

QUERY PROCESSING METHOD AND APPARATUS

[0001] The present application claims priority to Chinese Patent Application No. 202111110468.2, filed with the China National Intellectual Property Administration on September 23, 2021, and entitled "QUERY PROCESSING METHOD AND APPARATUS",
5 which is incorporated herein by reference in its entirety.

TECHNICAL FIELD

[0002] One or more implementations of the present specification relate to the field of data processing technologies, and in particular, to a query processing method and apparatus.

BACKGROUND

10 [0003] With the advent of the big data era, how to mine data values has become a current research hotspot. In a mining manner, statistical processing is performed on a large amount of data to provide a statistical query service to the outside. However, there is a risk of leakage of a single data record in the query service. For example, Zhang San finds that a mean of the first 500 lines of data is 20, and then finds that a mean of the first 501 lines of data is 20.1. In this
15 case, Zhang San can learn that a value of a 501st line of data is 70.1.

[0004] The differential privacy (DP) technology is used in a statistical query scenario to address the privacy leakage problem. A difficulty in this technology lies in how to balance privacy security of data and accuracy of a query result, because improvement of the former usually leads to reduction of the latter. Therefore, an improved differential privacy solution is
20 urgently needed to minimize a risk of privacy leakage of sensitive data while maximizing accuracy of a query result.

SUMMARY

[0005] One or more implementations of the present specification describe a query processing method and apparatus, to allocate, to each query in a batch of queries, a lowest noise
25 power that prevents data privacy from being leaked to maximize accuracy of a query result.

[0006] According to a first aspect, a query processing method for data privacy protection is

provided, including: determining query types of L queries to be performed on a target dataset, where the target dataset includes data of a business object; determining query sensitivity of each query type of the query types for the target dataset; and determining, based on the query sensitivity corresponding to each query and a privacy budget parameter predetermined for a total set of the L queries, a noise power allocated to each query.

[0007] In an implementation, the determining the query types of the L queries to be performed on the target dataset includes: receiving L query requests for the target dataset, where each query request indicates a query type of the query request.

[0008] In an implementation, the determining the query types of the L queries to be performed on the target dataset includes: obtaining a number L of queries that can be performed that is preconfigured for the target dataset and a query type of each of the queries.

[0009] In an implementation, the query type is any one of following: a counting query, a maximum value query, a minimum value query, a mean value query, or a variance query.

[0010] In an implementation, the business object is any one of following: a user, a commodity, or a business event.

[0011] In an implementation, the business event is any one of following: registration, access, login, or payment.

[0012] In an implementation, the business object is a user, and the data of the business event is any one of following: age, gender, income, interests and hobbies, physiological indicators, or operation indicators.

[0013] In an implementation, the determining the query sensitivity of each query type of the query types for the target dataset includes: for each query type of the query types, obtaining the query sensitivity corresponding to the query type based on a greatest absolute difference between a first result and a second result, where the first result is a result obtained by performing the type of query on the target dataset, and the second result is a result obtained by performing the type of query on an adjacent dataset of the target dataset.

[0014] In an implementation, the query types include a counting query, and the determining the query sensitivity of each query type of the query types for the target dataset includes: determining the query sensitivity of the counting query to be a value of 1.

[0015] In an implementation, the query types include a maximum value query or a minimum value query, and the determining the query sensitivity of each query type of the query types for the target dataset includes: determining a greatest value and a smallest value in the target dataset; and determining a result of subtracting the smallest value from the greatest value as the query sensitivity of the maximum value query or the minimum value query.

- 5 [0016] In an implementation, the query types include a mean value query, and the determining the query sensitivity of each query type of the query types for the target dataset includes: determining a greatest value in the target dataset; and determining a ratio between an absolute value of the greatest value and an amount of data in the target dataset plus 1 as the query sensitivity of the mean value query.
- 10 [0017] In an implementation, the query types include a variance query, and the determining the query sensitivity of each query type of the query types for the target dataset includes: determining a greatest value and a smallest value in the target dataset; and determining a product of following factors as the query sensitivity of the variance query: a square of a difference between the greatest value and the smallest value, an amount of data in the target dataset, and a reciprocal of a result obtained after a square operation is performed on the amount of data plus 1.
- 15 [0018] In an implementation, the determining, based on the query sensitivity corresponding to each query and the privacy budget parameter predetermined for the total set of the L queries, the noise power allocated to each query includes: determining a sum of the query sensitivity of the L queries based on the query sensitivity of each query; and for any query, determining, based on the query sensitivity of the query, the sum of the query sensitivity, and the privacy budget parameter, the noise power allocated to the query.
- 20 [0019] In an implementation, the determining, based on the query sensitivity of the query, the sum of the query sensitivity, and the privacy budget parameter, the noise power allocated to the query includes: obtaining a variable value of a mean variable, where the variable value is determined based on a parameter value of the privacy budget parameter and a constraint relationship between the privacy budget parameter and the mean variable in a Gaussian mechanism of differential privacy; and determining a product of following factors as the noise power of the query: the query sensitivity of the query, the sum of the query sensitivity, and a reciprocal of a result obtained after a square operation is performed on the variable value.
- 25 [0020] In an implementation, the privacy budget parameter includes a budget parameter and a relaxation parameter.
- [0021] In an implementation, after the determining the noise power allocated to each query, 30 the method further includes: for a target query in the L queries, determining an actually returned result of the target query as an original query result of the target query added with a target noise sampled from target noise distribution of differential privacy, where the target noise distribution is determined based on the noise power allocated to the target query.
- [0022] In an implementation, the target noise distribution is Gaussian noise distribution, and

the Gaussian noise distribution uses the noise power of the target query as a variance and uses 0 as a mean.

[0023] In an implementation, the method further includes: receiving a current query request for the target dataset, where the current query request corresponds to a current query type; determining whether a number of processed requests corresponding to the current query type is less than a predetermined threshold, where query requests corresponding to the number of processed requests are directed to the target dataset; and using the current query request as the target query in response to determining that the number of processed requests is less than the predetermined threshold.

[0024] According to a second aspect, a query processing apparatus for data privacy protection is provided, including: a query type determining unit, configured to determine query types of L queries to be performed on a target dataset, where the target dataset includes data of a business object; a sensitivity determining unit, configured to determine query sensitivity of each query type of the query types for the target dataset; and a noise power determining unit, configured to determine, based on the query sensitivity corresponding to each query and a privacy budget parameter predetermined for a total set of the L queries, a noise power allocated to each query.

