

COLUMN DATA COMPRESSION

TECHNICAL FIELD

[0001] One or more embodiments of this specification relate to the data processing field, and in particular, to a column data compression method and apparatus, and a storage medium.

5

BACKGROUND

[0002] Currently, data in a database can be stored in a column storage manner. To be specific, data in the same column in a database table can be continuously stored together. There is a higher data compression rate in comparison with a manner in which storage is performed by row. In addition, there is also higher read efficiency in a scenario in which access is performed by column.

10 [0003] To save storage space, column data can be compressed, but compression efficiency may be relatively low.

SUMMARY

[0004] One or more embodiments of this specification provide a column data compression method and apparatus, and a storage medium.

15 [0005] To achieve the above-mentioned objectives, one or more embodiments of this specification provide the following technical solutions.

[0006] According to a first aspect of one or more embodiments of this specification, a column data compression method is proposed, including: separately encoding the column data in a plurality of encoding schemes at a column level, to obtain at least one data stream corresponding to each encoding scheme; determining a target encoding scheme in which smallest storage space is occupied in the plurality of encoding schemes; determining a target compression algorithm with the highest compression rate at a stream level; and compressing at least one target data stream corresponding to the target encoding scheme based on the target compression algorithm, and storing compressed target data stream in a database.

20

25 [0007] According to a second aspect of one or more embodiments of this specification, a

- column data compression apparatus is provided, including: an encoding module, configured to separately encode the column data in a plurality of encoding schemes at a column level, to obtain at least one data stream corresponding to each encoding scheme; a first determining module, configured to determine a target encoding scheme in which smallest storage space is occupied in the plurality of encoding schemes; a second determining module, configured to determine a target compression algorithm with the highest compression rate at a stream level; and a compression module, configured to: compress at least one target data stream corresponding to the target encoding scheme based on the target compression algorithm, and store compressed target data stream in a database.
- 10 [0008] According to a third aspect of one or more embodiments of this specification, an electronic device is provided, including: a processor; and a storage, configured to store processor-executable instructions. The processor runs the executable instruction, to implement the column data compression method according to any one of the above-mentioned aspects.
- [0009] According to a fourth aspect of one or more embodiments of this specification, a computer-readable storage medium is provided. The computer-readable storage medium stores computer instructions, and when the instructions are executed by a processor, steps of the column data compression method according to any one of the above-mentioned aspects are implemented.
- 15 [0010] The technical solutions provided in the embodiments of this specification can include the following beneficial effects.
- [0011] In the present disclosure, column data in a database table can be separately converted into a data stream in a plurality of encoding schemes at a column level, and then, at least one target data stream corresponding to a target encoding scheme in which smallest storage space is occupied is compressed based on a target compression algorithm with the highest compression rate. In the present disclosure, the column data can be encoded and compressed at at least two levels: the column level and the stream level, to improve a compression rate of the column data, and save more storage space.
- 20 25

BRIEF DESCRIPTION OF DRAWINGS

[0012] FIG. 1 is a flowchart of a column data compression method according to an example embodiment;

[0013] FIG. 2 is a block diagram of a column data compression apparatus according to an example embodiment; and

[0014] FIG. 3 is a schematic structural diagram of an electronic device according to an example embodiment.

5

DESCRIPTION OF EMBODIMENTS

[0015] Example embodiments are described in detail here, and examples of the example embodiments are presented in the accompanying drawings. When the following description relates to the accompanying drawings, unless specified otherwise, same numbers in different accompanying drawings represent same or similar elements. Implementations described in the following example embodiments do not represent all implementations consistent with one or more
10 embodiments of this specification. On the contrary, the implementations are merely examples of apparatuses and methods that are described in the appended claims in detail and consistent with some aspects of one or more embodiments of this specification.

[0016] It is worthwhile to note that, in some other embodiments, the steps of the corresponding
15 method are not necessarily performed in the order shown and described in this specification. In some other embodiments, the method can include more or fewer steps than those described in this specification. In addition, a single step described in this specification may be decomposed into a plurality of steps in some other embodiments for description; and a plurality of steps described in this specification may be combined into a single step for description in some other embodiments.

