploration probability of the dual Q learning network according to the performance fluctuation degree;

performing exponential moving average processing on the performance data after the threshold dynamic adjustment strategy is executed, and generating a performance trend prediction value in combination with a time sequence prediction model; dynamically adjusting the initial path convergence time threshold and the resource utilization rate threshold based on the performance trend prediction value to form a new assessment reference; and

storing the performance data and the threshold dynamic adjustment strategy into an optimization knowledge base, and extracting an optimization rule to generate a scenario-based template; and taking the scenario-based template and the new assessment reference as a guidance basis for the next round of query optimization.

8. A swarm intelligence based federated learning collaboration system for an unstructured database, applied to implement the method according to any one of claims 1 to 7, comprising:

a first unit, configured to collect query statement feature vectors, query resource consumption indexes, query response time indexes and query result accuracy indexes of a plurality of unstructured database nodes, and construct the collected data into an initial training sample set; input the initial training sample set into a preset deep neural network, perform training through a federated learning framework to obtain a query mode identification sub-model, a path programming sub-model and a resource allocation sub-model through, and generate corresponding initial model parameters; and establish a swarm intelligence collaboration network based on the initial model parameters, and set an initial path convergence time threshold and a resource utilization rate threshold as network assessment references;

a second unit, configured to transmit a query request inputted by a user into the query mode identification sub-model for semantic parsing, extract query semantic features, and generate a candidate query path by the path programming sub-model according to the initial path convergence time threshold; input the candidate query path into a hybrid swarm intelligence algorithm that fuses an ant colony path optimization mechanism and a particle swarm parameter optimization mechanism, and dynamically optimize to generate a query execution scheme based on the resource utilization rate threshold; perform distributed resource scheduling by the resource allocation sub-model according to the query execution scheme to form a final query execution scheme; and execute the final query execution scheme, and record actual path convergence time and resource occupation condition; and

a third unit, configured to perform differential analysis on the actual path convergence time and the initial path convergence time threshold, as well as the resource occupation condition and the resource utilization rate threshold to generate a performance optimization target; synchronously update local model parameters of each node based on the performance optimization target, fuse the updated local model parameters through a security aggregation algorithm, and reconstruct to generate optimized global model parameters; and distribute the optimized global model parameters to the swarm intelligence

collaboration network, and dynamically adjust the initial path convergence time threshold and the resource utilization rate threshold to form a new assessment reference for guiding the next round of query optimization.

9. An electronic device, comprising: a processor; and

a memory configured to store processor executable instructions;

the processor is configured to call instructions stored in the memory to execute the method according to any one of claims 1 to 7.

10. A computer-readable storage medium having computer program instructions stored therein, wherein when executed by a processor, the computer program instructions execute the method according to any one of claims 1 to 7.