



(51) International Patent Classification:
G06F 16/00 (2019.01)

(21) International Application Number:
PCT/CN2019/073262

(22) International Filing Date:
25 January 2019 (25.01.2019)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **MICROSOFT TECHNOLOGY LICENSING, LLC** [US/US]; One Microsoft Way, Redmond, Washington 98052 (US).

(72) Inventors; and

(71) Applicants (*for BZ only*): **DENG, Weiwei** [CN/CN]; One Microsoft Way, Redmond, Washington 98052 (US). **GU, Chen** [CN/CN]; One Microsoft Way, Redmond, Washington 98052 (US). **TAN, Tunzi** [CN/CN]; One Microsoft Way, Redmond, Washington 98052 (US). **FU, Ding**

[CN/CN]; One Microsoft Way, Redmond, Washington 98052 (US). **FANG, Yixin** [CN/CN]; One Microsoft Way, Redmond, Washington 98052 (US). **LUO, Dan** [CN/CN]; One Microsoft Way, Redmond, Washington 98052 (US). **ZHANG, Qi** [CN/CN]; One Microsoft Way, Redmond, Washington 98052 (US).

(74) Agent: **NTD PATENT & TRADEMARK AGENCY LTD.**; 10th Floor, Tower C, Beijing Global Trade Center, 36 North Third Ring Road East, Dongcheng District, Beijing 100013 (CN).

(81) Designated States (*unless otherwise indicated, for every kind of national protection available*): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA,

(54) Title: SCORING DOCUMENTS IN DOCUMENT RETRIEVAL

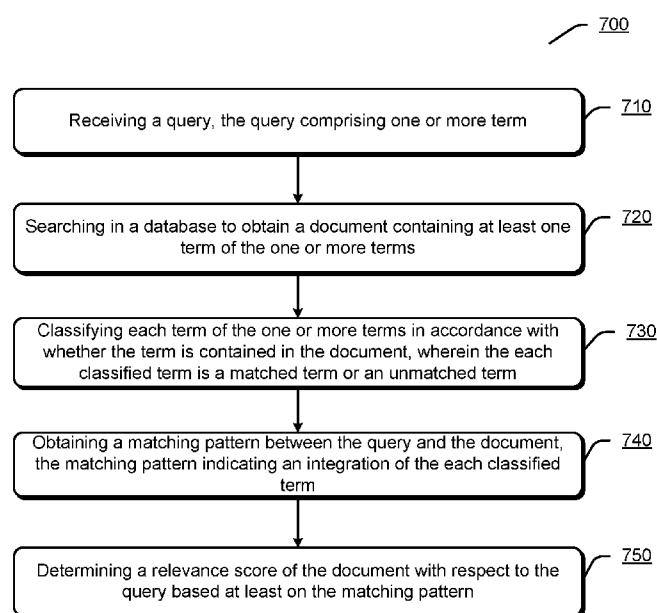


FIG 7

(57) Abstract: A method and apparatus is provided for scoring documents in document retrieval. A query may be received, in which the query comprises one or more terms. A database may be searched to obtain a document containing at least one term of the one or more terms. Each term of the one or more terms may be classified in accordance with whether the term is contained in the document, each classified term being a matched term or an unmatched term. A matching pattern between the query and the document may be obtained, the matching pattern indicating an integration of the each classified term. A relevance score of the document with respect to the query may be determined based at least on the matching pattern.

SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN,
TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *of inventorship (Rule 4.17(iv))*

Published:

— *with international search report (Art. 21(3))*

SCORING DOCUMENTS IN DOCUMENT RETRIEVAL

BACKGROUND

[0001] Information retrieval, such as document retrieval, deals with representation, storage, organization of information items, and access to the information items. In document retrieval, a plurality of documents may be obtained for a given query through searching. Inverted index has been widely used in document searching, especially for large document sets such as web search, sponsored search and e-commercial search. When a plurality of documents is obtained, scoring process may be applied to the obtained documents to determine respective score of the document. Then top-k documents for the given query may be selected and presented from the plurality of searched documents based on their scores determined by certain scoring methods.

SUMMARY

[0002] This Summary is provided to introduce a selection of concepts that are further described below in the Detailed Description. It is not intended to identify key features or essential features of the claimed subject matter, nor is it intended to be used to limit the scope of the claimed subject matter.

[0003] Embodiments of the present disclosure propose method and apparatus for scoring documents in document retrieval. In some implementations, a query may be received, in which the query comprises one or more terms. A database may be searched to obtain a document containing at least one term of the one or more terms. Each term of the one or more terms may be classified in accordance with whether the term is contained in the document, each classified term being a matched term or an unmatched term. A matching pattern between the query and the document may be obtained, the matching pattern indicating an integration of the each classified term. A relevance score of the document with respect to the query may be determined based at least on the matching pattern.

[0004] It should be noted that the above one or more aspects comprise the features hereinafter fully described and particularly pointed out in the claims. The following description and the drawings set forth in detail certain illustrative features of the one or more aspects. These features are only indicative of the various ways in which the

principles of various aspects may be employed, and this disclosure is intended to include all such aspects and their equivalents.

BRIEF DESCRIPTION OF THE DRAWINGS

[0005] The disclosed aspects will hereinafter be described in connection with the appended drawings that are provided to illustrate and not to limit the disclosed aspects.

[0006] FIG.1 illustrates an exemplary architecture of a system for document retrieval through a network according to an embodiment.

[0007] FIG.2 illustrates an exemplary process for document retrieval according to an embodiment.

[0008] FIG.3 illustrates an exemplary inverted index showing an exemplary matching pattern between a document and a query according to an embodiment.

[0009] FIG.4 illustrates an exemplary Deep Neural Network (DNN) scoring model for document retrieval according to an embodiment.

[0010] FIG.5 illustrates another exemplary process for document retrieval according to an embodiment.

[0011] FIG.6(A) illustrates an exemplary inverted index before cursors moving according to an embodiment.

[0012] FIG.6(B) illustrates another exemplary inverted index after cursors moving according to an embodiment.

[0013] FIG.7 illustrates a flowchart of an exemplary method for scoring documents in document retrieval according to an embodiment.

[0014] FIG.8 illustrates an exemplary apparatus for scoring documents in document retrieval according to an embodiment.

[0015] FIG.9 illustrates another exemplary apparatus for scoring documents in document retrieval according to an embodiment.

DETAILED DESCRIPTION

[0016] The present disclosure will now be discussed with reference to several example implementations. It is to be understood that these implementations are discussed only for enabling those skilled in the art to better understand and thus implement the embodiments of the present disclosure, rather than suggesting any limitations on the scope of the present disclosure.