20 [0017] FIG. 1 is a flowchart of a column data compression method according to an example embodiment. As shown in FIG. 1, the method can be performed by an electronic device. Specifically, the electronic device can be a server in which a database is located. The present disclosure sets no limitation thereto. The method includes the following steps.

[0018] Step 101: Separately encode column data in a plurality of encoding schemes at a
25 column level, to obtain at least one data stream corresponding to each encoding scheme.

[0019] In the present disclosure, an encoding scheme at a column level corresponding to the column data can be determined based on a feature, for example, a storage type, of the column data, so that data in the same column of a database table is encoded based on different encoding schemes at the column level, to obtain at least one data stream corresponding to the encoding scheme.

- [0020] In an example, the storage type of the column data includes but is not limited to any one of the following: integer data, a fixed-length string, and a variable-length string.
- [0021] The integer data is data that has a fixed length and that is of an integer type (for example, a uint type). The integer type can further include uint8_t, uint16_t, uint32_t, and uint64_t based on a quantity of occupied bits.
- [0022] For example, uint8_t is an integer type occupying 8 bits, uint16_t is an integer type occupying 16 bits, and so on.
- [0023] The fixed-length string is a string with an invariable length. For example, the length is 4 bytes.
- 10 [0024] The variable-length string is a string with a variable length. For example, a length of one string is 4 bytes, and a length of another string is 2 bytes.
- [0025] In an example, the storage type of the column data is integer data, and the column can be referred to as an integer column (IntegerColumn).
- [0026] For example, the integer column can be encoded in the integer column encoding scheme, to obtain a first integer stream. The first integer stream can include each integer element
- 15 in a current column of the database table.
- [0027] The integer stream is a stream that continuously stores fixed-length integer data, namely, an integer array.
- [0028] Optionally, at least one candidate integer type can be determined based on a maximum
- 20 value of the integer element in the current column. A quantity of bits occupied by the candidate integer type is greater than or equal to a quantity of bits occupied by the maximum value. A candidate integer type occupying the smallest quantity of bits is determined as an integer corresponding to the integer column encoding scheme.
- [0029] For example, when an integer type corresponding to the current column is uint16_t, but
- 25 the maximum value of the integer element occupies only 7 bits, the candidate integer type can be uint8_t, uint16_t, uint32_t, and uint64_t. If uint8_t occupies the smallest quantity of bits in the candidate integer type, the integer type corresponding to the integer column encoding scheme is uint8_t.
- [0030] For example, the integer column can be encoded in a dictionary column encoding
- 30 scheme, to obtain a second integer stream and a first index stream. The second integer stream can include each integer element in the current column of the database table.

- [0031] The second integer stream can include integer elements that are different from each other in the current column. In other words, the second integer stream includes an array including a unique value of the current column.
- [0032] For example, the current column includes the following integer elements in an ascending order of row numbers: an integer element #1, the integer element #1, an integer element #5, an integer element #3, an integer element #4, the integer element #5, ..., and the second integer stream can include the integer element #1, the integer element #5, the integer element #3, and the integer element #4.
- [0033] It can be understood that the same integer column encoding scheme can be used for the second integer stream and the first integer stream, and the integer type corresponding to the integer column encoding scheme is the candidate integer type occupying the smallest quantity of bits.
- [0034] The first index stream can be a ref array, and is used to record a row number of each row of data in the second integer stream. That is, the first index stream can be used to query the second integer stream.
- [0035] For example, the first index stream includes: {0, 1}, {2, 5}, {3}, {4},
- [0036] In the present disclosure, data in the current column can be encoded in the dictionary column encoding scheme when a quantity of unique values included in the current column is less than or equal to a preset threshold.
- [0037] In an example, the storage type of the column data is a string, and each string has an invariable length, that is, is a fixed-length string. The column can be referred to as a string column (StringColumn).
- [0038] For example, the string column can be encoded in the string column encoding scheme, to obtain a first string stream. The first string stream can include each string in the current column of the database table.
- [0039] The string stream is a string array for continuously storage, and includes only specific content of a string, but does not include string length information.
- [0040] In this embodiment of the present disclosure, because each string has an invariable length, only the first string stream can be obtained after the string column is encoded in the string column encoding scheme.
- [0041] For example, the string column can be encoded in the dictionary column encoding scheme, to obtain a second string stream and a second index stream.