[0025] In an implementation, the query type determining unit is configured to receive L query requests for the target dataset, where each query request indicates a query type of the query request.

[0026] In an implementation, the query type determining unit is configured to obtain a number L of queries that can be performed that is preconfigured for the target dataset and a query type of each of the queries.

[0027] In an implementation, the business object is any one of following: a user, a commodity, or a business event.

[0028] In an implementation, the sensitivity determining unit is configured to: for each query type of the query types, obtain the query sensitivity corresponding to the query type based on a greatest absolute difference between a first result and a second result, where the first result is a result obtained by performing the type of query on the target dataset, and the second result is a result obtained by performing the type of query on an adjacent dataset of the target dataset.

[0029] In an implementation, the apparatus further includes an actual result determining unit, configured to: for a target query in the L queries, determine an actually returned result of the target query as an original query result of the target query added with a target noise sampled from target noise distribution of differential privacy, where the target noise distribution is

determined based on the noise power allocated to the target query.

[0030] In an implementation, the apparatus further includes a target query determining unit, configured to: receive a current query request for the target dataset, where the current query request corresponds to a current query type; determine whether a number of processed requests
5 corresponding to the current query type is less than a predetermined threshold, where query requests corresponding to the number of processed requests are directed to the target dataset; and use the current query request as the target query in response to determining that the number of processed requests is less than the predetermined threshold.

[0031] According to a third aspect, a computer-readable storage medium is provided. The
10 computer-readable storage medium stores a computer program, and when the computer program is executed in a computer, the computer is enabled to perform the method provided in the first aspect.

[0032] According to a fourth aspect, a computing device is provided, including a memory and a processor. The memory stores executable code, and when executing the executable code,
15 the processor implements the method provided in the first aspect.

[0033] The method and the apparatus provided in the implementations of the present specification, and the query processing method for data privacy protection disclosed in the implementations of the present specification can be used to allocate, to each query in a batch of queries, a lowest noise power prevents data privacy from being leaked to maximize accuracy of
20 a query result.

BRIEF DESCRIPTION OF DRAWINGS

[0034] To describe the technical solutions in the implementations of the present invention more clearly, the following is a brief introduction of the accompanying drawings for illustrating such technical solutions. Clearly, the accompanying drawings described below are merely some
25 implementations of the present invention, and a person of ordinary skill in the art can still derive other drawings from such accompanying drawings without making innovative efforts.

[0035] FIG. 1 is a schematic diagram illustrating an implementation architecture of a query processing solution according to an implementation;

[0036] FIG. 2 is a schematic flowchart illustrating a query processing method according to
30 an implementation;

[0037] FIG. 3 is a schematic diagram illustrating an implementation architecture of a query processing solution according to an implementation;

[0038] FIG. 4 is a schematic diagram illustrating an implementation architecture of a query

processing solution according to still an implementation;

[0039] FIG. 5 is a schematic diagram illustrating an implementation architecture of a query processing solution according to yet an implementation; and

[0040] FIG. 6 is a schematic diagram illustrating a structure of a query processing apparatus according to an implementation.

DESCRIPTION OF EMBODIMENTS

[0041] The following describes the solutions provided in the present specification with reference to the accompanying drawings.

[0042] As described herein, the differential privacy (DP) technology may be used in a statistical query scenario, which addresses the privacy leakage problem. Generally, random noise is added to an original query result to prevent an actually returned query result from leaking original private data. However, it is difficult for existing solutions to achieve a balance between data security and accuracy of a returned result.

[0043] Based on this, implementations of the present specification disclose a query processing solution implemented based on differential privacy. An optimal noise power can be allocated to each query (or referred to as each time of query) in a batch of queries, so that even if an attacker obtains returned results of the batch of queries, the attacker cannot decipher the queried original data, and accuracy and availability of a returned result of each query are maximized.

[0044] FIG. 1 is a schematic diagram illustrating an implementation architecture of a query processing solution according to an implementation. As shown in FIG. 1, a service platform provides a query service. First, the service platform receives L query requests from a terminal, where FIG. 1 shows a first query request Q_1 and an L^{th} query request Q_L . Then, the service platform determines, based on a query type of each of the L query requests, query sensitivity of each query request for a target dataset. Afterwards, the service platform determines, based on the query sensitivity corresponding to each query request and a total number of privacy budget parameters predetermined for the L queries, a noise power allocated to each query request, where FIG. 1 shows a noise power P_{Q_1} and a noise power P_{Q_L} allocated to query request Q_1 and query request Q_L . Next, for a target query request (e.g., query request Q_1) in the L query requests, the service platform determines target noise distribution (e.g., noise distribution D_{Q_1}) of differential privacy based on the noise power (e.g., a noise power P_{Q_1}) allocated to the target query request, samples a target noise (e.g., a noise N_{Q_1}) from the target noise distribution, and adds the target noise to an original query result (e.g., an original query result R_1) of the target

query to obtain an actually returned result (e.g., an actually returned result R1') of the target query.

[0045] Example implementation steps of the example query processing solution are described below with reference to example implementations.

5 [0046] FIG. 2 is a schematic flowchart illustrating a query processing method according to an implementation. The method can be performed by any server, apparatus, platform, or device cluster that has a computing and processing capability, for example, can be performed by the service platform shown in FIG. 1 or a user terminal. As shown in FIG. 2, the method includes the following steps.

10 [0047] Step S210: Determine query types of L queries to be performed on a target dataset, where the target dataset includes data of a business object. Step S220: Determine query sensitivity of each query type of the query types for the target dataset. Step S230: Determine, based on the query sensitivity corresponding to each query and a privacy budget parameter predetermined for a total set of the L queries, a noise power allocated to each query.

15 [0048] The above steps are described in detail as follows.