[0017] In information retrieval (IR), such as document retrieval, given a query Q and a document database D , top- k documents may be returned according to some scoring functions, $score(d, Q)$, $d \in D$. Here, a score of a document with respect to a query indicates relevance between a query and the document and it may be referred to as relevance score. As for top- k document retrieval in existing systems, in particular for text retrieval tasks, inverted index is usually a primary choice to catch original text representations. In document retrieval, relevance scores of documents are used to rank the corresponding documents so as to return top- k documents to be presented to a user. As for most of current scoring functions, a re-ranking process may be needed to refine candidates for the best top- k documents. This may add additional cost to rank the candidates in two stages, such as top- k retrieval stage and re-ranking stage. Moreover, in the top- k retrieval stage of the current scoring functions, potential candidates may be lost due to limitation of the scoring functions, since the current scoring functions in the document retrieval are unsupervised, such as based on statistical formulas or probabilistic models. These scoring functions cannot be benefited from available labeled data sets and only matched terms in a query are considered in documents scoring.

[0018] Embodiments of the present invention propose a method for scoring documents in document retrieval, for example, by leveraging a supervised machine learning (ML) scoring function, which considers matched terms and unmatched terms between a query and a document, coexistence of matched and unmatched terms, and positions of unmatched and matched terms in the query. Herein, the matched terms represent terms in a query contained in a document and the unmatched terms represent terms in the query not contained in the document. In some implementations, logistic regression (LR) and deep neural network (DNN) models may be used as examples of the machine learning scoring models. Herein, a “term” in a query may represent an index unit, which may comprise one or more of characters, words, phrases, numbers, and so on. The presented machine learning scoring function outperforms traditional scoring functions on both accuracy and efficiency in document retrieval and may be applied to different tasks, such as web search or recommendation system. For example, the machine learning scoring function may be used in search recommendation, news recommendation, advertisement recommendation, product recommendation, question-answer application, etc.

[0019] FIG.1 illustrates an exemplary architecture of a system 100 for document retrieval through a network according to an embodiment.

[0020] In FIG.1, a network 110 is applied for interconnecting among a terminal device 120 and a server 130.

[0021] The network 110 may be any type of networks capable of interconnecting network entities. The network 110 may be a single network or a combination of various networks. In terms of coverage range, the network 110 may be a Local Area Network (LAN), a Wide Area Network (WAN), etc. In terms of carrying medium, the network 110 may be a wireline network, a wireless network, etc. In terms of data switching techniques, the network 110 may be a circuit switching network, a packet switching network, etc.

[0022] The terminal device 120 may be any type of electronic computing devices capable of connecting to the network 110, accessing servers or websites on the network 110, processing data or signals, etc. For example, the terminal device 120 may be desktop computers, laptops, tablets, smart phones, AI terminals, etc. Although only one terminal device is shown in FIG.1, it should be appreciated that a different number of terminal devices may connect to the network 110.

[0023] In an implementation, the terminal device 120 may comprise a query input component 121, a display interface 122, a memory 123, a processor 124, and at least one communication connection 125 for connecting to the network 110.

[0024] The query input component 121 may be configured to receive a search query from a user who is attempting to obtain retrieved information, such as documents, via an input device, not shown in FIG.1, in communication with the terminal device 120. For example, the input device may include a mouse, joystick, key pad, microphone, I/O components or any other component capable of receiving a user input and communicating an indication of that input to the terminal device 120. A search query or a query as used herein, may comprise one or more terms reflecting the user's intent for attempting to obtain retrieved information or documents including at least one term in the query from a database 160, such as a website, cloud storage, a document corpus, and so on. In some examples, the document retrieval may be performed by utilizing inverted indexes from the database 160. The database 160 may include a large number of documents, such as webpages, advertisements, news, and so on.

[0025] Top-k retrieved documents may be returned from the server 130 and presented to the user through the display interface 122. The display interface 122 may be specific to the terminal device 120, or be incorporated in a third-party device, such as, a displayer external to the terminal device 120, a TV, a projector, etc.

[0026] The memory 123 may be one or more devices for storing data, which may comprise readable-only memory (ROM), random access memory (RAM), magnetic RAM, core memory, magnetic disk storage medium, optical storage medium, flash memory, and/or other machine readable medium for storing information. The term of “machine readable medium” may comprise but not be limited to portable or fixed storage, optical storage, wireless channel and various other medium for storing, containing or carrying instructions and/or data.

[0027] The processor 124 may include any number of processing units, and is programmed to execute computer-executable instructions for implementing aspects of the disclosure. In some examples, the processor 124 is programmed to execute computer-executable instructions stored in the memory 123 to implement method such as illustrated in accompanying FIG.7.

[0028] The communication connection 125 may comprise wireless connection or wired connection to make the terminal device 120 connecting to the network 110 or any other device.

[0029] The server 130 may connect to the database 160. Although not shown in FIG.1, the server 130 may incorporate the database 160. The server 130 may provide data or retrieved documents to the terminal device 120. As one example, the server 130 may be a web server providing data over a web. The server 130 may provide the data over the web to the terminal device 120 through the network 110.

[0030] The server 130 may comprise a query receiving component 140 and a scoring model 150 including a feature weight index 151. The query receiving component 140 may receive the query inputted by the user from the terminal device 120, for example, through the network 110. The received query may be provided to the scoring model 150 for searching one or more documents matched with the query from the database 160. That is, the matched documents contain one or more terms in the query. A plurality of features associated with the query may be generated from the query, each having a weight in the feature weight index 151. Scores of the matched documents may be calculated or determined by the scoring model 150 based at least

one the features generated from the query and the corresponding weights in the feature weight index 151. Although the feature weight index 151 is shown as included in the scoring model 150 in FIG.1, it may be stored in a storage separated from the scoring model 150 and/or the server 130, and respective weight of each feature may be provided to the scoring model 150, if applicable. Top-k documents may be selected from the matched documents based on the scores of the matched documents and be provided to the terminal device 120, for example through the network 110, to be presented to the user through the display interface 122.

[0031] Although the query receiving component 140 and the scoring model 150 are shown as being incorporated in the server 130 in FIG.1, they may be configured as independent components separated from the server 130.

[0032] It should be understood that, all components shown in FIG.1 may be exemplary, and any components in the terminal device 120 may be added, omitted or replaced in FIG.1 depending on specific application requirements.

[0033] FIG.2 illustrates an exemplary process 200 for document retrieval according to an embodiment.

[0034] At 202, a query Q may be received, for example, from a user. The query may be a search query and contain one or more terms. As an example, a query “*home depot wood floor*” may be received herein. The exemplary terms contained in the query may be “*home*”, “*depot*”, “*wood*” and “*floor*”.

[0035] At 204, a database D may be searched to obtain documents containing at least one term of the one or more terms, for example, by utilizing inverted index structure. Herein, such documents may be referred to as “matched documents”. In the example shown herein, documents containing at least one term of “*home*”, “*depot*”, “*wood*” and “*floor*” may be obtained. Each of the matched documents may be assigned a DocID in the inverted index structure, where DocID is the ID of a document $d \in D$ containing the at least one term and may be shown as Doc 1, Doc 3, Doc7, etc. as shown below in a simple inverted index structure in FIG.3, or simply shown as <1>, <3>, <7>.... The inverted index structure used herein may comprise two main sections: dictionary and posting lists. The dictionary comprises various terms, which may be followed by some possible items, such as inverse document frequency (IDF), term frequency (TF), upperbound and entry address of related posting lists, which are not shown in the present disclosure. The followed items, such

as IDF, TF, etc., are well known and the detailed description for them is omitted herein for simplicity. Each posting list may contain a series of postings represented by DocIDs. Each term in the query is mapped to a posting list. For example, as for term 1 shown in FIG.3, which is “home” in this example, a posting list for term 1 in the query may be < Doc3, Doc7, Doc9, Doc13...>.