- [0042] The second string stream can include strings that are different from each other in the current column of the database table. In other words, the second string stream includes the unique value of the current column.
- [0043] For example, the current column includes the following strings in ascending order of row numbers: a string #1, a string #3, a string #2, the string #2, a string #4, a string #5, ..., and the second string stream can include the string #1, the string #3, the string #2, the string #4, the string #5,
- [0044] It can be understood that the second string stream can be encoded in the same string column encoding scheme as the first string stream.
- 10 [0045] The second index stream can be a ref array, and is used to record a row number of each row of data in the second string stream. That is, the second index stream can be used to query the second string stream.
- [0046] For example, the second index stream includes: {0}, {1}, {2, 3}, {4}, {5},
- [0047] In an example, the storage type of the column data is a string, and each string has a variable length, that is, is a variable-length string. The column can also be referred to as a string column (StringColumn).
- 15 [0048] For example, the string column can be encoded in a string column encoding scheme, to obtain a third string stream and a third integer stream. The third string stream can include each string in the current column of the database table. The third integer stream is used to determine a length of each string in the third string stream.
- 20 [0049] The string stream is a string array for continuously storage, and includes only specific content of a string, but does not include string length information.
- [0050] In this embodiment of the present disclosure, because each string has a variable length, after the string column is encoded in the string column encoding scheme, the third string stream and the third integer stream are obtained.
- 25 [0051] Optionally, the third integer stream can store an offset of an end position of each row of strings in the third string stream, and a difference between two adjacent offsets is a length of a current row of strings.
- [0052] For example, the string column can be encoded in the dictionary column encoding scheme, to obtain a fourth string stream, a fourth integer stream, and a third index stream. The fourth string stream includes strings that are different from each other in the current column, the
- 30

fourth integer stream is used to determine a length of each string in the fourth string stream, and the third index stream is used to query the fourth string stream. The third index stream can also be a ref array.

[0053] Because the string length has a variable length, after encoding in the dictionary column encoding scheme, three streams are obtained, and are a fourth string stream used to record unique values (strings that are different from each other) in the current column, a fourth integer stream used to record a length of each string in the fourth string stream, and a third index stream used to query the fourth string stream.

[0054] The above-mentioned descriptions are merely example descriptions. An encoding scheme of the column data at the column level is not limited in the present disclosure.

[0055] Step 102: Determine a target encoding scheme in which smallest storage space is occupied in the plurality of encoding schemes.

[0056] In the present disclosure, storage space occupied by original stream data can be determined based on the original stream data. Storage space occupied by the integer stream can be determined based on a size of integer array space, and storage space occupied by a string stream can be determined based on a total length of a string. For the sum of storage space occupied by at least one data stream corresponding to the same encoding scheme, a size of storage space occupied by the data stream corresponding to the encoding scheme is obtained.

[0057] For example, the storage type corresponding to the column data is integer data, and the first integer stream is obtained in the integer column encoding scheme. The second integer stream and the first index stream are obtained in the dictionary column encoding scheme. A size of storage space occupied by the first integer stream is a first byte quantity, and a size of storage space occupied by the second integer stream and the first index stream is a second byte quantity. If the first byte quantity is less than the second byte quantity, the target encoding scheme is the integer column encoding scheme; or if the first byte quantity is greater than the second byte quantity, the target encoding scheme is the dictionary column encoding scheme.

