

METHOD AND DEVICE FOR TRAINING LANGUAGE MODEL, INFORMATION PROCESSING METHOD AND DEVICE

[0001] The present application claims priority to Chinese Patent Application No. 202411191440.X, titled "METHOD AND DEVICE TRAINING FOR LANGUAGE MODEL, INFORMATION PROCESSING METHOD AND DEVICE", filed on August 28, 2024 with the National Intellectual Property Administration, PRC, which is incorporated herein by reference in its entirety.

FIELD

[0002] Embodiments of the present disclosure relate to the technical field of machine learning, and in particular to a method and device for training a language model, and an information processing method and device.

BACKGROUND

[0003] With the development of the computer technology and the Internet technology, the large language model is applied in more and more scenarios. In order to meet the diverse needs of users, the capability of the large language model is gradually enhanced. The current large language models deployed in the businesses are mainly obtained by pre-training based on large-scale document data. These models typically only use context information of document prefixes when performing document prediction tasks. However, with the expansion of model scale and the diversification of application scenarios, the large language model still faces many challenges when processing complex inference across documents, instruction execution, and reading and comprehension tasks. For example, the model may not be able to accurately follow instructions, have limited inference capability on conditional documents, and have high variance in context learning. In the conventional art, to avoid this problem, the standard pre-training method or the retrieval-based pre-training method is usually used to improve the comprehension capability of the model. However, these two methods cannot guarantee the coherence of the context during the pre-training of the model, and there may be data duplication, which leads to poor performance of the model on complex tasks and

over-fitting problem. Therefore, an effective solution is urgently needed to solve the above problem.

SUMMARY

5 **[0004]** In view of this, a method for training a language model is provided according to an embodiment of the present disclosure. An information processing method, a device for training a language model, an information processing device, a computing device, a computer-readable storage medium and a computer program product are further provided according to one or more embodiments of the present disclosure to solve the technical
10 deficiencies existing in the conventional art.

[0005] In a first aspect of the embodiments according to the present disclosure, a method for training a language model is provided, which includes:

acquiring a document set of an initial language model, and selecting associated documents in the document set to form an associated document set;

15 constructing a document relationship diagram corresponding to the associated document set based on the associated documents in the associated document set and similarity between the associated documents;

determining a document splicing path in the document relationship diagram, and extracting target associated documents from the associated document set based on the
20 document splicing path for splicing processing, to obtain target context information; and

training the initial language model based on the target context information until obtaining a target language model that meets a training stopping condition.

[0006] Preferably, the selecting associated documents in the document set to form an associated document set includes:

25 constructing, by a text retrieval model, document embedding vectors corresponding to respective initial documents in the document set;

calculating cosine similarity between the document embedding vectors corresponding to the respective initial documents; and

selecting the associated documents in the document set based on a cosine similarity

calculation result to form the associated document set.

[0007] Preferably, after the constructing, by a text retrieval model, document embedding vectors corresponding to respective initial documents in the document set, the method further includes:

5 quantizing, by a preset quantizer, the document embedding vectors corresponding to the respective initial documents, and constructing a document index based on the quantized document embedding vectors; and

 searching the document set according to a preset search strategy and the document index, and determining the associated documents based on a search result to form the
10 associated document set.

[0008] Preferably, the constructing a document relationship diagram corresponding to the associated document set based on the associated documents in the associated document set and similarity between the associated documents includes:

 constructing graph nodes based on the associated documents in the associated document
15 set, and constructing graph edges based on the similarity between the associated documents in the associated document set; and

 establishing a connectivity relationship between the graph nodes based on the graph edges, and generating the document relationship diagram corresponding to the associated document set based on a connectivity relationship establishment result.

20 **[0009]** Preferably, the determining a document splicing path in the document relationship diagram includes:

 determining an i-th graph node in the document relationship diagram and constructing an initial document path corresponding to the i-th graph node, where i is a positive integer and starts from 1, and a maximum value of i is the number of graph nodes in the document
25 relationship diagram;

 selecting a j-th graph node from neighboring graph nodes connected to the i-th graph node, and updating the initial document path to an intermediate document path based on the j-th graph node, where j is a positive integer and starts from 1, and a maximum value of j is the number of the neighboring graph nodes; and

determining whether there are any unvisited neighboring nodes in the neighboring graph nodes connected to the i-th graph node, where

if there are any unvisited neighboring nodes in the neighboring graph nodes connected to the i-th graph node, j is increased by 1 and the step of selecting a j-th graph node from neighboring graph nodes connected to the i-th graph node is performed; and

if there are no unvisited neighboring nodes in the neighboring graph nodes connected to the i-th graph node, i is increased by 1 and the step of determining an i-th graph node in the document relationship diagram is performed until i equals the number of the graph nodes, multiple intermediate document paths are obtained, and an intermediate document path that meets a document splicing condition is extracted from the multiple intermediate document paths as the document splicing path.

[0010] Preferably, the extracting target associated documents from the associated document set based on the document splicing path for splicing processing, to obtain target context information includes:

determining splicing graph nodes in the document splicing path, and extracting the target associated documents from the associated document set based on the splicing graph nodes; and

splicing the target associated documents based on path direction information of the document splicing path to obtain target context information that matches an input condition of the initial language model.

[0011] Preferably, the training the initial language model based on the target context information until obtaining a target language model that meets a training stopping condition includes:

inputting the target context information into the initial language model for processing, to obtain predication text information;

optimizing the initial language model based on the predication text information and reference text information corresponding to the target context information;

determining whether the optimized initial language model meets the training stopping condition;

selecting, if the optimized initial language model does not meet the training stopping

condition, candidate associated documents from the document set to form a candidate associated document set, and using the candidate associated document set as the associated document set, and performing the step of constructing a document relationship diagram corresponding to the associated document set based on the associated documents in the associated document set and similarity between the associated documents; and

using the optimized initial language model as the target language model if the optimized initial language model meets the training stopping condition.

[0012] In a second aspect of the embodiments according to the present disclosure, an information processing method is provided, which includes:

acquiring question information inputted by a client, and searching for multiple answer documents in a business knowledge database based on the question information;

splicing the multiple answer documents and inputting a splicing result and the question information into a target language model for processing to obtain answer information, where the target language model is trained by the above method; and

feeding back the answer information to the client.

[0013] In a third aspect of the embodiments according to the present disclosure, a device for training a language model is provided, which includes: an acquisition module, a construction module, a splicing module and a training module. The acquisition module is configured to acquire a document set of an initial language model and select associated documents in the document set to form an associated document set. The construction module is configured to construct a document relationship diagram corresponding to the associated document set based on the associated documents in the associated document set and similarity between the associated documents. The splicing module is configured to determine a document splicing path in the document relationship diagram, and extract target associated documents from the associated document set based on the document splicing path for splicing processing to obtain target context information. The training module is configured to train the initial language model based on the target context information until obtaining a target language model that meets a training stopping condition.