[0036] At 206, each term in the query may be classified in accordance with whether the term is contained in the document. The each classified term is a matched term or an unmatched term. For example, as shown in FIG.3, for document Doc3 which contains terms “home” and “floor”, term 1 (home), term 4 (floor) in a query “home depot wood floor” are classified as matched terms between the query and Doc3, and term 2 (depot) and term 3 (wood) are classified as unmatched terms.

[0037] At 208, respective matching pattern between each matched document and the query may be obtained. The matching pattern may indicate an integration of the each classified term in the query. Herein the integration may identify each term in the query, classification of the each term and position of the each term in the query. That is, the matching pattern may indicate respective position of each term in the query and classification of the each term. Herein the matching pattern is represented by a vector, and each dimension of the vector has a binary value, with 1 on a position of matched term in a query and 0 on a position of unmatched term in a query. If there are 4 terms in a query and for a particular document, there is only the first term matched, then the exemplary matching pattern between the query and the particular document may be represented as [1000], wherein “1” means that the pair of query and document has a matched term on this position, and “0” means that there is no matched term on these positions. Taking a query “home depot wood floor” as an example, if a matching pattern between the query and a particular matched document is represented as [1001], the matching pattern indicates that the particular matched document contains the first and fourth terms in the query and does not contain the second and third terms. An exemplary matching pattern determination will be illustrated in connection with FIG.3 described below.

[0038] At 210, respective relevance score of each matched document may be calculated with respect to the query, for example, through a machine learning scoring model based at least on the matching pattern obtained at 208. For example, a relevance score of a matched document may be determined by calculating a score

corresponding to a matching pattern between the query and the matched document. Herein the calculated score corresponding to a matching pattern may be referred to as “original score” below, for sake of explanation. The machine learning scoring model may comprise logistic regression (LR) model, gradient boosted decision tree (GBDT) model, and deep neural network (DNN) model and so on. Herein LR model and DNN model may be taken as exemplary scoring models below to calculate the relevance score of the matched document. It should be appreciated that any other appropriate scoring model may be utilized to implement the present disclosure according to particular practice requirements.

[0039] Taking the query Q “*home depot wood floor*” and a particular document A containing terms of “*home*” and “*floor*” as an example, the calculation for an original score corresponding to the matching pattern will be described below.

Original score calculation

[0040] From the above query Q and the particular document A, a matching pattern between the query Q and the document A may be obtained as [1001]. Based at least on the query Q, in particular, based on matched terms and unmatched terms in the query, and the matching pattern [1001], a plurality of related features of different types may be generated, each feature having a weight. In some examples, respective weight may be trained offline and pre-assigned to each feature. The original score corresponding to the matching pattern may be calculated based on the weights of the plurality of related features, for example, being considered as a sum of the weights of the related features.

[0041] The weight of each feature may be trained or learned from a set of training data. The training data may comprise a historical query and a plurality of retrieved historical documents. Each historical document may be labeled with a relevance score calculated based on e.g., a click number, etc., or assigned by a human. The weights of features may be updated periodically or on demand, according to a set of updated training data. For example, the updated training data may comprise at least one of a retrieved historical document with an updated or different score, or retrieved documents for a new query, etc.

[0042] In one implementation, the different types may be predefined or predesigned. For example, the following types may be predefined:

- QueryMatchedTerm: matched terms in the query,

- QueryUnmatchedTerm: unmatched terms in the query,
- QueryMatchedTerm Cross QueryUnmatchedTerm: a combination of anyone matched term and anyone unmatched term in the query,
- QueryMatchedTerm Cross QueryMatchedOtherTerm: a combination of anyone matched term and any another matched term in the query,
- QueryLength Cross QueryMatchedLength: a combination of the number of terms and the number of matched terms in the query,
- QueryContinuousMatchedTerms: continuous matched terms in the query,
- QueryMatchedChunk: a chunk of matched terms in the query, no matter if they are continuous,
- QueryMatchedUnmatchedBigram: bigram of anyone matched term and anyone unmatched term in the query, for example, two continuous terms with one matched and another unmatched,
- QueryUnmatchedBigram: bigram of unmatched terms in the query, for example, two continuous unmatched terms,
- QueryMatchedTerm Cross QueryMatchedPattern: a combination of anyone matched term in the query and a corresponding matching pattern,
- QueryUnmatchedTerm Cross QueryMatchedPattern: a combination of anyone unmatched term in the query and a corresponding matching pattern.

[0043] It should be understood that, although several feature types are listed above, there may be any other appropriate feature types predefined or predesigned according to particular implementation requirements.

[0044] Table 1 shows the exemplary feature types and features generated from the query Q “*home depot wood floor*” and the matching pattern [1001].

Feature Type ID	Feature Type	Features/Feature list
1	QueryMatchedTerm	home, floor
2	QueryUnmatchedTerm	depot, wood
3	QueryMatchedTerm Cross QueryUnmatchedTerm	home_×_depot, home_×_wood, floor_×_depot, floor_×_wood

4	QueryMatchedTerm QueryMatchedOtherTerm	Cross	home_×_floor
5	QueryLength QueryMatchedLength	Cross	4×2
6	QueryContinuousMatchedTerms		home, floor
7	QueryMatchedChunk		home floor
8	QueryMatchedUnmatchedBigram		home depot, wood floor
9	QueryUnmatchedBigram		depot wood
10	QueryMatchedTerm QueryMatchedPattern	Cross	home_×_1001, wood_×_1001
11	QueryUnmatchedTerm QueryMatchedPattern	Cross	depot_×_1001, floor_×_1001

Table 1

[0045] Herein LR model and DNN model will be discussed below as examples of implementing the machine learning scoring function through using the above features.

[0046] Logarithmic loss (LogLoss) is the loss function for these models, as indicated by Equation (1).

$$\text{LogLoss}(y, p) = -(y \log(p) + (1 - y) \log(1 - p)) \quad \text{Equation (1)}$$

where “y” stands for the label and $p(\bullet)$ is the predicted probability of model.

[0047] A LR model for scoring is represented as the following *score* (d, Q) by Equation (2).

$$\text{score}(d, Q) = p(y = 1|x) = \frac{e^{w^T x + b}}{1 + e^{w^T x + b}} \quad \text{Equation (2)}$$

where “y = 1” stands for “the document is relevant to the query” and $p(\bullet)$ returns a possibility value, d represents the document, Q represents the query, w is a weight of a feature for the query, x is a feature input vector, T represents transposition, and b represents a bias.

[0048] An exemplary DNN model for calculating a relevance score of a matched document at 210 will be described below in connection with FIG.4.