[0058] For another example, the storage type corresponding to the column data is a string, and each string has an invariable length. The first string stream is obtained in the string column encoding scheme. The second string stream and the second index stream are obtained in the dictionary column encoding scheme. A size of storage space occupied by the first string stream is a third byte quantity, and a size of storage space occupied by the second string stream and the

second index stream is a fourth byte quantity. If the third byte quantity is less than the fourth byte quantity, the target encoding scheme is the string column encoding scheme; or if the third byte quantity is greater than the fourth byte quantity, the target encoding scheme is the dictionary column encoding scheme.

5 **[0059]** For another example, the storage type corresponding to the column data is a string, and each string has a variable length. The third string stream and the third integer stream are obtained in the string column encoding scheme. The fourth character stream, the fourth integer stream, and the third index stream are obtained in the dictionary column encoding scheme. A size of storage space occupied by the third character stream and the third integer stream is a fifth byte quantity,
10 and a size of storage space occupied by the fourth character stream, the fourth integer stream, and the third index stream is a sixth byte quantity. If the fifth byte quantity is less than the sixth byte quantity, the target encoding scheme is the string column encoding scheme; or if the fifth byte quantity is greater than the sixth byte quantity, the target encoding scheme is the dictionary column encoding scheme.

15 **[0060]** Step 103: Determine a target compression algorithm with the highest compression rate at a stream level.

[0061] In the present disclosure, a data stream obtained through column level encoding can be an integer stream or a string stream.

[0062] Candidate compression algorithm sets respectively corresponding to the integer stream
20 and the string stream can be preset on the electronic device.

[0063] In an example, a candidate compression algorithm set corresponding to the integer stream can include but is not limited to at least one of the following: a Pfor (Pfor) compression algorithm, a run length encoding (RLE) compression algorithm, a Huffman (ZigZag) compression algorithm, a delta compression algorithm, or a double delta compression algorithm.

25 **[0064]** In an example, a candidate compression algorithm set corresponding to the string stream can include but is not limited to at least one of the following: a prefix compression algorithm or a general compression algorithm.

[0065] The general compression algorithm includes but is not limited to a zip compression algorithm, an LZW compression algorithm, an LZ77 compression algorithm, etc.

30 **[0066]** In the present disclosure, a corresponding candidate compression algorithm can be selected based on a feature of the data stream, at least a part of data included in at least one target

data stream is separately compressed based on each candidate compression algorithm, to obtain compressed data, and then a candidate compression algorithm with the smallest quantity of bits occupied by the compressed data is determined as the target compression algorithm.

[0067] For example, all data included in the target data stream can be separately compressed based on different candidate compression algorithms, and then a candidate compression algorithm in which compressed data occupies the smallest quantity of bits is determined as the target compression algorithm.

[0068] It is considered that it takes a large amount of time to compress all data included in the target data stream. In the present disclosure, the data included in the at least one target data stream can be sampled based on a preset sampling rate, to obtain sampled data. Further, the sampled data can be separately compressed based on each candidate compression algorithm, to obtain compressed data. Further, the candidate compression algorithm in which compressed data occupies the smallest quantity of bits is determined as the target compression algorithm. Therefore, a delay of determining the target compression algorithm is reduced.

[0069] The sampling rate can be any value in an interval (0, 1). In other words, the sampling rate can be any value greater than 0 and less than 1. For example, the sampling rate can be 0.25 (25%).

[0070] Step 104: Compress at least one target data stream corresponding to the target encoding scheme based on the target compression algorithm, and store compressed target data stream in a database.

[0071] In the present disclosure, a data stream corresponding to the target encoding scheme is a target data stream, and there can be one or more target data streams.

[0072] For example, the target encoding scheme is an integer column encoding scheme, the target data stream is a first integer stream, and there is one target data stream.

[0073] For another example, the target encoding scheme is a dictionary column encoding scheme, the target data stream includes a second integer stream and a first index stream scheme, and there are two target data streams.

[0074] For another example, the target encoding scheme is the string column encoding scheme, the target data stream includes a first string stream, and there is one target data stream.

[0075] For another example, the target encoding scheme is the dictionary column encoding scheme, the target data stream includes a second string stream and a second index stream, and

there are two target data streams.