[0049] First, in step S210, query types of L queries to be performed on a target dataset are determined. The target dataset includes a plurality of data records (or referred to as a plurality of data samples) corresponding to data of (or referred to as private data) of business objects. It should be noted that, business objects corresponding to any two data records may be the same
20 (e.g., corresponding to the same user), or may be different (e.g., corresponding to different commodities), and the plurality of data records correspond to the same feature of the business object (e.g., the age of a user). In addition, a storage environment of the plurality of data records is not limited. In a possible storage case, the plurality of data records are stored in a database. For example, a service platform pre-collects personal information authorized by a user, and
25 stores the personal information in the database. In an possible storage case, the plurality of data records are stored in a user terminal. For example, the user terminal (e.g., a smartphone) collects terminal operation data of a user (behavior data of inputting emoticons), and stores the terminal operation data.

[0050] In an implementation, the business object can be a user. In an example
30 implementation, the business object can be an individual user. Correspondingly, the plurality of data samples correspond to features of the individual user, such as the individual user's name, age, gender, income, identity card number, mobile phone number, email address, interests and hobbies, physiological indicators, or operation indicators. In an example, the physiological indicator is used to measure a human health condition, and can be, for example, blood pressure,

blood lipids, cholesterol, or blood oxygen concentration. In an example, the operation indicator corresponds to identified operation behavior, such as payment, click, or browsing, and is used to measure duration or a frequency of performing the specific operation behavior by the user. In an example implementation, the business object can be an enterprise user. Correspondingly, the plurality of data samples correspond to features of the enterprise user, such as the enterprise use's tax data, suppliers, flow funds on the book, or annual revenue.

[0051] In an implementation, the business object can be a commodity. For example, the commodity can be a physical commodity, such as a garage kit or a paper book; can be an empirical service, such as a tourism product, a massage service, a video membership service, a network course, or a payment membership service; or can be a virtual product, such as a game skin, an e-book, content information, a video resource, cloud disk space, or traffic. Correspondingly, the plurality of data samples correspond to features of the commodity, such as costs, an inventory, a sales volume, or a gross profit of the commodity.

[0052] In still an implementation, the business object can be a business event, such as a registration event, an access event, a login event, or a payment event. Correspondingly, the plurality of data samples correspond to event features, such as an occurrence moment, a network environment, a geographical location, or duration of the business event. In an example, the network environment can be an IP address, a network type, etc. It should be understood that the network type can include Wi-Fi, a 4G network, a 5G network, etc.

[0053] The target dataset is described herein by using the examples in which the business object is a user, a commodity, and a business event. It should be understood that the business object is not limited to these example types. The example query for the target dataset is a statistical query, which is used to query statistical information or aggregation information of the target dataset. For example, if the target dataset includes a plurality of user names, a number of users whose last name is Huang can be queried.

[0054] For the L queries and the query types of the queries, it should be noted that, L is a positive integer, and can have a value of 1, but is usually greater than 1. When $L > 1$, query types corresponding to any two queries may be the same or different. For example, several (i.e., one or more) query types related to the L queries can include a counting query, a maximum value query (or "maximum query"), a minimum value query (or "minimum query"), a mean value query (or "mean query"), and a variance query. In addition, in an implementation scenario, the L queries correspond to query information preconfigured for future query requests, including a number L of queries that can be performed and a query type of each of the queries. In an implementation scenario, the L queries depend on received L query requests, where each query

request indicates a query type of the query request. For detailed descriptions of the two implementation scenarios, references can be made to the following descriptions.

[0055] According to the descriptions herein, the query types of the L queries can be determined. Next, in step S220, query sensitivity of each of the several query types related to the L queries for the target dataset is determined.

[0056] It should be noted that, query sensitivity of a certain type is intended to reflect a maximum difference between a first result obtained by performing the type of query based on the target dataset and a second result obtained by performing the type of query based on an adjacent dataset of the target dataset. The adjacent dataset is a dataset that can be obtained after any piece of data is added to or deleted from the target dataset. A measurement criterion of the maximum difference measurement can be set based on an actual requirement, for example, the measurement criterion is quantized as a greatest value of an absolute value of a difference (or referred to as a greatest absolute difference in the present specification) between the first query result and the second query result. However, actually, not all maximum differences such as greatest absolute differences corresponding to all query types can be precisely solved. In this case, an approximate value or an estimated value of the maximum difference can be solved and used as the query sensitivity.

[0057] Calculation of the query sensitivity is described below with reference to specific examples of the query type.

[0058] According to an implementation, the several query types include a counting query. For the counting query, it should be noted that the plurality of data samples in the target dataset correspond to features of a business object, and value space of the feature includes a plurality of possible values, for example, possible values of a user gender include "male" and "female". Correspondingly, the plurality of values can be used as a plurality of data categories, so that the counting query can be performed on a certain data category, that is, a number of data samples corresponding to the data category is queried. For example, a number of female users in a user dataset is queried.

[0059] It can be learned from the description herein of the counting query that, when the counting query is performed on a certain data category, because a difference between a number of pieces of data in the target dataset and a number of pieces of data in an adjacent dataset of the target dataset is 1, and the difference of the piece of data either belongs to the data category or does not belong to the data category, an absolute value of a difference between two query results is either 1 or 0. Therefore, 1 can be determined as the query sensitivity of the counting query, which can be denoted as:

$$\text{sens}_c = 1 \quad (1)$$

[0060] As such, the query sensitivity sens_c corresponding to the counting query can be determined.

[0061] According to an implementation, the several query types include a maximum value query. For clear description, the target dataset is denoted as $x = (x_1, \dots, x_N)$ in the present specification, where N represents a total number of data samples, and x_i represents an i^{th} data sample. In addition, a result of sorting the N data samples in ascending order is denoted as $x_{(1)} \leq x_{(2)} \leq \dots \leq x_{(N)}$. It can be understood that $x_{(1)}$ and $x_{(N)}$ respectively represent a smallest value ("minimum value") and a greatest value ("maximum value") in the N data samples. Value space of a feature corresponding to the target dataset has a natural upper value bound and a lower value bound, or an employee can set the upper value bound and the lower value bound for the feature. The upper value bound and the lower value bound can be respectively denoted as l and u , and therefore $l \leq x_{(1)} \leq x_{(2)} \leq \dots \leq x_{(N)} \leq u$ can be obtained.

[0062] Based on the definition of the maximum value query type and the query sensitivity, in an example, the query sensitivity corresponding to the maximum value query can be determined as:

$$\text{sens}_{\max} = u - l \quad (2)$$

[0063] In an example, the query sensitivity corresponding to the maximum value query can be determined as:

$$\text{sens}_{\max} = x_{(N)} - x_{(1)} \quad (3)$$

[0064] As such, the query sensitivity $\text{sens}_{\max}$ corresponding to the maximum value query can be determined.