[0014] In a fourth aspect of the embodiments according to the present disclosure, an information processing device is provided, which includes: an information acquisition module,

a model prediction module and an information feedback module. The information acquisition module is configured to acquire question information inputted by a client, and searching for multiple answer documents in a business knowledge database based on the question information. The model prediction module is configured to splice the multiple answer documents and inputting a splicing result and the question information into a target language model for processing to obtain answer information, where the target language model is trained by the above method. The information feedback module is configured to feed back the answer information to the client.

[0015] In a fifth aspect of the embodiments according to the present disclosure, a computing device is provided, which includes a memory and a processor. The memory is configured to store computer executable instructions, the processor is configured to execute the computer executable instructions, where the computer executable instructions, when executed by the processor, perform the steps of the document processing method described above.

[0016] In a sixth aspect of the embodiments according to the present disclosure, a computer-readable storage medium is provided, which includes computer executable instructions stored thereon, where the computer executable instructions, when executed by a processor, perform the steps of the method for training a language model or the document processing method described above.

[0017] In a seventh aspect of the embodiments according to the present disclosure, a computer program product is provided, which includes computer programs or instructions, where the computer programs or instructions, when executed by a processor, perform the steps of the method for training a language model or the document processing method described above.

[0018] With the method for training a language model according to the embodiment, in order to improve the performance of the language model in processing complex contexts and inference across documents, after the document set of the initial language model is acquired, associated documents with similarity are selected from the document set of the initial language model to form an associated document set. Afterwards, in order to enhance the comprehension capability of the model to text across documents, a document relationship diagram corresponding to the associated document set may be constructed based on the

associated documents in the associated document set and the similarity between the associated documents, where the similarity relationship between the associated documents may be reflected through the document relationship diagram. On this basis, the document splicing path may be determined in the document relationship diagram, and the target associated documents may be extracted from the associated document set based on the document splicing path for splicing processing to obtain the target context information. Finally, the initial language model may be trained based on the target context information until the target language model that meets the training stopping condition is obtained. The model can be pre-trained based on a sequence of associated documents, so that the language model can read and infer across documents, thereby effectively improving the performance of the model in complex context tasks.

BRIEF DESCRIPTION OF THE DRAWINGS

[0019] Figure 1 is a schematic diagram of a method for training a language model according to an embodiment of the present disclosure;

[0020] Figure 2 is a flowchart of a method for training a language model according to an embodiment of the present disclosure;

[0021] Figure 3 is a flowchart of an information processing method according to an embodiment of the present disclosure;

[0022] Figure 4 is a flowchart of a processing procedure of an information processing method according to an embodiment of the present disclosure;

[0023] Figure 5 is a schematic diagram of a structure of a device for training a language model according to an embodiment of the present disclosure;

[0024] Figure 6 is a schematic diagram of a structure of an information processing device according to an embodiment of the present disclosure; and

[0025] Figure 7 is a block diagram of a structure of a computing device according to an embodiment of the present disclosure.

DETAILED DESCRIPTION OF EMBODIMENTS

[0026] Many specific details are set forth in the following description to facilitate a full understanding of the present disclosure. However, the present disclosure may be implemented in many other ways different from those described herein, and those skilled in the art may make similar generalizations without violating the connotations of the present disclosure, and thus the present disclosure is not limited by the specific embodiments described below.

[0027] Terms used in one or more embodiments of the present disclosure are used only for the purpose of describing a particular embodiment and are not intended to limit one or more embodiments of the present disclosure. The singular forms of "a", "the", and "said" used in one or more embodiments of the present disclosure and the claims are also intended to encompass the plural form, unless the context clearly indicates otherwise. It should also be understood that the term "and/or" used in one or more embodiments of the present disclosure refers to and encompasses any or all possible combinations of one or more of the listed items in association.

[0028] It should be understood that while the terms "first", "second", etc. may be used in one or more embodiments of the present disclosure to describe various information, such information should not be limited to these terms. These terms are used only to distinguish the same type of information from one another. For example, without departing from the scope of the one or more embodiments of the present disclosure, "a first..." may also be referred to as "a second...", and similarly, "a second..." may be referred to as "a first...". Depending on the context, the phrase "if" as used herein may be interpreted as "at the time of" or "when" or "in response to a determination".

[0029] In addition, it should be noted that the user information (including but not limited to user device information, user personal information, etc.) and data (including but not limited to data used for analysis, stored data, displayed data, etc.) involved in one or more embodiments of the present disclosure are authorized by the user or sufficiently authorized by the parties, and that the set, use, and processing of the relevant data are subject to relevant laws, regulations, and regulations of relevant countries and regions, and a corresponding operation portal is provided for the user to choose to authorize or reject.

[0030] First, the terminologies involved in one or more embodiments of the present disclosure are explained below.

[0031] Large Language Model (LLM) refers to a deep learning model trained by using a

large amount of text data, the core idea of which is to learn the patterns and structures of natural language through large-scale unsupervised training, simulating to a certain extent the human language cognition and generation process.

[0032] Greedy algorithm is an algorithm that takes the best or optimal (i.e. most advantageous) selection in the current state at each step of the selection process, to obtain a global best or optimal result. Every selection made by the greedy algorithm is a local optimal selection in some sense.

[0033] In the present disclosure, a method for training a language model is provided. An information processing method, a device for training a language model, an information processing device, a computing device, a computer-readable storage medium, and a computer program product are further provided according to one or more embodiments of the present disclosure, which will be described in detail in the following embodiments.

[0034] In practice, in order to provide better prediction performance by the language model, the standard pre-training method and the retrieval-based pre-training (kNN pre-training) method are usually used for pre-training. In the standard pre-training method, documents are randomly shuffled and spliced into a context window of a fixed length for language model training. Although this method is simple, it cannot guarantee the coherence of the context, resulting in poor performance of the model when processing complex tasks. In the retrieval-based pre-training method, each document and retrieved associated documents are placed in the same context for training. Although this method can improve the semantic comprehension capability of the model to a certain extent, there is a problem of data duplication, which may lead to over-fitting of the model. Therefore, an effective solution is urgently needed to solve the above problem.

[0035] Referring to the schematic diagram shown in Figure 1, with the method for training a language model according to the embodiment, in order to improve the performance of the language model in processing complex contexts and inferring across documents, after a document set of an initial language model is acquired, associated documents with similarity are selected from the document set of the initial language model to form an associated document set. Afterwards, in order to enhance the comprehension capability of the model to text across documents, a document relationship diagram corresponding to the associated document set may be constructed based on associated documents in the associated document

set and the similarity between the associated documents, where the similarity relationship between the associated documents may be reflected through the document relationship diagram. On this basis, a document splicing path may be determined in the document relationship diagram, and target associated documents may be extracted from the associated document set based on the document splicing path for splicing processing to obtain target context information. Finally, the initial language model may be trained based on the target context information until the target language model that meets the training stopping condition is obtained. The model can be pre-trained based on a sequence of associated documents, so that the language model can read and infer across documents, thereby effectively improving the performance of the model in complex context tasks.

[0036] Referring to Figure 2, Figure 2 shows a flowchart of a method for training a language model according to an embodiment of the present disclosure, which includes the following steps S202 to S208.

[0037] In step S202, a document set of an initial language model is acquired, and associated documents are selected in the document set to form an associated document set.