[0076] For another example, the target encoding scheme is the string column encoding scheme, the target data stream includes a third string stream and a third integer stream, and there are two target data streams.

5 **[0077]** For another example, the target encoding scheme is the dictionary column encoding scheme, the target data stream includes a fourth character series stream, a fourth integer stream, and a third index stream, and there are three target data streams.

[0078] In the present disclosure, after the target compression algorithm is determined, all data of the target data stream can be compressed based on the target compression algorithm, and the
10 compressed target data stream is stored in the database.

[0079] For example, the compressed target data stream can be stored in a storage medium of the database.

[0080] For example, the compressed target data stream can alternatively be stored in other hardware or software, or an associated storage medium of the database. The present disclosure sets
15 no limitation thereto.

[0081] When the database table needs to be queried, compressed target data stream corresponding to a field that needs to be queried, namely, a field to which the column data belongs, can be found from the storage medium of the database based on the field. Further, target data streams can be obtained through decompression based on the target compression algorithm.
20 Further, a required integer element or string is found from the target data stream. A search result is returned.

[0082] In the above-mentioned embodiment, the column data is effectively encoded and compressed by fully considering a feature of data at at least two layers (the column level and the stream level) in a hierarchical encoding scheme, to improve a compression rate, save more storage
25 space, and improve availability.

[0083] In some optional embodiments, to better process an online analytical processing (OLAP) service in the database, data in the same column in the database table is continuously stored together in a column storage manner, and data in the same column can be better encoded and compressed due to the same data type.

30 **[0084]** To save storage space, relatively high data encoding/decoding performance needs to be ensured when column storage is performed. Therefore, the present disclosure provides the above-

mentioned column data compression method.

[0085] First, the column data is encoded in different encoding schemes at the column level, to obtain at least one data stream corresponding to each encoding scheme. The candidate compression algorithm with the highest compression rate is selected from the candidate compression algorithm based on a feature of each data stream, to perform compression. A process of encoding the column data to obtain the data stream can be referred to as column level encoding, and a process of compressing the data stream can be referred to as stream level encoding.

[0086] In comparison with a case in which the column data is directly compressed based on a common algorithm, in the present disclosure, data is encoded and compressed at at least two levels by repeatedly considering the feature of the data, to implement a higher compression rate.

[0087] In addition, in the present disclosure, a data length obtained based on each candidate compression algorithm can be precisely calculated, so that the target compression algorithm with the highest compression rate can be selected to compress the stream data, the compression rate is also improved, and a size of occupied storage space can be reduced. In this way, availability is high.

[0088] As shown in FIG. 2, a column data compression apparatus can be applied to an electronic device, for example, a server in which a database is located, to implement the technical solutions of this specification. The column data compression apparatus can include: an encoding module 201, configured to separately encode a column data in a plurality of encoding schemes at a column level, to obtain at least one data stream corresponding to each encoding scheme; a first determining module 202, configured to determine a target encoding scheme in which smallest storage space is occupied in the plurality of encoding schemes; a second determining module 203, configured to determine a target compression algorithm with the highest compression rate at a stream level; and a compression module 204, configured to: compress at least one target data stream corresponding to the target encoding scheme based on the target compression algorithm, and store compressed target data stream in a database.

[0089] The system, apparatus, module, or unit illustrated in the above-mentioned embodiments can be implemented by using a computer chip or an entity, or can be implemented by using a product having a certain function. A typical implementation device is a computer, and a specific form of the computer may be a personal computer, a laptop computer, a cellular phone, a camera phone, a smartphone, a personal digital assistant, a media player, a navigation device, an

e-mail transceiver device, a game console, a tablet computer, a wearable device, or any combination of several devices in these devices.

[0090] In a typical configuration, the computer includes one or more processors (CPU), one or more input/output interfaces, one or more network interfaces, and one or more memories.

5 [0091] The memory may include at least one of a non-persistent memory, a random access memory (RAM), and a nonvolatile memory in a computer-readable medium, for example, a read-only memory (ROM) or a flash read-only memory (flash RAM). The memory is an example of the computer-readable medium.