[0065] According to still an implementation, the several query types include a minimum value query. Similar to the determining the query sensitivity $\text{sens}_{\max}$ corresponding to the maximum value query, in an example, the query sensitivity corresponding to the minimum value query can be determined as:

$$\text{sens}_{\min} = u - l \quad (4)$$

[0066] In an example, the query sensitivity corresponding to the minimum value query can be determined as:

$$\text{sens}_{\min} = x_{(N)} - x_{(1)} \quad (5)$$

[0067] As such, the query sensitivity $\text{sens}_{\min}$ corresponding to the minimum value query can be determined.

[0068] According to yet an implementation, the several query types include a mean value

query. Mean calculation for the target dataset can be denoted as:

$$x_{\text{mean}} := f(x) = \frac{\sum_{i=1}^N x_i}{N} \quad (6)$$

[0069] It is assumed that data y is added to the adjacent dataset x' compared with the target dataset x , and $x_{(1)} \leq y \leq x_{(N)}$, that is, $x' = (x_1, \dots, x_N, y)$. Therefore, in an example, the query sensitivity corresponding to the mean value query can be calculated by using the following equation:

$$\begin{aligned} |f(x') - f(x)| &= \left| \frac{\sum_{i=1}^N x_i + y}{N+1} - \frac{\sum_{i=1}^N x_i}{N} \right| \\ &\leq \left| \frac{\sum_{i=1}^N x_i + y}{N+1} - \frac{\sum_{i=1}^N x_i}{N+1} \right| \\ &= \left| \frac{\sum_{i=1}^N x_i + y - \sum_{i=1}^N x_i}{N+1} \right| \\ &\leq \frac{|x_{(N)}|}{N+1} \end{aligned} \quad (7)$$

[0070] Therefore, the query sensitivity corresponding to the mean value query can be denoted as:

$$\text{sens}_{\text{mean}} = \frac{|x_{(N)}|}{N+1} \quad (8)$$

[0071] As such, the query sensitivity $\text{sens}_{\text{mean}}$ corresponding to the mean value query can be determined.

[0072] According to an implementation, the several query types include a variance query. A mean of the data samples in the target dataset x is denoted as μ , and therefore variance calculation for the target dataset x can be denoted as:

$$\begin{aligned} x_{\text{var}} := f(x) &= \frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2 \\ &= \frac{1}{2N^2} \sum_{i=1}^N \sum_{j=1}^N (x_i - x_j)^2 \end{aligned} \quad (9)$$

[0073] It is assumed that data y is added to the adjacent dataset x' compared with the target dataset x , and $x_{(1)} \leq y \leq x_{(N)}$, that is, $x' = (x_1, \dots, x_N, y)$. Therefore, the query sensitivity corresponding to the variance query can be calculated by using the following equation:

$$\begin{aligned}
|f(x') - f(x)| &= \left| \frac{\sum_{i=1}^{N+1} \sum_{j=1}^{N+1} (x_i - x_j)^2}{2(N+1)^2} - \frac{\sum_{i=1}^N \sum_{j=1}^N (x_i - x_j)^2}{2N^2} \right| \\
&\leq \left| \frac{\sum_{i=1}^{N+1} \sum_{j=1}^{N+1} (x_i - x_j)^2}{2(N+1)^2} - \frac{\sum_{i=1}^N \sum_{j=1}^N (x_i - x_j)^2}{2(N+1)^2} \right| \\
&= \left| \frac{2 \sum_{i=1}^N (x_i - y)^2}{2(N+1)^2} \right| \\
&\leq \frac{N(x_{(N)} - x_{(1)})^2}{(N+1)^2}
\end{aligned} \tag{10}$$

[0074] Therefore, the query sensitivity corresponding to the variance query can be denoted as:

$$\text{sens}_{\text{var}} = \frac{N}{(N+1)^2} (x_{(N)} - x_{(1)})^2 \tag{11}$$

5 **[0075]** As such, the query sensitivity sens_{var} corresponding to the variance query can be determined.

[0076] Calculation of the query sensitivity is described herein by using the examples in which the query type is the counting query, the maximum value query, the minimum value query, the mean value query, and the variance query. Actually, query types related to the L queries are not limited to these several example types, and can further include other types, such as a quantile query and a weight of evidence query. Query sensitivity corresponding to the other query types can be calculated with reference to the definition of the two queries, and exhaustive listing is not performed herein.

15 **[0077]** According to the descriptions herein, the query sensitivity corresponding to each query type of the query types related to the L queries can be determined. It should be understood that each query has a corresponding query type, and each query type of the query types has corresponding query sensitivity. Therefore, the query sensitivity corresponding to each query can be obtained.

20 **[0078]** Then, in step S230, a noise power allocated to each query is determined based on the query sensitivity corresponding to each query and the privacy budget parameter predetermined for the total set of the L queries.

25 **[0079]** In an implementation, this step can include following: For any ℓ^{th} query in the L queries, the noise power allocated to the query is determined based on query sensitivity sens_{ℓ} of the query, a sum of the query sensitivity of the L queries, and the privacy budget parameter. It should be noted that the sum of the query sensitivity is obtained by performing summation based on the query sensitivity of each of the L queries.

[0080] In an example implementation, a variable value of a mean variable μ is first determined based on a parameter value of the privacy budget parameter and a constraint relationship between the privacy budget parameter and the mean variable in a Gaussian mechanism of differential privacy. The constraint relationship is existing in the Gaussian mechanism of differential privacy, and can be expressed by the following equation:

$$\delta(\varepsilon; \mu) = \Phi\left(-\frac{\varepsilon}{\mu} + \frac{\mu}{2}\right) - e^{\varepsilon} \Phi\left(-\frac{\varepsilon}{\mu} - \frac{\mu}{2}\right) \quad (12)$$

[0081] In the equation (12), ε and δ respectively represent a budget parameter and a relaxation parameter in the privacy budget parameter, and parameter values of the two parameters can be manually set by an employee based on an actual requirement; μ represents the mean variable; and $\Phi(t)$ represents a probability distribution function of standard Gaussian distribution, where $\Phi(t) = \mathbb{P}[\mathcal{N}(0, 1) \leq t] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^t e^{-y^2/2} dy$.