[0049] In some examples, after training, some features with low weights may be deleted from the feature list for the consideration of model capacity.

[0050] When the original score corresponding to the exemplary matching pattern

[1001] is calculated, the relevance score of the matched document A will be determined based on the original score, for example, the same as the original score, multiple of the original score, and so on. Relevance score of each matched document may be calculated based on respective original score corresponding to its matching pattern.

[0051] Optionally, at 212, a plurality of matched documents with respective relevance score, comprising the document A may be returned or provided to a user in a sequence, as a result of document retrieval. For example, the returned or provided plurality of matched documents may be the top-k matched documents. The returned or provided top-k documents may be ranked based on the calculated relevance scores of the matched documents.

[0052] It should be appreciated that the method 200 may further comprise any other or alternative steps for document retrieval according to the embodiments of the present disclosure as mentioned above.

[0053] FIG.3 illustrates an exemplary inverted index 300 showing an exemplary matching pattern 310 for Document 3 according to an embodiment. The inverted index may show each term in a query and a plurality of documents matched with any term in the query.

[0054] A simple inverted index is shown in FIG.3. Each row represents one term in the query, and each column represents a document indexed by a DocID. In FIG.3, the exemplary query comprises four terms, term 1 (*home*), term 2 (*depot*), term 3 (*wood*), term 4 (*floor*), and for each term in the query, there are several matched documents indicated by DocIDs in the inverted index. For example, for term 1, there are matched documents indicated by Doc3, Doc7, Doc9, Doc13 in the inverted index; for term 2, there are matched documents indicated by Doc1, Doc7, Doc10, Doc11, and so on. From the inverted index in FIG.3, a matching pattern between the query and a document may be obtained. For example, a matching pattern between the query and Doc1 is [0101], a matching pattern between the query and Doc3 is [1001] which is exemplarily highlighted by a dashed box 310 in FIG. 3, a matching pattern between the query and Doc7 is [1110], and so on.

[0055] It should be understood that, the structure of the inverted index 300 shown in FIG.3 is just exemplary; any other structure of the inverted index for document retrieval may be possible for the present disclosure.

[0056] FIG.4 illustrates an exemplary Deep Neural Network (DNN) scoring model 400 for document retrieval according to an embodiment.

[0057] As shown in FIG.4, features of different feature types are inputted into the DNN model 400 in a form of sparse input, such as, features of feature type i , features of feature type j , etc. In an embedding layer, instead of one weight per feature, the DNN model 400 applies a weight vector per feature. For example, a weight vector for the a^{th} feature of the feature type i is $[w_{ia}^1, w_{ia}^2, \dots, w_{ia}^k]$ with k values. In FIG.4, a and b are numbers of features of feature type i and j respectively. Each vector in the embedding layer is passed through a pooling layer to obtain a weight vector for all the features of each feature type. After non-linear transformation of the weight vectors through two hidden layers, e.g., Hidden Layer1 and Hidden Layer2, a score of a document is generated and outputted through an output layer.

[0058] FIG.5 illustrates another exemplary process 500 for document retrieval according to an embodiment. The exemplary process 500 is an optimization way over the process 200 by considering a plurality of potential matching patterns for a given query and pre-calculating a plurality of scores for the plurality of potential matching patterns, which may be performed before, at the same time of, or after operations of searching in a database to obtain matched documents, classifying each term in the query and/or obtaining matching patterns between the query and the matched documents. In some examples, the plurality of potential matching patterns for a given query may comprise all possible or potential matching patterns for the given query. In some other examples, the plurality of potential matching patterns for a given query may comprise a number of potential matching patterns less than all potential matching patterns.

[0059] At 502, a query Q may be received, for example, from a user. The query may be a search query and comprise one or more terms. As an example, a query “*home depot wood floor*” may be received herein. The exemplary terms contained in the query may be “*home*”, “*depot*”, “*wood*” and “*floor*”.

[0060] At 504, a plurality of potential matching patterns of the query Q may be determined. For example, if there are n terms in the query Q , each term can potentially be matched or unmatched with a document indicated by a DocID in the inverted index. Therefore, up to 2^n potential matching patterns of the query Q , which may be all possible or potential matching patterns, may be determined. For example,

given a query including three terms, this query may have 2^3 potential matching patterns, including [000], [001], [010], [100], [011], [101], [110], [111]. Since the matching pattern with all zero, such as [000] indicates that there is no document matched with the query, such matching pattern may be not considered in scoring document for document retrieval. In this case, the number of potential matching patterns whose score to be calculated may be less than all potential matching patterns of a given query, for example, 2^n-1 potential matching patterns excluding the pattern with all zero.

[0061] At 506, for each of the plurality of potential matching patterns determined at 504, an original score or an upperbound score may be obtained.

[0062] An original score corresponding to each potential matching pattern may be obtained through making calculation based on weights of a plurality of features associated with the query, which is similar to the above original score calculation. Therefore, the detailed description for the original score calculation for each potential matching pattern may be omitted herein for simplicity.

[0063] Herein the upperbound score corresponding to a matching pattern may be a highest score among scores of sub-patterns of the matching pattern, such as a max score among multiple scores. For example, sub-patterns of a particular matching pattern may comprise a set of all matching pattern with at least one matched term in the particular matching pattern. For example, if a matching pattern is [101], then its sub-patterns may be [100], [001], [101]; if a matching pattern is [010], then its sub-pattern may be [010]; if a matching pattern is [111], then its sub-patterns may be [001], [010], [011], [100], [101], [110], [111]. The upperbound score for a matching pattern may be utilized to skip less related documents from a plurality of matched documents, to improve retrieval performance. The obtaining of upperbound score corresponding to the matching pattern will be described below.

Upperbound score obtaining

[0064] Given a query $Q = [t_1, t_2, \dots, t_m]$ with m distinct terms, the original score for each potential matching pattern may be calculated, and an upperbound score for a particular potential matching pattern may be derived or obtained according to the original scores of sub-patterns of the particular potential matching pattern.

[0065] If a query has 3 terms, {a, b, c}, then an upperbound score for each potential matching pattern may be obtained as follow:

$\text{upperbound}(\{a, b, c\})$
 $= \text{maxScore}(\{1/0, 1/0, 1/0\})$
 $= \max(\text{maxScore}(\{1/0, 1/0, 0\}), \text{maxScore}(\{0, 1/0, 1/0\}), \text{maxScore}(\{1/0, 0, 1/0\}), \text{score}(\{1, 1, 1\}))$
 $= \max(\text{upperbound}(\{a, b\}), \text{upperbound}(\{b, c\}), \text{upperbound}(\{a, c\}), \text{score}(\{1, 1, 1\}))$

[0066] Table 2 lists a plurality of potential matching patterns for an exemplary query including 3 terms and calculated original scores and upperbound scores for each potential matching pattern.

Matching Pattern	Original Score	Upperbound Score
001	2	2
010	3	3
011	7	7
100	6	6
101	5	6
110	8	8
111	9	9

Table 2

It should be understood that the specific original scores and upperbound scores for matching patterns shown in Table 2 are just exemplary, any other possible scores may be calculated based on different weights of different features for different queries.