[0092] The computer-readable medium includes a persistent and a non-persistent, removable
10 and non-removable media, which implement information storage by using any method or technology. The information may be a computer-readable instruction, a data structure, a program module, or other data. Examples of a computer storage medium include but are not limited to a phase change random access memory (PRAM), a static RAM (SRAM), a dynamic RAM (DRAM), a RAM of another type, a read-only memory (ROM), an electrically erasable programmable ROM
15 (EEPROM), a flash memory or another memory technology, a compact disc ROM (CD-ROM), a digital versatile disc (DVD) or another optical storage, a cassette tape, a magnetic disk storage, a quantum memory, a storage medium based on graphene, another magnetic storage device, or any other non-transmission medium. The computer storage medium can be configured to store information that can be accessed by the computing device. Based on the definition in this
20 specification, the computer-readable medium does not include transitory media such as a modulated data signal and carrier.

[0093] FIG. 3 is a schematic structural diagram of an electronic device according to an example embodiment. As shown in FIG. 3, at a hardware level, the device includes a processor 302, an internal bus 304, a network interface 306, a memory 308, and a nonvolatile memory 310,
25 and certainly may further include other hardware needed by a service. One or more embodiments of this specification can be implemented based on software. For example, the processor 302 reads a corresponding computer program from the nonvolatile memory 310 to the memory 308 and then runs the computer program. Certainly, in addition to software implementations, one or more embodiments of this specification do not preclude other implementations, such as a logic device
30 or a combination of software and hardware. In other words, an execution body of the following processing procedure is not limited to each logical unit, and can be hardware or a logic device.

[0094] It is worthwhile to further note that, the terms "include", "comprise", or any other variant thereof are intended to cover a non-exclusive inclusion, so that a process, a method, a product, or a device that includes a list of elements not only includes those elements but also includes other elements which are not expressly listed, or further includes elements inherent to such process, method, product, or device. Without more constraints, an element preceded by "includes a ..." does not preclude the existence of additional identical elements in the process, method, product, or device that includes the element.

[0095] Specific embodiments of this specification are described above. Other embodiments fall within the scope of the appended claims. In some cases, the actions or steps described in the claims can be performed in an order different from that in the embodiments, and the desired results can still be achieved. In addition, the process depicted in the accompanying drawings does not necessarily need a particular order or consecutive order to achieve the desired results. In some implementations, multi-tasking and parallel processing are feasible or may be advantageous.

[0096] Terms used in one or more embodiments of this specification are merely used to describe specific embodiments, and are not intended to limit the one or more embodiments of this specification. The terms "a" and "the" of singular forms used in one or more embodiments of this specification and the appended claims are also intended to include plural forms, unless otherwise specified in the context clearly. It should be further understood that the term "and/or" used in this specification indicates and includes any or all possible combinations of one or more associated listed items.

[0097] It should be understood that although terms "first", "second", "third", etc. may be used in one or more embodiments of this specification to describe various types of information, the information should not be limited to these terms. These terms are used merely to differentiate information of the same type. For example, without departing from the scope of one or more embodiments of this specification, first information can also be referred to as second information, and similarly, the second information can also be referred to as the first information. Depending on the context, for example, the word "if" used here can be explained as "while", "when", or "in response to determining".

[0098] The above-mentioned descriptions are merely example embodiments of one or more embodiments of this specification, but are not intended to limit the one or more embodiments of this specification. Any modification, equivalent replacement, improvement, etc. made without

departing from the spirit and principle of the one or more embodiments of this specification shall fall within the protection scope of the one or more embodiments of this specification.

CLAIMS

What is claimed is:

1. A column data compression method, comprising:

separately encoding a column data in a plurality of encoding schemes at a column level, to

5 obtain at least one data stream corresponding to each encoding scheme;

determining a target encoding scheme in which smallest storage space is occupied in the plurality of encoding schemes;

determining a target compression algorithm with the highest compression rate at a stream level; and

10 compressing at least one target data stream corresponding to the target encoding scheme based on the target compression algorithm, and storing compressed target data stream in a database.