[0038] The method for training a language model according to the embodiment may be applied to a pre-training stage of any structured large language model, such as a Large Language Model Meta AI (LLaMA), which is not limited in the embodiment.

[0039] Specifically, the initial language model refers to a language model that needs to be pre-trained at the current stage, which may be deployed in a question and answer interaction scenario. In a case that a user inputs question information, the model may search for associated documents in the knowledge database and generate answer information that matches the question information based on the associated documents, thereby feeding back to the user answer information that meets the question and answer needs of the user. Correspondingly, the document set specifically refers to a set of documents required for the initial language model during pre-training. The documents contained in the set may come from different data sources and correspond to different business fields, such as fields of mathematical knowledge, film knowledge, and transaction knowledge, which is not limited in this embodiment. Correspondingly, the associated documents specifically refer documents with similarity selected from the document set, which are used to form an associated document set. It may be understood that the associated documents in the associated document

set all have similarity, which are used for the pre-training of the language model subsequently, so that the language model can obtain the capability of reading and inferring across documents.

[0040] Based on this, in order to improve the performance of the language model in processing complex contexts and inferring across documents, after the document set of the initial language model is acquired, associated documents with similarity are selected from the document set of the initial language model to form an associated document set. Afterwards, in order to enhance the comprehension capability of the model to text across documents, a document relationship diagram corresponding to the associated document set may be constructed based on the associated documents in the associated document set and the similarity between the associated documents, where the similarity relationship between the associated documents may be reflected through the document relationship diagram. On this basis, the document splicing path may be determined in the document relationship diagram, and the target associated documents may be extracted from the associated document set based on the document splicing path for splicing processing to obtain the target context information. Finally, the initial language model may be trained based on the target context information until the target language model that meets the training stopping condition is obtained.

[0041] Furthermore, after acquiring the document set of the initial language model, in order to select documents with high similarity to form an associated document set to improve the comprehension capability of the model across documents, the associated documents may be selected by calculating cosine similarity. In this embodiment, the implementation is specifically described as follows:

constructing, by a text retrieval model, document embedding vectors corresponding to respective initial documents in the document set; calculating cosine similarity between the document embedding vectors corresponding to the respective initial documents; and selecting the associated documents in the document set based on a cosine similarity calculation result to form the associated document set.

[0042] Specifically, the text retrieval model specifically refers to a model that may construct an embedding vector for each initial document, which may be a Contriever model or other models that may construct embedding vectors, and is not limited in this embodiment. Correspondingly, the document embedding vector specifically refers to a vector expression

obtained by performing embedding processing on the initial document using the text retrieval model. Correspondingly, the cosine similarity specifically refers to a value obtained by calculating a similarity of respective initial texts based on the document embedding vector.

[0043] Based on this, after determining the document set that used to pre-train the initial language model, in order to select similar and non-repeated documents for model pre-training, the associated document set may be constructed by calculating cosine similarity. During this process, document embedding vectors corresponding to respective initial documents in the document set may be constructed by a text retrieval model. On this basis, the cosine similarity between the document embedding vectors corresponding to the respective initial documents may be calculated. Afterwards, the cosine similarity calculation result may be compared with a preset similarity threshold, and documents with the cosine similarity calculation result greater than the similarity threshold may be selected as associated documents to form an associated document set. It can be ensured that the associated documents in the associated document set are all documents with higher similarity, thereby improving the context comprehension and inference capability of the model.

[0044] In specific implementation, when performing embedding processing on the documents in a document set by using the Contriever model, each document d_i may be first mapped to an embedding vector $E(d_i)$ through the Contriever model, and the similarity between documents may be calculated based on the cosine similarity, which may be achieved through the formula $s(d_i, d_j) = \cos(E(d_i), E(d_j))$, where d_i and d_j represent two documents that need to calculate the similarity therebetween.

[0045] In summary, similar documents are selected by calculating cosine similarity to form an associated document set, pre-training processing may be performed by using the similar documents in the model pre-training stage, so that the model can have strong prediction capability on the basis of text processing across documents.

[0046] In addition, in order to improve the efficiency of constructing the associated document set, approximate nearest neighbor search and reverse file indexing may be used to construct the associated document set. In this embodiment, the implementation is specifically described as follows:

quantizing, by a preset quantizer, the document embedding vectors corresponding to the respective initial documents, and constructing a document index based on the quantized

document embedding vectors; and searching the document set according to a preset search strategy and the document index, and determining the associated documents based on a search result to form the associated document set.

[0047] Specifically, the quantizer refers to a unit that quantizes the document embedding vector of each initial document. Correspondingly, the document indexing specifically refers to the construction of index information for searching for similar documents based on the quantified document embedding vector. Correspondingly, the search strategy specifically refers to a strategy used to search for k similar documents.

[0048] Based on this, in order to achieve efficient retrieval and quickly determine the documents with high similarity to form the associated document set, a preset quantizer may be used to first quantize the document embedding vector corresponding to each initial document, to construct a document index based on the quantized document embedding vectors. On this basis, the document set may be searched according to the preset search strategy and the document index, and then the associated documents with high similarity may be determined based on the search result to form the associated document set, which may be used for pre-training the initial language model subsequently.

[0049] In practice, when constructing the associated document set using approximate nearest neighbor search and inverted file index (FAISS), the FAISS index may be first constructed, and the document embedding vectors may be quantified and indexed using a product quantization method. Afterwards, similarity search is performed, during which approximate nearest neighbor search may be performed in the FAISS index to quickly find the top k most similar documents of each document, to form an associated document set for use.

[0050] In a specific implementation, firstly, product quantification of the document embedding vectors may be performed. Specifically, the PQ quantizer is configured in a way that the product quantizer is configured by using *faiss.ProductQuantizer* class from the FAISS database. It is required to specify the dimension of the quantization center (dsub) and the number of sub quantizers (M), so that the entire vector space can be divided into multiple subspaces, and each subspace can be quantized independently. The PQ quantizer is trained in a way that the PQ quantizer is trained by using a set of training data (which may be a part of the entire dataset). An optimal quantization center will be found for each subspace in the training process. The quantized document is embedded in a way that the embedding vector of

the document is quantized through a trained PQ quantizer to obtain the quantized encoding.

[0051] Secondly, the FAISS index is constructed. Specifically, an index type is selected, the Inverted File with Product Quantification (IVFPQ) index type is commonly used in FAISS for index based on product quantification. This index combines inverted indexing (IVF) and product quantification (PQ) to achieve efficient approximate nearest neighbor search. Index is created in a way that the index is constructed by using *faiss.IndexIVFPQ* class of FAISS. It is required to specify the quantizer (i.e. the PQ quantizer trained in first step), the number of cluster centers in the inverted list (nlist), and the number of nearest neighbors retained under each cluster center. Data is added to the index in a way that the quantized document embedding vector is added to the constructed FAISS index.