[0067] Each upperbound score of the listed potential matching patterns may be obtained as follow:

for potential matching pattern [001], $\text{upperbound}([001]) = \max\{\text{original score}([001])\} = 2;$

for potential matching pattern [010], $\text{upperbound}([010]) = \max\{\text{original score}([010])\} = 3;$

for potential matching pattern [011], $\text{upperbound}([011]) = \max\{\text{original score}([001]), \text{original score}([010]), \text{original score}([011])\} = 7;$

for potential matching pattern [100], $\text{upperbound}([100]) = \max\{\text{original score}([100])\} = 6;$

for potential matching pattern [101], $\text{upperbound}([101]) = \max\{\text{original score}([001]), \text{original score}([100]), \text{original score}([101])\} = 6;$

for potential matching pattern [110], $\text{upperbound}([110]) = \max\{\text{original score}([100]),$

original score([010]), original score([110])}=8;

for potential matching pattern [111], upperbound([111])=max{original score([001]), original score([010]), original score([100]) , original score([011]) , original score([101]) , original score([110]) , original score([111])}=9.

[0068] From the above obtained upperbound score and Table 2, although the original score “5” of matching pattern [101] (Term 1 & 3 are matched) is lower than the original score “6” of its sub-pattern [100] (Only Term 1 is matched), the upperbound score of pattern [101] may be equal to a highest original score “6” among the scores of its sub-patterns.

[0069] As the documents without any matched term (that is, the matching patterns between the query and such documents are “0...00”) may be not retrieved or less useful in document retrieval, there is no need to calculate their scores.

[0070] The obtained original score or upperbound score for each potential matching pattern of the query may be fed to operation 514 to determine relevance scores of matched documents.

[0071] At 508, a database may be searched to obtain matched documents, which is similar to operation 204 in FIG.2. Therefore, the detailed description of the operation 508 may be omitted herein for simplicity.

[0072] At 510, each term in the query may be classified in accordance with whether the term is contained in the document, which is similar to operation 206 in FIG.2. Therefore, the detailed description of the operation 510 may be omitted herein for simplicity.

[0073] At 512, respective matching pattern between each matched document and the query may be obtained, which is similar to operation 208 in FIG.2. Therefore, the detailed description of the operation 512 may be omitted herein for simplicity.

[0074] At 514, respective relevance score of each matched document may be determined with respect to the query, based on the original score or upperbound score corresponding to each potential matching pattern obtained at 506 and the matching pattern between the document and the query obtained at 512. In some examples, a score corresponding to a potential matching pattern identical to the matching pattern of the matched document obtained at 512 may be selected from the plurality of original scores or upperbound scores obtained at 506, and the relevance score of the matched document may be determined according to the selected score.

[0075] Optionally, at 516, a plurality of matched documents with respective relevance score, such as top-k documents, may be returned or provided to a user in a sequence, as a result of document retrieval, which is similar to step 212 in FIG.2. Therefore, the detailed description of the step 516 may be omitted herein for simplicity.

[0076] It should be understood that, although the operations 504, 506 are shown being performed in parallel with the operations 508, 510, 512 in FIG.5, they may be performed before, at the same time of or after the operations 508, 510 and 512.

[0077] FIG.6(A) illustrates an exemplary inverted index 610 before cursors moving according to an embodiment, and FIG.6(B) illustrates another exemplary inverted index 620 after cursors moving by using upperbound score according to an embodiment.

[0078] The usage of upperbound score may bring retrieval optimization in the inverted index. For the m distinct terms in a query Q, there may be m corresponding posting lists in the inverted index, including the empty ones.

[0079] Taking a query including three terms as an example, there may be three posting lists with cursors, as shown in FIG.6(A). In a retrieval process based on the upperbound score, a threshold may be used to determine a “pivot” document. Herein, the pivot document may represent a document which has a score above a threshold. In this example, a threshold θ for determining the “pivot” document may be assumed as $\theta=8$. A pivot document in a plurality of matched documents associated with the query may be determined through ranking the posting lists based on sequence of DocIDs to which cursors pointed currently. For example, in FIG.6(A), cursor 1 for term 1 in the query is pointed to <4> currently, cursor 2 for term 2 in the query is pointed to <3> currently, and cursor 3 for term 3 is pointed to <7> currently. Based on respective term and corresponding DocID to which the cursors pointed, ranked posting lists may be shown below as in Table 3.

Posting list (p) ID	1	2	3
TermID	Term2	Term1	Term3
DocID	<3>	<4>	<7>

Table 3

[0080] According to the order in Table 3 in combination with the upperbound score in Table 2, the pivot document may be determined below step by step:

- For $p = 1$, Upperbound $([0, 1, 0]) = 3$, which does not satisfy $\text{score} > \theta$ ($\theta=8$), so the retrieval process goes to the next posting list $p = 2$;
- For $p = 2$, Upperbound $([1, 1, 0]) = 8$, which does not satisfy $\text{score} > \theta$ ($\theta=8$), so the retrieval process goes to the next posting list $p = 3$;
- For $p = 3$, Upperbound $([1, 1, 1]) = 9$, which is above θ ($\theta=8$), so the retrieval process returns DocID “<7>” to indicate the pivot document. Herein DocID “<7>” may be considered as a pivot DocID.

[0081] When the pivot DocID is determined, which is <7> for Term 3 as shown in FIG. 6(A), the cursors of Term1 and Term2 (cursor1 and cursor2) may be moved to a first DocID whose number is more than or equal to <7> respectively, to fully evaluate the score of the document.

[0082] FIG. 6(B) shows the status after cursors moving, wherein cursor1 points to <10> and cursor2 points to <7>. Thus, the matching pattern for document <7> is “011” and the score of document <7> is 7 based on the matching pattern “011”, which is less than the threshold 8. So this document <7> may be skipped and cursor2 and cursor3 may be moved to the respective next DocID. The posting lists may be ranked based on current DocIDs again to find the next pivot DocID until all the posting lists have moved to the end or the next pivot cannot be found due to each of the rest upperbound scores is less than or equal to the threshold θ .

[0083] Algorithm 1 below shows pseudo code on how to calculate upperbound score of a matching pattern as described above in a recursive way.

Algorithm 1 CalculateUpperBound algorithm

Require: *postingListSet*: A pattern with each element initialed as “true” or “false”;

PatternLen: Returns the number of “true” values in *postingListSet*;

if *patternIndice* in *upperBoundCache* then

Return *upperBoundCache* [*patternIndice*]

end if

upperBound = *scoreCache* [*patternIndice*];

for $i = 0; i < \text{PatternLen}(\text{postingListSet}); ++i$ do

if *postingListSet* [*i*] then

postingListSet .set (*i*, false);

subUpperBound = CalculateUpperBound (*postingListSet*);

if *subUpperBound* > *upperBound* then

```
        upperBound = subUpperBound ;  
    end if  
    postingListSet .set (i, true);  
    end if  
end for  
upperBoundCache [patternIndice ] = upperBound;  
Return upperBound
```

[0084] FIG.7 illustrates a flowchart of an exemplary method 700 for scoring documents in document retrieval according to an embodiment.