2. The method according to claim 1, wherein a storage type of the column data is integer data, and the separately encoding the column data in a plurality of encoding schemes at a column level, to obtain at least one data stream corresponding to each encoding scheme comprises:

15 encoding the column data in an integer column encoding scheme, to obtain a first integer stream, wherein the first integer stream comprises each integer element in a current column; and

encoding the column data in a dictionary column encoding scheme, to obtain a second integer stream and a first index stream, wherein the second integer stream comprises integer elements that are different from each other in a current column, and the first index stream is used to query the
20 second integer stream.

3. The method according to claim 2, wherein the method further comprises:

determining at least one candidate integer type based on a maximum value of the integer element in the current column, wherein storage space occupied by the candidate integer type is greater than or equal to storage space occupied by the maximum value; and

25 determining that a candidate integer type occupying smallest storage space is an integer type corresponding to the integer column encoding scheme.

4. The method according to claim 1, wherein a storage type of the column data is a string, each string has an invariable length, and the separately encoding the column data in a plurality of encoding schemes at a column level, to obtain at least one data stream corresponding to each

30 encoding scheme comprises:

encoding the column data in a string column encoding scheme, to obtain a first string stream, wherein the first string stream comprises each string in a current column; and

encoding the column data in a dictionary column encoding scheme, to obtain a second string stream and a second index stream, wherein the second string stream comprises strings that are
5 different from each other in the current column, and the second index stream is used to query the second string stream.

5. The method according to claim 1, wherein a storage type of the column data is a string, each string has a variable length, and the separately encoding the column data in a plurality of encoding schemes at a column level, to obtain at least one data stream corresponding to each
10 encoding scheme comprises:

encoding the column data in a string column encoding scheme, to obtain a third string stream and a third integer stream, wherein the third string stream comprises each string in a current column, and the third integer stream is used to determine a length of each string in the third string stream;
and

15 encoding the column data in a dictionary column encoding scheme, to obtain a fourth string stream, a fourth integer stream, and a third index stream, wherein the fourth string stream comprises strings that are different from each other in the current column, the fourth integer stream is used to determine a length of each string in the fourth string stream, and the third index stream is used to query the fourth string stream.

20 6. The method according to any one of claims 1 to 5, wherein the determining a target compression algorithm with a highest compression rate at a stream level comprises:

separately compressing, based on each candidate compression algorithm, at least a part of data comprised in the at least one target data stream, to obtain compressed data; and

determining that a candidate compression algorithm in which compressed data occupies
25 smallest storage space is the target compression algorithm.

7. The method according to claim 6, wherein the method further comprises:

sampling, based on a preset sampling rate, the data comprised in the at least one target data stream, to obtain sampled data; and

the separately compressing, based on each candidate compression algorithm, at least a part of
30 data comprised in the at least one target data stream, to obtain compressed data comprises:

separately compressing the sampled data based on each candidate compression algorithm, to

obtain the compressed data.

8. A column data compression apparatus, comprising:

an encoding module, configured to separately encode a column data in a plurality of encoding schemes at a column level, to obtain at least one data stream corresponding to each encoding
5 scheme;

a first determining module, configured to determine a target encoding scheme in which smallest storage space is occupied in the plurality of encoding schemes;

a second determining module, configured to determine a target compression algorithm with a highest compression rate at a stream level; and

10 a compression module, configured to: compress at least one target data stream corresponding to the target encoding scheme based on the target compression algorithm, and store compressed target data stream in a database.

9. An electronic device, comprising:

a processor; and

15 a storage, configured to store instructions that can be executed by the processor, wherein the processor runs the executable instructions, to implement the column data compression method according to any one of claims 1 to 7.

10. A computer-readable storage medium, wherein the computer-readable storage medium stores computer instructions, and when the instructions are executed by a processor, the steps of
20 the column data compression method according to any one of claims 1 to 7 are implemented.