[0082] Further, for any ℓ^{th} query, a product of following factors is determined as the noise power of the query: the query sensitivity $sens_{\ell}$, the sum of the query sensitivity, and a reciprocal of a result obtained after a square operation is performed on the variable value of the mean variable. Therefore, an equation for calculating the noise power can be expressed as:

$$\sigma_{\ell, opt}^2 = \frac{sens_{\ell}}{\mu^2} \left(\sum_{k=1}^L sens_k \right), \quad \ell = 1, \dots, L \quad (13)$$

[0083] In the equation (13), $\sigma_{\ell, opt}^2$ represents the noise power of the ℓ^{th} query, $sens_{\ell}$ represents the query sensitivity of the ℓ^{th} query, μ represents the mean variable, and $\sum_{k=1}^L sens_k$ represents the sum of the query sensitivity of the L queries.

[0084] It should be noted that a derivation process of equation (13) is as follows.

[0085] A total noise power of the L queries is minimized under a privacy constraint, which is equivalent to the following optimization problem:

$$\text{Constraint function: } \sum_{k=1}^L \frac{S_k^2}{\sigma_k^2} \leq \mu^2 \quad (14)$$

$$\text{Original function that needs to be solved: } \begin{cases} \text{minimize} \\ \{\sigma_k^2 > 0\}_{k=1}^L \end{cases} \sum_{k=1}^L \sigma_k^2 \quad (15)$$

[0086] In equations (14) and (15), S_k represents the query sensitivity of the k^{th} query, μ represents the mean variable, and σ_k^2 represents the noise power of the k^{th} query.

[0087] It should be noted that the optimization problem is a convex optimization problem

and satisfies the Slater's condition, and therefore has strong duality. For this optimization problem, a Lagrange multiplier λ is introduced, and the following Lagrange $\mathcal{L}$ is constructed, to associate the constraint function with the original function:

$$\mathcal{L}(\{\sigma_k^2\}_{k=1}^L, \lambda) = \sum_{k=1}^L \sigma_k^2 + \lambda \left(\sum_{k=1}^L \frac{S_k^2}{\sigma_k^2} - \mu^2 \right) \quad (16)$$

5 **[0088]** A feasible solution σ_k^2 of equation (16) needs to satisfy the Karush-Kuhn-Tucker (KKT) condition:

$$\begin{cases} \frac{\partial \mathcal{L}}{\partial \sigma_\ell^2} = 1 - \lambda \frac{S_\ell^2}{\sigma_\ell^4}, & \ell = 1, \dots, L \\ \lambda \left(\sum_{k=1}^L \frac{S_k^2}{\sigma_k^2} - \mu^2 \right) = 0 \end{cases} \quad (17)$$

[0089] Therefore, if Lagrange $\mathcal{L}$ is minimized when the KKT condition is satisfied, the following optimal noise power allocation policy can be obtained:

$$\sigma_{\ell, \text{opt}}^2 = \frac{S_\ell}{\mu^2} \left(\sum_{k=1}^L S_k \right) \quad (18)$$

[0090] It should be understood that equation (18) is equivalent to equation (13), except that the notation of mathematical symbols is slightly different.

15 **[0091]** According to the descriptions herein, the noise power allocated to each query in the Gaussian noise mechanism of differential privacy can be determined. It should be noted that, actually, the noise power of each query in other noise mechanisms, such as a Laplace mechanism and an exponential mechanism, can also be determined.

20 **[0092]** According to an implementation of an aspect, after step S230, the method can further include following: In step S240, for a target query in the L queries, an actually returned result of the target query is determined as an original query result of the target query added with a target noise sampled from target noise distribution of differential privacy, where the target noise distribution is determined based on the noise power allocated to the target query.

25 **[0093]** In an implementation, the target noise distribution can be Gaussian noise distribution. Correspondingly, the noise power of the target query is used as a variance of Gaussian distribution and 0 is used as a mean to generate corresponding Gaussian noise distribution. For clear description, in the following, the target query is denoted as the ℓ^{th} query and the noise power of the target query is denoted as $\sigma_{\ell, \text{opt}}^2$. Therefore, the Gaussian noise distribution corresponding to the target query can be denoted as $\mathcal{N}(0, \sigma_{\ell, \text{opt}}^2)$.

[0094] Based on the determined Gaussian noise distribution $\mathcal{N}(0, \sigma_{\ell, \text{opt}}^2)$, the target noise

$\mathcal{Z}$ can be sampled from the Gaussian noise distribution and added to the original query result q_ℓ of the target query, to obtain a corresponding actually returned result. An equation is as follows:

$$\tilde{q}_\ell = q_\ell + z, \quad z \sim \mathcal{N}(0, \sigma_{\ell, \text{opt}}^2) \quad (19)$$

5 **[0095]** Equation (19) is described in detail with reference to examples of the query type.

[0096] According to an implementation, if the query type of the target query is a counting query, an actually returned counting query result can be calculated based on the following equation:

$$\widetilde{\text{Count}}_{c,N} = \lfloor \text{Count}_{c,N} + z_c \rfloor \quad c = 1, \dots, C \quad (20)$$

10 **[0097]** In the equation (20), N represents the total number of data samples in the target dataset, c represents a c^{th} data category, C represents a total number of data categories, z_c represents noise randomly sampled from Gaussian noise distribution corresponding to the counting query, $\text{Count}_{c,N}$ represents an original query result of the counting query for the c^{th} category, $\widetilde{\text{Count}}_{c,N}$ represents an actually returned result obtained after the noise is added, and
15 the operator $\lfloor \cdot \rfloor$ represents rounding to obtain an integer.

[0098] As such, the actually returned result corresponding to the counting query type can be determined. It should be understood that, when an actually returned result corresponding to each query is determined, noise sampling is performed based on corresponding Gaussian noise distribution. For example, for two queries of the same type, random sampling is performed
20 twice based on the same Gaussian noise distribution corresponding to the two queries, and then the sampled noise is separately added to original query results corresponding to the query types, to correspondingly obtain two actually returned results.