[0085] At 710, a query is received, in which the query comprises one or more terms.

[0086] At 720, a database may be searched to obtain a document containing at least one term of the one or more terms.

[0087] At 730, each term of the one or more terms is classified in accordance with whether the term is contained in the document. The each classified term may be a matched term or an unmatched term.

[0088] At 740, a matching pattern between the query and the document is obtained. The matching pattern may indicate an integration of the each classified term.

[0089] At 750, a relevance score of the document with respect to the query is determined based at least on the matching pattern.

[0090] In an implementation, the integration identifies each term in the query, classification of the each term and position of the each term in the query.

[0091] In an implementation, the method 700 may further comprise: determining a plurality of potential matching patterns of the query; and obtaining a plurality of scores corresponding to the plurality of potential matching patterns by calculating a score corresponding to each potential matching pattern of the plurality of potential matching patterns.

[0092] In an implementation, the relevance score is determined by selecting, from the plurality of scores, a score corresponding to a potential matching pattern identical to the matching pattern.

[0093] In an implementation, calculating the score corresponding to each potential matching pattern comprises: generating a plurality of features of different types for the potential matching pattern based on the potential matching pattern and the query,

wherein each feature has a weight; and calculating the score corresponding to the potential matching pattern based on weights of the plurality of features.

[0094] In an implementation, calculating the score corresponding to each potential matching pattern comprises: calculating at least one original score corresponding to at least one sub-pattern of the potential matching pattern; and obtaining the score corresponding to the potential matching pattern by selecting a highest original score among the at least one original score.

[0095] In an implementation, the relevance score is determined by calculating a score corresponding to the matching pattern.

[0096] In an implementation, calculating the score corresponding to the matching pattern comprises: generating a plurality of features of different types for the matching pattern based on the matching pattern and the query, wherein each feature has a weight; and calculating the score corresponding to the matching pattern based on weights of the plurality of features.

[0097] In an implementation, the different types comprise at least one of: matched terms in the query, unmatched terms in the query, combination of anyone matched term and anyone unmatched term in the query, combination of anyone matched term and any another matched term in the query, combination of the number of terms and the number of matched terms in the query, continuous matched terms in the query, chunk of matched terms in the query, bigram of anyone matched term and anyone unmatched term in the query, bigram of unmatched terms in the query, combination of anyone matched term in the query and a corresponding matching pattern, and combination of anyone unmatched term in the query and a corresponding matching pattern.

[0098] In an implementation, the weight of each feature is trained based on a set of training data. Each training data may comprise a historical query and a plurality of retrieved historical documents, and each historical document may be labeled with a relevance value.

[0099] In an implementation, the matching pattern is represented by a vector, and each dimension of the vector has a value of zero or one.

[00100] It should be appreciated that the method 700 may further comprise any steps/operations for scoring documents in document retrieval according to the embodiments of the present disclosure as mentioned above.

[00101] FIG.8 illustrates an exemplary apparatus 800 for scoring documents in document retrieval according to an embodiment.

[00102] The apparatus 800 may comprise: a query receiving module 810, for receiving a query, the query comprising one or more terms; a searching module 820, for searching in a database to obtain a document containing at least one term of the one or more terms; a classifying module 830, for classifying each term of the one or more terms in accordance with whether the term is contained in the document, wherein the each classified term is a matched term or an unmatched term; a matching pattern obtaining module 840, for obtaining a matching pattern between the query and the document, the matching pattern indicating an integration of the each classified term; and a score determining module 850, for determining a relevance score of the document with respect to the query based at least on the matching pattern.

[00103] In an implementation, the integration identifies each term in the query, classification of the each term and position information of the each term in the query.

[00104] In an implementation, the matching pattern obtaining module 840 is further for determining a plurality of potential matching patterns of the query, and the score determining module 850 is further for obtaining a plurality of scores corresponding to the plurality of potential matching patterns by calculating a score corresponding to each potential matching pattern of the plurality of potential matching patterns.

[00105] In an implementation, the relevance score is determined by selecting, from the plurality of scores, a score corresponding to a potential matching pattern identical to the matching pattern.

[00106] In an implementation, the score determining module 850 is further for: generating a plurality of features of different types for the potential matching pattern based on the potential matching pattern and the query, wherein each feature has a weight; and calculating the score corresponding to the potential matching pattern based on weights of the plurality of features.

[00107] In an implementation, the different types comprise at least one of: matched terms in the query, unmatched terms in the query, combination of anyone matched term and anyone unmatched term in the query, combination of anyone matched term and any another matched term in the query, combination of the number of terms and the number of matched terms in the query, continuous matched terms in the query,

chunk of matched terms in the query, bigram of anyone matched term and anyone unmatched term in the query, bigram of unmatched terms in the query, combination of anyone matched term in the query and a corresponding matching pattern, and combination of anyone unmatched term in the query and a corresponding matching pattern.

[00108] In an implementation, the score determining module 850 is further for: calculating at least one original score corresponding to at least one sub-pattern of the potential matching pattern; and obtaining the score corresponding to the potential matching pattern by selecting a highest original score among the at least one original score.

[00109] In an implementation, the relevance score is determined by calculating a score corresponding to the matching pattern.

[00110] Moreover, the apparatus 800 may also comprise any other modules configured for scoring documents in document retrieval according to the embodiments of the present disclosure as mentioned above.

[00111] FIG.9 illustrates another exemplary apparatus 900 for scoring documents in document retrieval according to an embodiment.

[00112] The apparatus 900 may comprise at least one processor 910. The apparatus 900 may further comprise a memory 920 that is connected with the processor 910. The memory 920 may store computer-executable instructions that, when executed, cause the processor 910 to perform any operations of the methods for scoring documents in document retrieval according to the embodiments of the present disclosure as mentioned above.

[00113] The embodiments of the present disclosure may be embodied in a non-transitory computer-readable medium. The non-transitory computer-readable medium may comprise instructions that, when executed, cause one or more processors to perform any operations of the methods for assisting psychological cure in automated chatting according to the embodiments of the present disclosure as mentioned above.

[00114] It should be appreciated that all the operations in the methods described above are merely exemplary, and the present disclosure is not limited to any operations in the methods or sequence orders of these operations, and should cover all other equivalents under the same or similar concepts.

[00115] It should also be appreciated that all the modules in the apparatuses

described above may be implemented in various approaches. These modules may be implemented as hardware, software, or a combination thereof. Moreover, any of these modules may be further functionally divided into sub-modules or combined together.

[00116] Processors have been described in connection with various apparatuses and methods. These processors may be implemented using electronic hardware, computer software, or any combination thereof. Whether such processors are implemented as hardware or software will depend upon the particular application and overall design constraints imposed on the system. By way of example, a processor, any portion of a processor, or any combination of processors presented in the present disclosure may be implemented with a microprocessor, microcontroller, digital signal processor (DSP), a field-programmable gate array (FPGA), a programmable logic device (PLD), a state machine, gated logic, discrete hardware circuits, and other suitable processing components configured to perform the various functions described throughout the present disclosure. The functionality of a processor, any portion of a processor, or any combination of processors presented in the present disclosure may be implemented with software being executed by a microprocessor, microcontroller, DSP, or other suitable platform.