[0052] Finally, an approximate nearest neighbor search is performed. Specifically, a search parameter is set. Before searching, a parameter *nprobe* is set as needed, which determines the number of nearest neighbors to be explored under each cluster center during the search, thereby affecting the speed and accuracy of the search. Search is performed in a way that an approximate nearest neighbor search is performed on a given query vector by using a search function (such as *index.search*) of FAISS. The query vector may be any new document embedding vector, which also needs to be quantized by a PQ quantizer. A result is obtained in a way that the search function will return the indexes and distances of the top k most similar documents for each query vector. Thus, an associated document set is constructed for pre-training of the initial language model.

[0053] For example, a document set corresponding to the pre-trained language model, i.e. LLaMA model, is determined, which contains m documents. By performing embedding processing on each document in the document set using the Contriever model, the embedding vector E corresponding to each document is obtained. Then, the cosine similarity between the embedding vectors E corresponding to the documents is calculated, and a result is calculated based on the cosine similarity, n documents with similarity are selected from m documents to form an associated document set ($n < m$). For example, the n documents may be documents 1, 2,... n associated with image processing knowledge. The LLaMA model may be pre-trained subsequently, so that the model can have a better context comprehension and inference capability.

[0054] In summary, by constructing an index to search for similar documents for retrieval,

the efficiency of constructing associated documents can be improved, thereby saving time for model pre-training.

[0055] In step S204, a document relationship diagram corresponding to the associated document set is constructed based on the associated documents in the associated document set and the similarity between the associated documents.

[0056] Specifically, after determining the associated document set including the associated documents, in order to further select highly associated documents for model pre-training, a document relationship diagram may be constructed to reflect the strength of the association between documents. Then, highly associated documents may be selected for model pre-training based on the document relationship diagram. Therefore, based on the associated documents in the associated document set and the similarity between the associated documents, a document relationship diagram corresponding to the associated document set may be constructed for subsequent use. The document relationship diagram specifically refers to an undirected weighted graph constructed by using the associated documents as graph nodes and using the similarity between the associated documents as graph edges.

[0057] Furthermore, when constructing a document relationship diagram, it is required to fully utilize the documents and the similarity between the documents to complete the construction considering that the document relationship diagram will subsequently serve as a foundation for constructing context information inputted to the model. In this embodiment, the implementation is specifically described as follows:

constructing graph nodes based on the associated documents in the associated document set, and constructing graph edges based on the similarity between the associated documents in the associated document set; and establishing a connectivity relationship between the graph nodes based on the graph edges, and generating the document relationship diagram corresponding to the associated document set based on a connectivity relationship establishment result.

[0058] Based on this, when constructing a document relationship diagram, graph nodes may be first draw based on the associated documents in the associated document set, and graph edges are draw based on the similarity between the associated documents in the associated document set. When drawing the graph edges, the weight of the edge is determined by the similarity between the associated documents. Then, a connectivity relationship between graph

nodes may be established by combining the graph edges. On this basis, the document relationship diagram corresponding to the associated document set may be generated based on an establishment result of the connectivity relationship, so as to select and splice highly associated documents subsequently.

5 **[0059]** Following the previous example, after obtaining n documents with associated image processing knowledge, n graph nodes may be drawn based on the documents, and graph edges may be constructed based on the similarity between the n documents. On this basis, the n graph nodes may be connected based on the graph edges to obtain an undirected weighted graph, which may be used to select a set number of documents for splicing by analyzing the
10 undirected weighted graph, and then the LLaMA model is pre-trained.

[0060] In summary, by constructing a document relationship diagram to reflect the association between associated documents, it is facilitated to select highly associated documents for splicing, thereby improving the pre-training effect of the initial language model.

15 **[0061]** In step S206, a document splicing path is determined in the document relationship diagram, and target associated documents are extracted from the associated document set based on the document splicing path for splicing processing, to obtain target context information.

[0062] Specifically, after constructing the document relationship diagram as described
20 above, since the document relationship diagram can reflect the strength of the association between the associated documents, the document relationship diagram may be traversed to find the document splicing path. Highly relevant associated documents may be determined based on the document splicing path. In this case, the target associated documents may be extracted from the associated document set based on the document splicing path for splicing
25 processing, and the target context information that may be inputted into the model for processing may be obtained. Based on this, the language model may be trained by using the highly associated documents, so that the model can obtain the context comprehension and inference capability from the sequence of associated documents.

[0063] The document splicing path specifically refers to a link composed of nodes
30 corresponding to documents with high relevance found based on the traversal result of traversing the document relationship diagram. This path is composed of multiple graph nodes,

and the associated documents corresponding to the graph nodes are highly relevant documents. Correspondingly, the target associated document refers to the associated documents involved in the document splicing path. Correspondingly, the target context information specifically refers to a text obtained by splicing the target associated documents associated with the document splicing path.

[0064] Furthermore, when determining the document splicing path in the document relationship diagram, considering that this path will determine the association between similar documents in the target context information, in order to ensure that the target context information is constructed by using strongly associated documents to improve the comprehension and inference capability of the model across documents, a greedy algorithm may be used to determine the document splicing path. In this embodiment, the implementation is specifically described as follows:

determining an i -th graph node in the document relationship diagram and constructing an initial document path corresponding to the i -th graph node, where i is a positive integer and starts from 1, and a maximum value of i is the number of graph nodes in the document relationship diagram; selecting a j -th graph node from neighboring graph nodes connected to the i -th graph node, and updating the initial document path to an intermediate document path based on the j -th graph node, where j is a positive integer and starts from 1, and a maximum value of j is the number of the neighboring graph nodes; and determining whether there are any unvisited neighboring nodes in the neighboring graph nodes connected to the i -th graph node, where

if there are any unvisited neighboring nodes in the neighboring graph nodes connected to the i -th graph node, j is increased by 1 and the step of selecting a j -th graph node from neighboring graph nodes connected to the i -th graph node is performed; and if there are no unvisited neighboring nodes in the neighboring graph nodes connected to the i -th graph node, i is increased by 1 and the step of determining an i -th graph node in the document relationship diagram is performed until i equals the number of nodes in the graph nodes, multiple intermediate document paths are obtained, and an intermediate document path that meets the document splicing condition is extracted from the multiple intermediate document paths as the document splicing path.

[0065] Specifically, the i -th graph node refers to a graph node with the least number of

edges in the document relationship diagram, and the initial document path refers to an initial path constructed with the i -th graph node as a starting point of the path. Correspondingly, the j -th graph node specifically refers to an unvisited neighboring node in the neighboring nodes of the i -th graph node. Correspondingly, the intermediate document path specifically refers to a path obtained by adding the j -th graph node to the initial document path.

[0066] Based on this, when determining the document splicing path, in order to extract a path composed of nodes corresponding to highly associated documents from the document relationship diagram, an i -th graph node with the lowest dimension in the document relationship diagram can be determined based on the graph dimension, and the initial document path corresponding to the i -th graph node may be constructed, where i is a positive integer and starts from 1, and the maximum value of i is the number of graph nodes in the document relationship diagram.

[0067] Then, an unvisited j -th graph node may be selected from the neighboring graph nodes connected to the i -th graph node, and the j -th graph node may be added to the initial document path to obtain an intermediate document path, where j is a positive integer and starts from 1, and the maximum value of j is the number of neighboring graph nodes.