[0099] According to an implementation, if the query type of the target query is a maximum value query, an actually returned maximum value query result can be calculated based on the
25 following equation:

$$\tilde{x}_{\max} = x_{(N)} + z \quad (21)$$

[00100] In the equation (21), $\mathcal{Z}$ represents noise randomly sampled from Gaussian noise distribution corresponding to the maximum value query; $x_{(N)}$ represents an original query result of the maximum value query, i.e., the greatest value in the target dataset; and $\tilde{x}_{\max}$
30 represents an actually returned result corresponding to the maximum value query.

[00101] According to still an implementation, if the query type of the target query is a minimum value query, an actually returned minimum value query result can be calculated based on the following equation:

$$\tilde{x}_{\min} = x_{(1)} + z \quad (22)$$

[00102] In the equation (22), z represents noise randomly sampled from Gaussian noise distribution corresponding to the minimum value query; $x_{(1)}$ represents an original query result of the minimum value query, i.e., a smallest value in the target dataset; and $\tilde{x}_{\min}$ represents an actually returned result corresponding to the minimum value query.

[00103] According to an implementation, if the query type of the target query is a mean value query, an actually returned mean value query result can be calculated based on the following equation:

$$\tilde{x}_{\text{mean}} = \frac{\sum_{i=1}^N x_i}{N} + z \quad (23)$$

[00104] In the equation (23), z represents noise randomly sampled from Gaussian noise distribution corresponding to the mean value query, $\frac{\sum_{i=1}^N x_i}{N}$ represents a mean of the target dataset, and $\tilde{x}_{\text{mean}}$ represents an actually returned result corresponding to the mean value query.

[00105] According to an implementation, if the query type of the target query is a variance query, an actually returned variance query result can be calculated based on the following equation:

$$\tilde{x}_{\text{var}} = \sum_{i=1}^N (x_i - \mu)^2 + z \quad (24)$$

[00106] In the equation (24), z represents noise sampled from Gaussian noise distribution corresponding to the variance query, $\sum_{i=1}^N (x_i - \mu)^2$ represents a variance of the target dataset, and $\tilde{x}_{\text{var}}$ represents an actually returned result corresponding to the variance query.

[00107] Sampling and addition of the target noise is described herein by using the Gaussian mechanism of differential privacy as an example. In an implementation, the target noise power is determined based on a Laplace mechanism of differential privacy. Correspondingly, the noise power of the target query can be used as a scale parameter of Laplace distribution and 0 can be used as a location parameter to generate corresponding Laplace noise distribution, to sample and add the target noise to obtain an actually returned result. In an implementation, the target noise can be sampled and added based on an exponential mechanism of differential privacy to obtain an actually returned result.

[00108] According to the description herein, an actually returned result corresponding to each target query in the L queries can be determined.

[00109] It should be noted that the query processing method disclosed in this implementation of the present specification can be applied to a plurality of implementation scenarios. In a

typical scenario, as shown in FIG. 1, a batch of to-be-processed query requests is first received. Then, in step S210, a number L of to-be-processed queries and a query type of each of the to-be-processed queries is determined based on the batch of query requests. In an implementation, a plurality of query requests received in a predetermined time period (for example, in a previous day or in last 10 min) can be obtained, a number of the plurality of query requests is denoted as L , and a query type indicated by each query request is determined. Afterwards, in step S240, the L query requests are respectively used as target queries to correspondingly sample noise and perform noise superposition based on an original query result, to obtain an actually returned result of each query request. As such, differential privacy processing for a batch of query requests can be completed.

[00110] In an implementation scenario, for query requests that may be received in the future, a number L of queries that can be performed and a query type of each of the queries can be preconfigured. Therefore, this part of preconfigured information can be directly obtained in step S210. It should be noted that, configuration of the number of queries and the query type can be determined by an employee by analyzing historical query data corresponding to the target dataset.

[00111] Further, in a detailed implementation scenario, as shown in FIG. 3, in a preparation phase, the above steps S210, S220, and S230 are performed to determine a noise power allocated to each query. Then, in an online processing phase, step S240 is performed based on a received current query request Q_j . For example, the current query request Q_j is used as the target query, which corresponds to a current query type C_i . When it is determined that a number of processed requests for the target dataset corresponding to the current query type C_i is less than a predetermined number L_{C_i} of queries that can be performed, the current query request Q_j is used as the target query. In this case, an original query result F_{C_i} corresponding to the query type C_i is determined based on the target dataset and a noise power P_{C_i} corresponding to the current query type C_i determined in the preparation phase is obtained, and noise N_j sampled from noise distribution determined based on the noise power P_{C_i} is further added to the original query result F_{C_i} to obtain an actually returned result F_j . Alternatively, if it is determined that the number of processed requests is equal to the number L_{C_i} of queries that can be performed, the current query request Q_j is discarded.

[00112] In an implementation scenario, as shown in FIG. 4, in a preparation phase, the above steps S210, S220, and S230 are performed to determine a noise power allocated to each query, and further determine an original query result corresponding to each query type of the query types based on the target dataset. Then, in an online processing phase, step S240 is performed

based on a received current query request Q_j . For example, the current query request Q_j is used as the target query, which corresponds to a current query type C_i . When it is determined that a number of processed requests for the target dataset corresponding to this type is less than a predetermined number L_{C_i} of queries that can be performed, the current query request Q_j is used as the target query. In this case, an original query result F_{C_i} and a noise power P_{C_i} corresponding to the current query type C_i determined in the preparation phase are obtained, and noise N_j sampled from noise distribution determined based on the noise power P_{C_i} is further added to the original query result F_{C_i} to obtain an actually returned result F_j . Otherwise, the current query request Q_j is discarded.

[00113] In an implementation scenario, as shown in FIG. 5, in a preparation phase, step S210 to step S240 are performed to obtain an actually returned result corresponding to each of the L queries. Afterwards, in an online processing phase, a current query request Q_i is received, which corresponds to a current query type C_i . When it is determined that a number of processed requests for the target dataset corresponding to this type is less than a predetermined number of queries that can be performed, an actually returned result that corresponds to the current query type C_i and that is not yet used is obtained from L actually returned results determined in the preparation phase and is used as an actually returned result R_i of the current query request Q_i . Otherwise, the current query request Q_i is discarded.

[00114] Implementation scenarios of the query processing method disclosed in the implementations of the present specification are described above.

[00115] In conclusion, the query processing method for data privacy protection disclosed in the implementations of the present specification can be used to allocate, to each query in a batch of queries, a lowest noise power that still can prevent data privacy from being leaked and can maximize accuracy of a query result.