[00117] Software shall be construed broadly to mean instructions, instruction sets, code, code segments, program code, programs, subprograms, software modules, applications, software applications, software packages, routines, subroutines, objects, threads of execution, procedures, functions, etc. The software may reside on a computer-readable medium. A computer-readable medium may include, by way of example, memory such as a magnetic storage device (e.g., hard disk, floppy disk, magnetic strip), an optical disk, a smart card, a flash memory device, random access memory (RAM), read only memory (ROM), programmable ROM (PROM), erasable PROM (EPROM), electrically erasable PROM (EEPROM), a register, or a removable disk. Although memory is shown separate from the processors in the various aspects presented throughout the present disclosure, the memory may be internal to the processors (e.g., cache or register).

[00118] The previous description is provided to enable any person skilled in the art to practice the various aspects described herein. Various modifications to these aspects will be readily apparent to those skilled in the art, and the generic principles defined herein may be applied to other aspects. Thus, the claims are not intended to be

limited to the aspects shown herein. All structural and functional equivalents to the elements of the various aspects described throughout the present disclosure that are known or later come to be known to those of ordinary skill in the art are expressly incorporated herein by reference and are intended to be encompassed by the claims.

WHAT IS CLAIMED IS:

1. A method for scoring documents in document retrieval, comprising:
receiving a query, the query comprising one or more terms;
searching in a database to obtain a document containing at least one term of the one or more terms;
classifying each term of the one or more terms in accordance with whether the term is contained in the document, wherein the each classified term is a matched term or an unmatched term;
obtaining a matching pattern between the query and the document, the matching pattern indicating an integration of the each classified term; and
determining a relevance score of the document with respect to the query based at least on the matching pattern.
2. The method of claim 1, wherein the integration identifies each term in the query, classification of the each term and position of the each term in the query.
3. The method of claim 1, further comprising:
determining a plurality of potential matching patterns of the query; and
obtaining a plurality of scores corresponding to the plurality of potential matching patterns by calculating a score corresponding to each potential matching pattern of the plurality of potential matching patterns.
4. The method of claim 3, wherein the relevance score is determined by selecting, from the plurality of scores, a score corresponding to a potential matching pattern identical to the matching pattern.
5. The method of claim 3, wherein calculating the score corresponding to each potential matching pattern comprises:

generating a plurality of features of different types for the potential matching pattern based on the potential matching pattern and the query, wherein each feature has a weight; and

calculating the score corresponding to the potential matching pattern based on weights of the plurality of features.

6. The method of claim 3, wherein calculating the score corresponding to each potential matching pattern comprises:

calculating at least one original score corresponding to at least one sub-pattern of the potential matching pattern; and

obtaining the score corresponding to the potential matching pattern by selecting a highest original score among the at least one original score.

7. The method of claim 1, wherein the relevance score is determined by calculating a score corresponding to the matching pattern.

8. The method of claim 7, wherein calculating the score corresponding to the matching pattern comprises:

generating a plurality of features of different types for the matching pattern based on the matching pattern and the query, wherein each feature has a weight; and

calculating the score corresponding to the matching pattern based on weights of the plurality of features.

9. The method of claim 5 or 8, wherein the different types comprise at least one of:

matched terms in the query,

unmatched terms in the query,

combination of anyone matched term and anyone unmatched term in the query,

combination of anyone matched term and any another matched term in the query,

combination of the number of terms and the number of matched terms in the query,

continuous matched terms in the query,

chunk of matched terms in the query,

bigram of anyone matched term and anyone unmatched term in the query,

bigram of unmatched terms in the query,

combination of anyone matched term in the query and a corresponding matching pattern, and

combination of anyone unmatched term in the query and a corresponding matching pattern.

10. The method of claim 5 or 8, wherein the weight of each feature is trained based on a set of training data, each training data comprising a historical query and a plurality of retrieved historical documents, each historical document being labeled with a relevance value.

11. The method of claim 1, wherein the matching pattern is represented by a vector, and each dimension of the vector has a value of zero or one.

12. An apparatus for scoring documents in document retrieval, comprising:

a query receiving module, for receiving a query, the query comprising one or more terms;

a searching module, for searching in a database to obtain a document containing at least one term of the one or more terms;

a classifying module, for classifying each term of the one or more terms in accordance with whether the term is contained in the document, wherein the each classified term is a matched term or an unmatched term;

a matching pattern obtaining module, for obtaining a matching pattern between the query and the document, the matching pattern indicating an integration of the each classified term; and

a score determining module, for determining a relevance score of the document with respect to the query based at least on the matching pattern.

13. The apparatus of claim 12, wherein the integration identifies each term in the query, classification of the each term and position of the each term in the query.

14. The apparatus of claim 12, wherein the matching pattern obtaining module is further for determining a plurality of potential matching patterns of the query, and the score determining module is further for obtaining a plurality of scores corresponding to the plurality of potential matching patterns by calculating a score corresponding to each potential matching pattern of the plurality of potential matching patterns.

15. The apparatus of claim 14, wherein the relevance score is determined by selecting, from the plurality of scores, a score corresponding to a potential matching pattern identical to the matching pattern.

16. The apparatus of claim 14, wherein the score determining module is further for:

generating a plurality of features of different types for the potential matching pattern based on the potential matching pattern and the query, wherein each feature has a weight; and

calculating the score corresponding to the potential matching pattern based on weights of the plurality of features.

17. The apparatus of claim 16, wherein the different types comprise at least one of:

matched terms in the query,

unmatched terms in the query,

combination of anyone matched term and anyone unmatched term in the query,

combination of anyone matched term and any another matched term in the query,

combination of the number of terms and the number of matched terms in the query,

continuous matched terms in the query,

chunk of matched terms in the query,

bigram of anyone matched term and anyone unmatched term in the query,

bigram of unmatched terms in the query,

combination of anyone matched term in the query and a corresponding matching pattern, and

combination of anyone unmatched term in the query and a corresponding matching pattern.

18. The apparatus of claim 14, wherein the score determining module is further for:

calculating at least one original score corresponding to at least one sub-pattern of the potential matching pattern; and

obtaining the score corresponding to the potential matching pattern by selecting a highest original score among the at least one original score.

19. The apparatus of claim 12, wherein the relevance score is determined by calculating a score corresponding to the matching pattern.