[0068] On this basis, it may be determined whether there are any unvisited neighboring nodes in the neighboring graph nodes connected to the i -th graph node. If there are any unvisited neighboring nodes in the neighboring graph nodes connected to the i -th graph node, it indicates that the document path corresponding to the i -th graph node may still be extended, j may be increased by 1, and the step of selecting the j -th graph node from the neighboring graph nodes connected to the i -th graph node is performed. If there are no unvisited neighboring nodes in the neighboring graph nodes connected to the i -th graph node, it indicates that the extension path of the i -th graph node has reached the upper limit, other nodes with the lowest dimension than the i -th graph node may be selected for path construction. Therefore, i may be increased by 1 and the step of determining the i -th graph node in the document relationship diagram is performed. In a case that i equals the number of graph nodes in the graph, it indicates that paths corresponding to all the graph nodes in the document relationship diagram have been constructed, multiple intermediate document paths may be obtained, and an intermediate document path that meets the document splicing condition may be extracted from the multiple intermediate document paths as the document

splicing path. In specific implementation, the longest intermediate document path may be selected as the document splicing path.

[0069] In practice, considering that the document relationship diagram is an undirected weighted graph, in order to fully explore the undirected weighted graph and quickly and accurately determine the target associated documents through graph analysis, a maximum traveling salesman path (a document splicing path) may be explored based on the concept that the greedy algorithm may be used to solve the maximum traveling salesman problem. That is, the path may be initialized first by starting from a node with the lowest dimension (a node with the smallest number of connections to other nodes) and constructing an initial path. Afterwards, path extension is performed by selecting unvisited neighboring nodes with the highest weight of the current node, which are added to the initial path to update the initial path. Similarly, when all neighboring nodes are visited, the current path construction is completed. In this case, a next node with the lowest dimension may be selected to continue path expansion until path construction is performed for all nodes, thus the maximum travel salesman path, i.e. the document splicing path, may be selected from the constructed path. Based on this, documents may be selected for splicing based on the document splicing path to obtain target context information for model pre-training.

[0070] In summary, by solving the problem of the maximum traveling salesman path through the greedy algorithm, it is possible to maximize continuous access to associated documents in the document relationship diagram, thereby improving the coherence of the inputted context.

[0071] Furthermore, after obtaining the document splicing path, since the path not only reflects the required document content but also has directionality, the target associated documents may be spliced based on the directional information. In this embodiment, the implementation is specifically described as follows:

determining splicing graph nodes in the document splicing path, and extracting the target associated documents from the associated document set based on the splicing graph nodes; and splicing the target associated documents based on path direction information of the document splicing path to obtain target context information that matches an input condition of the initial language model.

[0072] Specifically, the splicing graph node refers to a graph node in the document splicing

path. Correspondingly, the path direction information specifically refers to information that characterizes a relationship of documents before and after splicing. Based on this, after obtaining the document splicing path, the splicing graph node in the document splicing path may be determined. On this basis, the target associated document may be extracted from the associated document set based on the splicing graph node. Then, in order to ensure the coherence of a splicing result, the target associated documents may be spliced based on the path direction information of the document splicing path. Based on the splicing result, the target context information that matches the input condition of the initial language model may be obtained, which can be used for pre-training the initial language model subsequently.

[0073] Following the previous example, after obtaining the undirected weighted graph constructed based on the n documents, the undirected weighted graph may be processed by using the greedy algorithm to explore the maximum weight path. Specifically, the node with lowest dimension in the undirected weighted graph may be selected first, and used as a starting node to construct the initial path. On this basis, unvisited neighboring nodes with the highest weight of the graph node may be selected and added to the initial path to construct an intermediate path. This process is repeated until all neighboring nodes are visited, and then other graph node with the lowest dimension than the current node may be selected for path exploring. Finally, by traversing each node in the graph, the maximum weighted path may be generated, which includes a document 1, a document 2, a document 3, and a document 5. Based on this, the document 1, the document 2, the document 3, and the document 5 may be spliced along the maximum weight path to obtain an input context with a fixed size for subsequent model pre-training.

[0074] In summary, since the document splicing path is obtained by exploring based on the document relationship diagram, it can ensure the coherence of the context and the uniqueness of each target associated document when generating context information through document splicing based on this path, thereby solving the problem of data duplication, thus improving the training effect of the model.

[0075] In step S208, the initial language model is trained based on the target context information until a target language model that meets a training stopping condition is obtained.

[0076] Specifically, after obtaining the target context information by splicing the target associated documents as described above, further, since the target context information is

obtained by splicing the highly relevant associated documents, after training the initial language model based on the target context information until obtaining a target language model that meets the training stopping condition, the context comprehension and the inference capability of the target language model can be effectively improved, thereby significantly improving the performance of the target language model in complex tasks.

[0077] Furthermore, during the model training process, in order to obtain a target language model that meets deployment requirements, model detection may be performed during training. In this embodiment, the implementation is specifically described as follows:

inputting the target context information into the initial language model for processing, to obtain predication text information; optimizing the initial language model based on the predication text information and reference text information corresponding to the target context information; determining whether the optimized initial language model meets the training stopping condition; selecting, if the optimized initial language model does not meet the training stopping condition, candidate associated documents from the document set to form a candidate associated document set, and using the candidate associated document set as the associated document set, and performing the step of constructing a document relationship diagram corresponding to the associated document set based on the associated documents in the associated document set and similarity between the associated documents; and using the optimized initial language model as the target language model if the optimized initial language model meets the training stopping condition.

[0078] Specifically, the prediction text information specifically refers to prediction information obtained by the initial language model after processing of the target context information. Accordingly, the reference text information specifically refers to real information corresponding to the target context information. The model is optimized based on the prediction text information and the reference text information, so that the model can learn the knowledge required for training, thus meeting the business deployment requirements.

[0079] Based on this, after obtaining the target context information, the target context information may be inputted to the initial language model for processing, and the text information may be predicted based on the processing result. Thereafter, the reference text information corresponding to the target context information may be obtained, and the initial language model may be optimized based on the reference text information and prediction text

information. In a case that the optimization is completed, it may be detected whether the optimized initial language model meets the training stopping condition; if the optimized initial language model does not meet the training stopping condition, it means that the optimized initial language model cannot meet the deployment condition, and candidate associated documents may be selected from the document set to form a candidate associated document set, which may be used as the associated document set, and the step of constructing a document relationship diagram corresponding to the associated document set based on the associated documents in the associated document set and the similarity between the associated documents is performed, until a model that meets the condition is obtained, then the optimized initial language model may be used as a target language model to be deployed for use in a business scenario.

[0080] In practice, when pre-training the initial language model by using the generated target context information, the trained initial language model may adopt the same model architecture as that in the existing pre-training, and may be optimized using the AdamW optimizer and cosine learning rate scheduling. The context comprehension and inference capability of the model may further be improved by increasing the length of context information in the training batches.