[00116] Corresponding to the above query processing method, the implementations of the present specification further disclose a query processing apparatus. FIG. 6 is a schematic diagram illustrating a structure of a query processing apparatus according to an implementation. The apparatus can be implemented as any computing unit, platform, server, device cluster that has a computing and processing capability, for example, can be a user terminal or the service platform shown in FIG. 1. As shown in FIG. 6, the apparatus 600 includes the following units:

a query type determining unit 610, configured to determine query types of L queries to be performed on a target dataset, where the target dataset includes data of a business object; a sensitivity determining unit 620, configured to determine query sensitivity of each query type of the query types for the target dataset; and a noise power determining unit

630, configured to determine, based on the query sensitivity corresponding to each query and a privacy budget parameter predetermined for a total set of the L queries, a noise power allocated to each query.

5 [00117] In an implementation, the query type determining unit 610 is for example configured to receive L query requests for the target dataset, where each query request indicates a query type of the query request.

[00118] In an implementation, the query type determining unit 610 is for example configured to obtain a number L of queries that can be performed that is preconfigured for the target dataset and a query type of each of the queries.

10 [00119] In an implementation, the query type is any one of following: a counting query, a maximum value query, a minimum value query, a mean value query, or a variance query.

[00120] In an implementation, the business object is any one of following: a user, a commodity, or a business event.

15 [00121] In an implementation, the business event is any one of following: registration, access, login, or payment.

[00122] In an implementation, the business object is a user, and the data of the business event is any one of following: age, gender, income, interests and hobbies, physiological indicators, or operation indicators.

20 [00123] In an implementation, the sensitivity determining unit 620 is, for example, configured to: for each query type of the query types, obtain the query sensitivity corresponding to the query type based on a greatest absolute difference between a first result and a second result, where the first result is a result obtained by performing the type of query on the target dataset, and the second result is a result obtained by performing the type of query on an adjacent dataset of the target dataset.

25 [00124] In an implementation, the query type includes a counting query, and the sensitivity determining unit 620 is, for example, configured to determine the query sensitivity of the counting query to be a value of 1.

30 [00125] In an implementation, the query types include a maximum value query or a minimum value query, and the sensitivity determining unit 620 is, for example, configured to: determine a greatest value and a smallest value in the target dataset; and determine a result of subtracting the smallest value from the greatest value as the query sensitivity of the maximum value query or the minimum value query.

[00126] In an implementation, the query types include a mean value query, and the sensitivity determining unit 620 is, for example, configured to: determine a greatest value in the target

dataset; and determine a ratio between an absolute value of the greatest value and an amount of data in the target dataset plus 1 as the query sensitivity of the mean value query.

[00127] In an implementation, the query types include a variance query, and the sensitivity determining unit 620 is, for example, configured to: determine a greatest value and a smallest value in the target dataset; and determine a product of following factors as the query sensitivity of the variance query: a square of a difference between the greatest value and the smallest value, an amount of data in the target dataset, and a reciprocal of a result obtained after a square operation is performed on the amount of data plus 1.

[00128] In an implementation, the noise power determining unit 630 is, for example, configured to: determine a sum of the query sensitivity of the L queries based on the query sensitivity of each query; and for any query, determine, based on the query sensitivity of the query, the sum of the query sensitivity, and the privacy budget parameter, the noise power allocated to the query.

[00129] In an example implementation, the noise power determining unit 630 is further configured to: obtain a variable value of a mean variable, where the variable value is determined based on a parameter value of the privacy budget parameter and a constraint relationship between the privacy budget parameter and the mean variable in a Gaussian mechanism of differential privacy; and determine a product of following factors as the noise power of the query: the query sensitivity of the query, the sum of the query sensitivity, and a reciprocal of a result obtained after a square operation is performed on the variable value.

[00130] In an implementation, the privacy budget parameter includes a budget parameter and a relaxation parameter.

[00131] In an implementation, the apparatus 600 further includes an actual result determining unit 640, configured to: for a target query in the L queries, determine an actually returned result of the target query as an original query result of the target query added with a target noise sampled from target noise distribution of differential privacy, where the target noise distribution is determined based on the noise power allocated to the target query.

[00132] In an example implementation, the target noise distribution is Gaussian noise distribution, and the Gaussian noise distribution uses the noise power of the target query as a variance and uses 0 as a mean.

[00133] In an aspect, in an example implementation, the apparatus further includes a target query determining unit 650, configured to: receive a current query request for the target dataset, where the current query request corresponds to a current query type; determine whether a number of processed requests corresponding to the current query type is less than a

predetermined threshold, where query requests corresponding to the number of processed requests are directed to the target dataset; and use the current query request as the target query in response to determining that the number of processed requests is less than the predetermined threshold.

5 **[00134]** According to an implementation of an aspect, a computer-readable storage medium is further provided. The computer-readable storage medium stores a computer program, and when the computer program is executed in a computer, the computer is enabled to perform the method described with reference to FIG. 2.

10 **[00135]** According to an implementation of an aspect, a computing device is further provided, including a memory and a processor. The memory stores executable code, and when executing the executable code, the processor implements the method described with reference to FIG. 2.

[00136] A person skilled in the art should be aware that, in the above one or more examples, the functions described in the present invention can be implemented by hardware, software, firmware, or any combination thereof. When the functions are implemented by software, the
15 functions can be stored in a computer-readable medium or transmitted as one or more instructions or code in the computer-readable medium.

[00137] The characteristics, technical solutions, and beneficial effects of the present invention are further described in detail in the above example implementations. It should be understood that the above descriptions are merely example implementations of the present
20 invention, but are not intended to limit the protection scope of the present invention. Any modification, equivalent replacement, or improvement made based on the technical solutions in the present invention shall fall within the protection scope of the present invention.

CLAIMS

What is claimed is:

1. A query processing method, comprising:

determining query types of L queries to be performed on a target dataset, the target dataset
5 including data of an object;

determining query sensitivity of each query type of the query types for the target dataset; and

determining, based on query sensitivity corresponding to each query of the L queries and a
privacy budget parameter predetermined for a total set of the L queries, a noise power allocated to
each query of the L queries.

10

2. The method according to claim 1, wherein the determining the query types of the L queries to
be performed on the target dataset includes:

receiving L query requests for the target dataset, wherein each query request indicates a query
type of the query request.