[0081] Following the previous example, after obtaining the context information by splicing the document 1, the document 2, the document 3, and the document 5, the LLaMA model may be pre-trained based on the context information. The trained LLaMA model can enhance the comprehension and inference capability across documents. For example, when a user inputs a question {what steps are included in image sharpening process}, the question is inputted into the trained LLaMA model, the LLaMA model may first determine multiple documents associated with the question. By splicing the multiple documents, the LLaMA model may generate an answer based on the question of the user {1. selecting an image; 2. selecting a sharpening method; 3. applying a sharpening select...}. Finally, the answer may be fed back to the user.

[0082] With the method for training a language model according to the embodiment, in order to improve the performance of the language model in processing complex contexts and inferring across documents, after the document set of the initial language model is acquired, associated documents with similarity are selected from the document set of the initial

language model to form an associated document set. Afterwards, in order to enhance the comprehension capability of the model to text across documents, a document relationship diagram corresponding to the associated document set may be constructed based on the associated documents in the associated document set and the similarity between the associated documents, where the similarity relationship between the associated documents may be reflected through the document relationship diagram. On this basis, the document splicing path may be determined in the document relationship diagram, and the target associated documents may be extracted from the associated document set based on the document splicing path for splicing processing to obtain the target context information. Finally, the initial language model may be trained based on the target context information until the target language model that meets the training stopping condition is obtained. The model can be pre-trained based on a sequence of associated documents, so that the language model can read and infer across documents, thereby effectively improving the performance of the model in complex context tasks.

[0083] Referring to Figure 3, Figure 3 shows a flowchart of an information processing method according to an embodiment of the present disclosure, which includes the following steps S302 to S306.

[0084] In step S302, question information inputted by a client is acquired, and multiple answer documents are searched in a business knowledge database based on the question information.

[0085] In step S304, the multiple answer documents are spliced, and a splicing result and the question information are inputted into a target language model for processing to obtain answer information. The target language model is trained by using the above method.

[0086] In step S306, the answer information is fed back to the client.

[0087] The target language model described in the embodiment of the information processing method may refer to the description in the above embodiments, which will not be repeated in this embodiment.

[0088] The information processing method and the method for training a language model is further described with reference to Figure 4 by using an example of the application of the information processing method and the method for training a language model according to the

present disclosure to a question-and-answer interaction scenario. Figure 4 shows a flowchart of a processing procedure of an information processing method according to an embodiment of the present disclosure, which specifically includes the following steps S402 to S424.

[0089] In step S402, a document set of an initial language model is acquired, and document embedding vectors corresponding to respective initial documents in the document set is constructed by a text retrieval model.

[0090] In step S404, cosine similarity between the document embedding vectors corresponding to the respective initial documents are calculated.

[0091] In step S406, associated documents are selected in the document set based on a cosine similarity calculation result, to form an associated document set.

[0092] In step S408, graph nodes are constructed based on the associated documents in the associated document set, and graph edges are constructed based on the similarity between the associated documents in the associated document set.

[0093] In step S410, a connectivity relationship between the graph nodes is established based on the graph edges, and a document relationship diagram corresponding to the associated document set is generated based on a connectivity relationship establishment result.

[0094] In step S412, a document splicing path is determined in the document relationship diagram by using a greedy algorithm.

[0095] In step S414, splicing graph nodes in the document splicing path are determined, and target associated documents are extracted from the associated document set based on the splicing graph nodes.

[0096] In step S416, the target associated documents are spliced based on path direction information of the document splicing path to obtain target context information that matches an input condition of the initial language model.

[0097] In step S418, the initial language model is trained based on the target context information until a target language model that meets a training stopping condition is obtained, and the target language model is deployed.

[0098] In step S420, upon receiving question information inputted by a client, multiple

answer documents are searched for in a business knowledge database based on the question information.

[0099] In step S422, the multiple answer documents are spliced, and a splicing result and the question information are inputted into the target language model for processing to obtain answer information.

[0100] In step S424, the answer information is fed back to the client.

[0101] In summary, in order to improve the performance of the language model in processing complex contexts and inferring across documents, after the document set of the initial language model is acquired, associated documents with similarity are selected from the document set of the initial language model to form an associated document set. Afterwards, in order to enhance the comprehension capability of the model to text across documents, a document relationship diagram corresponding to the associated document set may be constructed based on the associated documents in the associated document set and the similarity between the associated documents, where the similarity relationship between the associated documents may be reflected through the document relationship diagram. On this basis, the document splicing path may be determined in the document relationship diagram, and the target associated documents may be extracted from the associated document set based on the document splicing path for splicing processing to obtain the target context information. Finally, the initial language model may be trained based on the target context information until the target language model that meets the training stopping condition is obtained. The model can be pre-trained based on a sequence of associated documents, so that the language model can read and infer across documents, thereby effectively improving the performance of the model in complex context tasks.

[0102] Corresponding to the above method embodiments, an embodiment of a device for training a language model is further provided according to the present disclosure. Figure 5 shows a schematic diagram of a structure of a device for training a language model according to an embodiment of the present disclosure. As shown in Figure 5, the device includes: an acquisition module 502, a construction module 504, a splicing module 506 and a training module 508. The acquisition module 502 is configured to acquire a document set of an initial language model and select associated documents in the document set to form an associated document set. The construction module 504 is configured to construct a document

relationship diagram corresponding to the associated document set based on the associated documents in the associated document set and similarity between the associated documents. The splicing module 506 is configured to determine a document splicing path in the document relationship diagram, and extract target associated documents from the associated document set based on the document splicing path for splicing processing to obtain target context information. The training module 508 is configured to train the initial language model based on the target context information until obtaining a target language model that meets a training stopping condition.

[0103] In a preferred embodiment, the acquisition module 502 is further configured to:

construct, by a text retrieval model, document embedding vectors corresponding to respective initial documents in the document set; calculate cosine similarity between the document embedding vectors corresponding to the respective initial documents; and select the associated documents in the document set based on a cosine similarity calculation result to form the associated document set.

[0104] In a preferred embodiment, the acquisition module 502 is further configured to:

quantize, by a preset quantizer, the document embedding vectors corresponding to the respective initial documents, and construct a document index based on the quantized document embedding vectors; and search the document set according to a preset search strategy and the document index, and determine the associated documents based on a search result to form the associated document set.

[0105] In a preferred embodiment, the construction module 504 is further configured to:

construct graph nodes based on the associated documents in the associated document set, and constructing graph edges based on the similarity between the associated documents in the associated document set; and establish a connectivity relationship between the graph nodes based on the graph edges, and generate the document relationship diagram corresponding to the associated document set based on a connectivity relationship establishment result.