15

3. The method according to claim 1, wherein the determining the query types of the L queries to
be performed on the target dataset includes:

obtaining a preconfigured number L of queries performable on the target dataset and a query
type of each of the queries.

20

4. The method according to claim 1, wherein a query type is any one of following: a counting
query, a maximum value query, a minimum value query, a mean value query, or a variance query.

5. The method according to claim 1, wherein the object is any one of following: a user, a
25 commodity, or a business event.

6. The method according to claim 5, wherein the business event is any one of following:
registration, access, login, or payment.

30

7. The method according to claim 1, wherein the object is a user, and the data of the object is
any one of following: age, gender, income, interests and hobbies, physiological indicators, or
operation indicators.

8. The method according to claim 1, wherein the determining the query sensitivity of each query

type of the query types for the target dataset includes:

for each query type of the query types, obtaining the query sensitivity corresponding to the query type based on a greatest absolute difference between a first result and a second result, wherein the first result is a result obtained by performing the type of query on the target dataset, and the second result is a result obtained by performing the type of query on an adjacent dataset of the target dataset.

9. The method according to claim 1 or 8, wherein the query types include a counting query, and the determining the query sensitivity of each query type of the query types for the target dataset includes:

determining the query sensitivity of the counting query to be a value of 1.

10. The method according to claim 1 or 8, wherein the query types include a maximum value query or a minimum value query, and the determining the query sensitivity of each query type of the query types for the target dataset includes:

determining a greatest value and a smallest value in the target dataset; and

determining a result of subtracting the smallest value from the greatest value as the query sensitivity of the maximum value query or the minimum value query.

11. The method according to claim 1 or 8, wherein the query types include a mean value query, and the determining the query sensitivity of each query type of the query types for the target dataset includes:

determining a greatest value in the target dataset; and

determining a ratio between an absolute value of the greatest value and an amount of data in the target dataset plus 1 as the query sensitivity of the mean value query.

12. The method according to claim 1 or 8, wherein the query types include a variance query, and the determining the query sensitivity of each query type of the query types for the target dataset includes:

determining a greatest value and a smallest value in the target dataset; and

determining a product of following factors as the query sensitivity of the variance query: a square of a difference between the greatest value and the smallest value, an amount of data in the target dataset, and a reciprocal of a result obtained after a square operation is performed on the amount of data plus 1.

13. The method according to claim 1, wherein the determining, based on the query sensitivity corresponding to each query of the L queries and the privacy budget parameter predetermined for the total set of the L queries, the noise power allocated to each query of the L queries includes:

determining a sum of query sensitivity of the L queries based on the query sensitivity of each query; and

for any query, determining the noise power allocated to the query based on the query sensitivity of the query, the sum of the query sensitivity, and the privacy budget parameter,.

14. The method according to claim 13, wherein the determining the noise power allocated to the query based on the query sensitivity of the query, the sum of the query sensitivity, and the privacy budget parameter includes:

obtaining a variable value of a mean variable, wherein the variable value is determined based on a parameter value of the privacy budget parameter and a constraint relationship between the privacy budget parameter and the mean variable in a Gaussian mechanism of differential privacy; and

determining a product of following factors as the noise power of the query: the query sensitivity of the query, the sum of the query sensitivity, and a reciprocal of a result obtained after a square operation is performed on the variable value.

15. The method according to claim 1 or 14, wherein the privacy budget parameter includes a budget parameter and a relaxation parameter.

16. The method according to claim 1, further comprising: after the determining the noise power allocated to each query allocation,

for a target query in the L queries, determining an actually returned result of the target query by adding to an original query result of the target query a target noise sampled from target noise distribution of differential privacy, wherein the target noise distribution is determined based on a noise power allocated to the target query.

17. The method according to claim 16, wherein the target noise distribution is Gaussian noise distribution, and the Gaussian noise distribution uses the noise power of the target query as a variance and uses 0 as a mean.

18. The method according to claim 16, further comprising:

receiving a current query request for the target dataset, wherein the current query request

corresponds to a current query type;

determining whether a number of processed requests corresponding to the current query type is less than a predetermined threshold, wherein query requests corresponding to the number of processed requests are directed to the target dataset; and

5 using the current query request as the target query in response to determining that the number of processed requests is less than the predetermined threshold.

19. A query processing apparatus, comprising:

10 a query type determining unit, configured to determine query types of L queries to be performed on a target dataset, the target dataset including data of a business object;

a sensitivity determining unit, configured to determine query sensitivity of each query type of the query types for the target dataset; and

15 a noise power determining unit, configured to determine, based on the query sensitivity corresponding to each query and a privacy budget parameter predetermined for a total set of the L queries, a noise power allocated to each query.

20. The apparatus according to claim 19, wherein the query type determining unit is configured to:

20 receive L query requests for the target dataset, wherein each query request indicates a query type of the query request.

21. The apparatus according to claim 19, wherein the query type determining unit is configured to:

25 obtain a number L of queries that can be performed that is preconfigured for the target dataset and a query type of each of the queries.

22. The apparatus according to claim 19, wherein the sensitivity determining unit is configured to:

30 for each query type of the query types, obtain the query sensitivity corresponding to the query type based on a greatest absolute difference between a first result and a second result, wherein the first result is a result obtained by performing the type of query on the target dataset, and the second result is a result obtained by performing the type of query on an adjacent dataset of the target dataset.

23. The apparatus according to claim 19, further comprising:

an actual result determining unit, configured to: for a target query in the L queries, determine an actually returned result of the target query as an original query result of the target query added with a target noise sampled from target noise distribution of differential privacy, wherein the target noise distribution is determined based on the noise power allocated to the target query.

5

24. The apparatus according to claim 23, further comprising a target query determining unit, configured to:

receive a current query request for the target dataset, wherein the current query request corresponds to a current query type;

10

determine whether a number of processed requests corresponding to the current query type is less than a predetermined threshold, wherein query requests corresponding to the number of processed requests are directed to the target dataset; and

use the current query request as the target query in response to determining that the number of processed requests is less than the predetermined threshold.

15

25. A computer-readable storage medium, the computer-readable storage medium storing a computer program, wherein when the computer program is executed in a computer, the computer is enabled to perform the method according to any one of claims 1 to 18.