[0106] In a preferred embodiment, the splicing module 506 is further configured to:

determine an i-th graph node in the document relationship diagram and construct an initial document path corresponding to the i-th graph node, where i is a positive integer and starts from 1, and a maximum value of i is the number of graph nodes in the document

relationship diagram;

select a j-th graph node from neighboring graph nodes connected to the i-th graph node, and update the initial document path to an intermediate document path based on the j-th graph node, where j is a positive integer and starts from 1, and a maximum value of j is the number of the neighboring graph nodes; and

determine whether there are any unvisited neighboring nodes in the neighboring graph nodes connected to the i-th graph node, where

if there are any unvisited neighboring nodes in the neighboring graph nodes connected to the i-th graph node, j is increased by 1 and the step of selecting a j-th graph node from neighboring graph nodes connected to the i-th graph node is performed; and

if there are no unvisited neighboring nodes in the neighboring graph nodes connected to the i-th graph node, i is increased by 1 and the step of determining an i-th graph node in the document relationship diagram is performed until i equals the number of the graph nodes, multiple intermediate document paths are obtained, and an intermediate document path that meets a document splicing condition is extracted from the multiple intermediate document paths as the document splicing path.

[0107] In a preferred embodiment, the splicing module 506 is further configured to:

determine splicing graph nodes in the document splicing path, and extract the target associated documents from the associated document set based on the splicing graph nodes; and

splice the target associated documents based on path direction information of the document splicing path to obtain target context information that matches an input condition of the initial language model.

[0108] In a preferred embodiment, the training module 508 is further configured to:

input the target context information into the initial language model for processing, to obtain predication text information;

optimize the initial language model based on the predication text information and reference text information corresponding to the target context information;

determine whether the optimized initial language model meets the training stopping

condition;

select, if the optimized initial language model does not meet the training stopping condition, candidate associated documents from the document set to form a candidate associated document set, and use the candidate associated document set as the associated document set, and perform the step of constructing a document relationship diagram corresponding to the associated document set based on the associated documents in the associated document set and similarity between the associated documents; and

use the optimized initial language model as the target language model if the optimized initial language model meets the training stopping condition.

[0109] A schematic solution of a device for training a language model of the embodiment is described above. It should be noted that the technical solution of the device for training a language model and the technical solution of the method for training a language model described above belong to the same concept, and the details not described in detail in the technical solution of the device for training a language model may be referred to the description of the technical solution of the method for training a language model described above.

[0110] Corresponding to the above method embodiment, an embodiment of an information processing device is further provided according to the present disclosure. Figure 6 shows a schematic diagram of a structure of an information processing device according to an embodiment of the present disclosure. As shown in Figure 6, the device includes: an information acquisition module 602, a model prediction module 604 and an information feedback module 606. The information acquisition module 602 is configured to acquire question information inputted by a client, and search for multiple answer documents in a business knowledge database based on the question information, splice the multiple answer documents and input a splicing result and the question information into a target language model for processing to obtain answer information, where the target language model is trained by the above method. The information feedback module 606 is configured to feed back the answer information to the client.

[0111] A schematic solution of the information processing device of the embodiment is described above. It should be noted that the technical solution of the information processing device and the technical solution of the information processing method described above

belong to the same concept, and the details not described in detail in the technical solution of the information processing device may be referred to the description of the technical solution of the information processing method described above.

[0112] Figure 7 illustrates a block diagram of a structure of a computing device 700 according to an embodiment of the present disclosure. The components of the computing device 700 include, but are not limited to, a memory 710 and a processor 720. The processor 720 is connected to the memory 710 via a bus 730, and a database 750 is used to store data.

[0113] The computing device 700 further includes an access device 740, the access device 740 enables the computing device 700 to communicate via one or more networks 760.

Examples of these networks include a public switched telephone network (PSTN), a local area network (LAN), a wide area network (WAN), a personal area network (PAN), or a combination of communication networks such as the Internet. The access device 740 may include one or more of any types of wired or wireless network interfaces (e.g., a network interface controller (NIC), such as an IEEE802.11 Wireless Local Area Network (WLAN) wireless interface, Worldwide Interoperability for Microwave Access (Wi-MAX) interface, Ethernet interface, Universal Serial Bus (USB) interface, cellular network interface, Bluetooth interface, Near Field Communication (NFC).

[0114] In an embodiment of the present disclosure, the above components of the computing device 700, as well as other components not shown in Figure 7, may also be connected to each other, such as via a bus. It should be understood that the block diagram of the structure of the computing device shown in Figure 7 is for exemplary purposes only and is not a limitation on the scope of the present disclosure. Those skilled in the art may add or replace other components as desired.

[0115] The computing device 700 may be any type of stationary or mobile computing device, including a mobile computer or a mobile computing device (e.g., a tablet computer, a personal digital assistant, a laptop computer, a notebook computer, a netbook, etc.), a cell phone (e.g., a smartphone), a wearable computing device (e.g., a smartwatch, smartglasses, etc.), or other types of mobile devices, or a stationary computing device such as desktop computer or a personal computer (PC). The computing device 700 may also be a mobile or stationary server.

[0116] The processor 720 is used to execute computer executable instructions, which, when

executed by the processor, perform the steps of the above-described method for training a language model or information processing method.

[0117] A schematic solution of the computing device of the other embodiment is described above. It should be noted that the technical solution of the computing device and the technical solution of the method for training a language model or the information processing method described above belong to the same concept, and the details not described in detail in the technical solution of the computing device may be referred to the description of the technical solution of the method for training a language model or the information processing method described above.

[0118] A computer-readable storage medium is further provided according to an embodiment of the present disclosure, which stores computer-executable instructions, the computer-executable instructions, when executed by a processor, perform the steps of the above-described method for training a language model or information processing method.

[0119] A schematic solution of the computer-readable storage medium of the other embodiment is described above. It should be noted that the technical solution of the computer-readable storage medium and the technical solution of the method for training a language model or the information processing method described above belong to the same concept, and the details not described in detail in the technical solution of the computer-readable storage medium may be referred to the description of the technical solution of the method for training a language model or the information processing method described above.

[0120] A computer program product is further provided according to an embodiment of the present disclosure, which includes computer programs or instructions, the computer programs or instructions, when executed by a processor, perform the steps of the above-described method for training a language model or information processing method.

[0121] A schematic solution of the computer program product of the other embodiment is described above. It should be noted that the technical solution of the computer program product and the technical solution of the method for training a language model or the information processing method described above belong to the same concept, and the details not described in detail in the technical solution of computer program product may be referred to the description of the technical solution of the method for training a language model or the

information processing method described above.

[0122] Preferred embodiments of the present disclosure are described above. Other embodiments are within the scope of the claims. In some cases, the actions or steps described in the claims may be performed in a different order than in the embodiments and still achieve the desired results. In addition, the processes depicted in the drawings do not necessarily require the particular order or consecutive order shown to achieve the desired results. In some embodiments, multi-tasking and parallel processing is also possible or may be advantageous.

[0123] The computer instruction includes computer program codes, the computer program code may be in the form of source code, in the form of object code, in the form of an executable file or in some intermediate form or the like. The computer readable medium may include: any entity or device capable of carrying the computer program code, a recording medium, a USB disk, a removable hard disk, a diskette, an optical disk, a computer memory, a Read-Only Memory (ROM), a Random Access Memory (RAM), an electric carrier signal, a telecommunications signal, and software distribution medium, etc. It should be noted that the content of the computer-readable medium may be increased or decreased according to the requirements of the patent practice, for example, in some regions, according to the patent practice, the computer-readable medium does not include carrier signals and telecommunication signals.

[0124] It should be noted that, for each of the foregoing method embodiments, for the sake of simplicity of description, they are all expressed as a series of combinations of actions, but those skilled in the art should know that the embodiments of the present disclosure are not limited by the order of the described actions, because according to the embodiments of the present disclosure, certain steps may be performed in other orders or simultaneously. Secondly, those skilled in the art should also be aware that the embodiments described in the disclosure are preferred embodiments, and the actions and modules involved are not necessarily necessary for the embodiments of the present disclosure.

[0125] In the above-described embodiments, the description of each embodiment has its own focus, and the part not described in detail in a certain embodiment can be referred to the relevant description of other embodiments.

[0126] The above preferred embodiments of the present disclosure are intended only to aid in the description of the present disclosure. The preferred embodiments are not an exhaustive

recitation of all details, nor do they limit the invention to the specific embodiments described only. Apparently, many modifications and variations may be made in accordance with the contents of the embodiments of the present disclosure. These embodiments are selected and specifically described in the present disclosure to better explain the principles and practice of
5 the embodiments of the present disclosure so that those skilled in the art to which they belong will be able to well understand and utilize the present disclosure.

CLAIMS

1. A method for training a language model, comprising:

acquiring a document set of an initial language model, and selecting associated
5 documents in the document set to form an associated document set;

constructing a document relationship diagram corresponding to the associated document
set based on the associated documents in the associated document set and similarity between
the associated documents;

determining a document splicing path in the document relationship diagram, and
10 extracting target associated documents from the associated document set based on the
document splicing path for splicing processing, to obtain target context information; and

training the initial language model based on the target context information until
obtaining a target language model that meets a training stopping condition.

2. The method for training a language model according to claim 1, wherein the selecting
15 associated documents in the document set to form an associated document set comprises:

constructing, by a text retrieval model, document embedding vectors corresponding to
respective initial documents in the document set;

calculating cosine similarity between the document embedding vectors corresponding to
the respective initial documents; and

20 selecting the associated documents in the document set based on a cosine similarity
calculation result to form the associated document set.

3. The method for training a language model according to claim 2, wherein after the
constructing, by a text retrieval model, document embedding vectors corresponding to
respective initial documents in the document set, the method further comprises:

25 quantizing, by a preset quantizer, the document embedding vectors corresponding to the
respective initial documents, and constructing a document index based on the quantized
document embedding vectors; and

searching the document set according to a preset search strategy and the document index,
and determining the associated documents based on a search result to form the associated

document set.

4. The method for training a language model according to claim 1, wherein the constructing a document relationship diagram corresponding to the associated document set based on the associated documents in the associated document set and similarity between the associated documents comprises:

constructing graph nodes based on the associated documents in the associated document set, and constructing graph edges based on the similarity between the associated documents in the associated document set; and

establishing a connectivity relationship between the graph nodes based on the graph edges, and generating the document relationship diagram corresponding to the associated document set based on a connectivity relationship establishment result.

5. The method for training a language model according to claim 1, wherein the determining a document splicing path in the document relationship diagram comprises:

determining an i-th graph node in the document relationship diagram and constructing an initial document path corresponding to the i-th graph node, wherein i is a positive integer and starts from 1, and a maximum value of i is the number of graph nodes in the document relationship diagram;

selecting a j-th graph node from neighboring graph nodes connected to the i-th graph node, and updating the initial document path to an intermediate document path based on the j-th graph node, wherein j is a positive integer and starts from 1, and a maximum value of j is the number of the neighboring graph nodes; and

determining whether there are any unvisited neighboring nodes in the neighboring graph nodes connected to the i-th graph node, wherein

if there are any unvisited neighboring nodes in the neighboring graph nodes connected to the i-th graph node, j is increased by 1 and the step of selecting a j-th graph node from neighboring graph nodes connected to the i-th graph node is performed; and

if there are no unvisited neighboring nodes in the neighboring graph nodes connected to the i-th graph node, i is increased by 1 and the step of determining an i-th graph node in the document relationship diagram is performed until i equals the number of the graph

nodes, a plurality of intermediate document paths are obtained, and an intermediate document path that meets a document splicing condition is extracted from the plurality of intermediate document paths as the document splicing path.

6. The method for training a language model according to claim 1, wherein the extracting target associated documents from the associated document set based on the document splicing path for splicing processing, to obtain target context information comprises:

determining splicing graph nodes in the document splicing path, and extracting the target associated documents from the associated document set based on the splicing graph nodes; and

splicing the target associated documents based on path direction information of the document splicing path to obtain target context information that matches an input condition of the initial language model.

7. The method for training a language model according to claim 1, wherein the training the initial language model based on the target context information until obtaining a target language model that meets a training stopping condition comprises:

inputting the target context information into the initial language model for processing, to obtain predication text information;

optimizing the initial language model based on the predication text information and reference text information corresponding to the target context information;

determining whether the optimized initial language model meets the training stopping condition;

selecting, if the optimized initial language model does not meet the training stopping condition, candidate associated documents from the document set to form a candidate associated document set, and using the candidate associated document set as the associated document set, and performing the step of constructing a document relationship diagram corresponding to the associated document set based on the associated documents in the associated document set and similarity between the associated documents; and

using the optimized initial language model as the target language model if the optimized initial language model meets the training stopping condition.

8. An information processing method, comprising:

acquiring question information inputted by a client, and searching for a plurality of answer documents in a business knowledge database based on the question information;

splicing the plurality of answer documents and inputting a splicing result and the question information into a target language model for processing to obtain answer information, wherein the target language model is trained by the method according to any one of claims 1 to 7; and

feeding back the answer information to the client.

9. A device for training a language model, comprising:

an acquisition module, configured to acquire a document set of an initial language model and select associated documents in the document set to form an associated document set;

a construction module, configured to construct a document relationship diagram corresponding to the associated document set based on the associated documents in the associated document set and similarity between the associated documents;

a splicing module, configured to determine a document splicing path in the document relationship diagram, and extract target associated documents from the associated document set based on the document splicing path for splicing processing to obtain target context information;

a training module, configured to train the initial language model based on the target context information until obtaining a target language model that meets a training stopping condition.

10. An information processing device, comprising:

an information acquisition module, configured to acquire question information inputted by a client, and searching for a plurality of answer documents in a business knowledge database based on the question information;

a model prediction module, configured to splice the plurality of answer documents and inputting a splicing result and the question information into a target language model for processing to obtain answer information, wherein the target language model is trained by the method according to any one of claims 1 to 7; and

an information feedback module, configured to feed back the answer information to the client.

11. A computing device, comprising:

a memory, configured to store computer executable instructions; and

5 a processor, configured to execute the computer executable instructions, wherein the computer executable instructions, when executed by the processor, perform the steps of the method according to any one of claims 1 to 8.

12. A computer-readable storage medium, comprising computer executable instructions stored thereon, wherein the computer executable instructions, when executed by a processor,

10 perform the steps of the method according to any one of claims 1 to 8.

13. A computer program product, comprising computer programs or instructions, wherein the computer programs or instructions, when executed by a processor, perform the steps of the method according to any one of claims 1 to 8.