20. An apparatus for scoring documents in document retrieval, comprising:

one or more processors; and

a memory storing computer-executable instructions that, when executed, cause the one or more processors to:

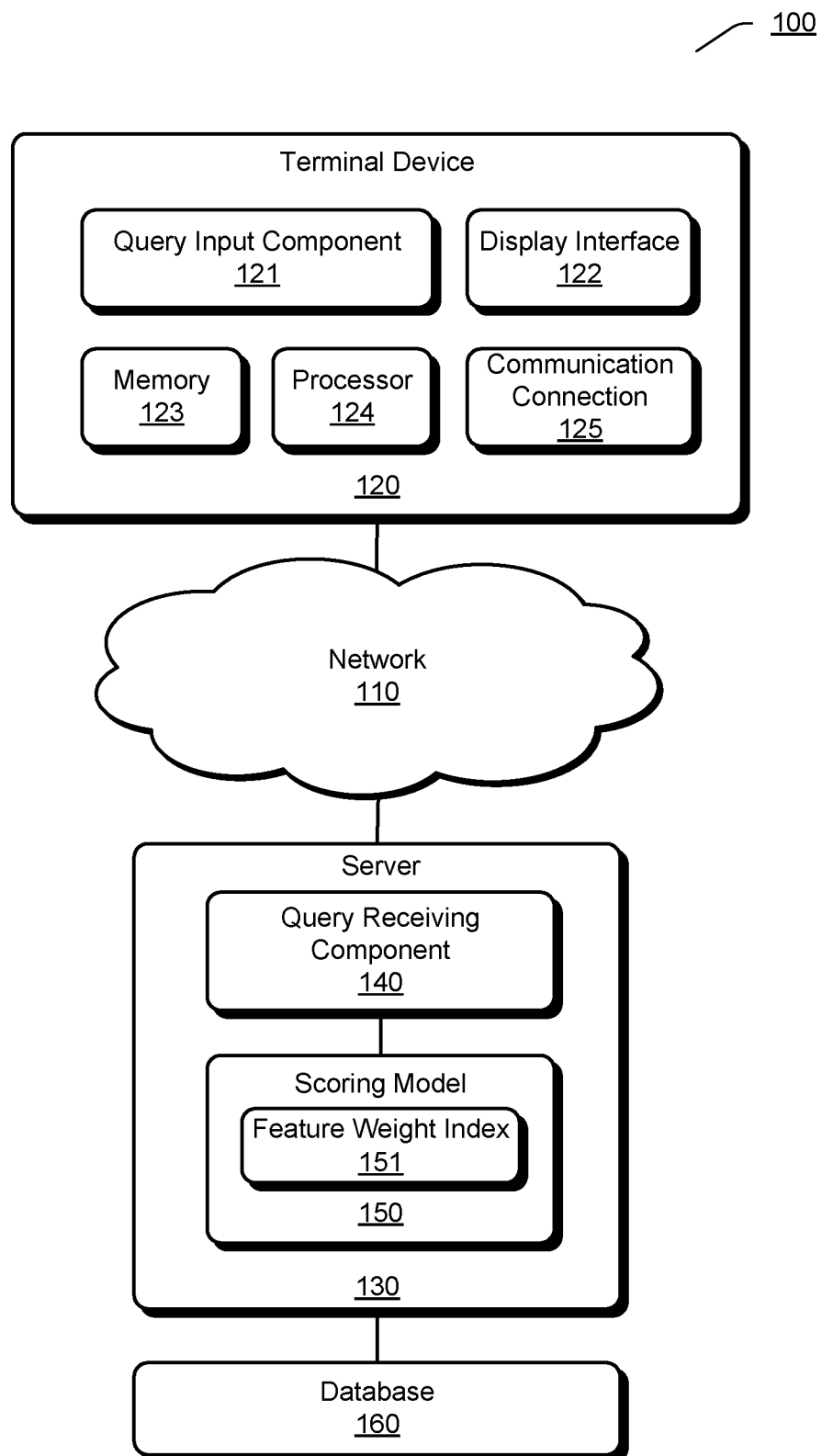
receive a query, the query comprising one or more terms;

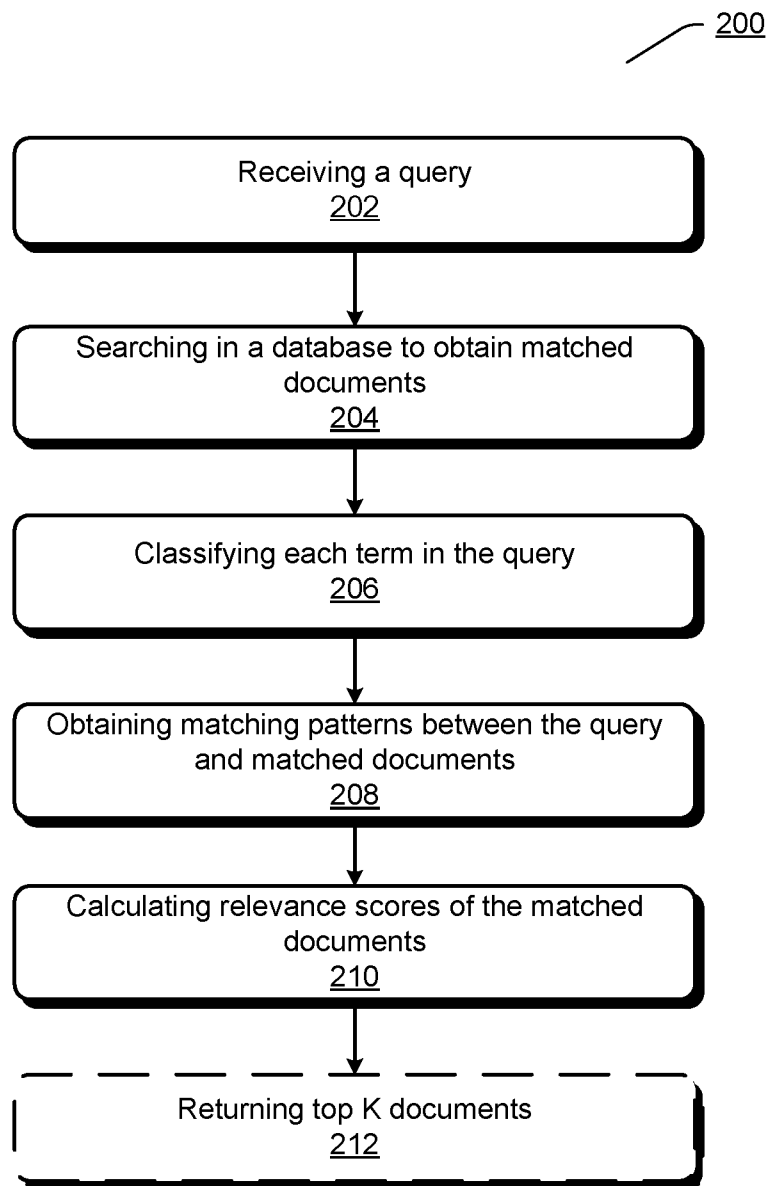
search in a database to obtain a document containing at least one term of the one or more terms;

classify each term of the one or more terms in accordance with whether the term is contained in the document, wherein the each classified term is a matched term or an unmatched term;

obtain a matching pattern between the query and the document, the matching pattern indicating an integration of the each classified term; and

determine a relevance score of the document with respect to the query based at least on the matching pattern.

**FIG 1**

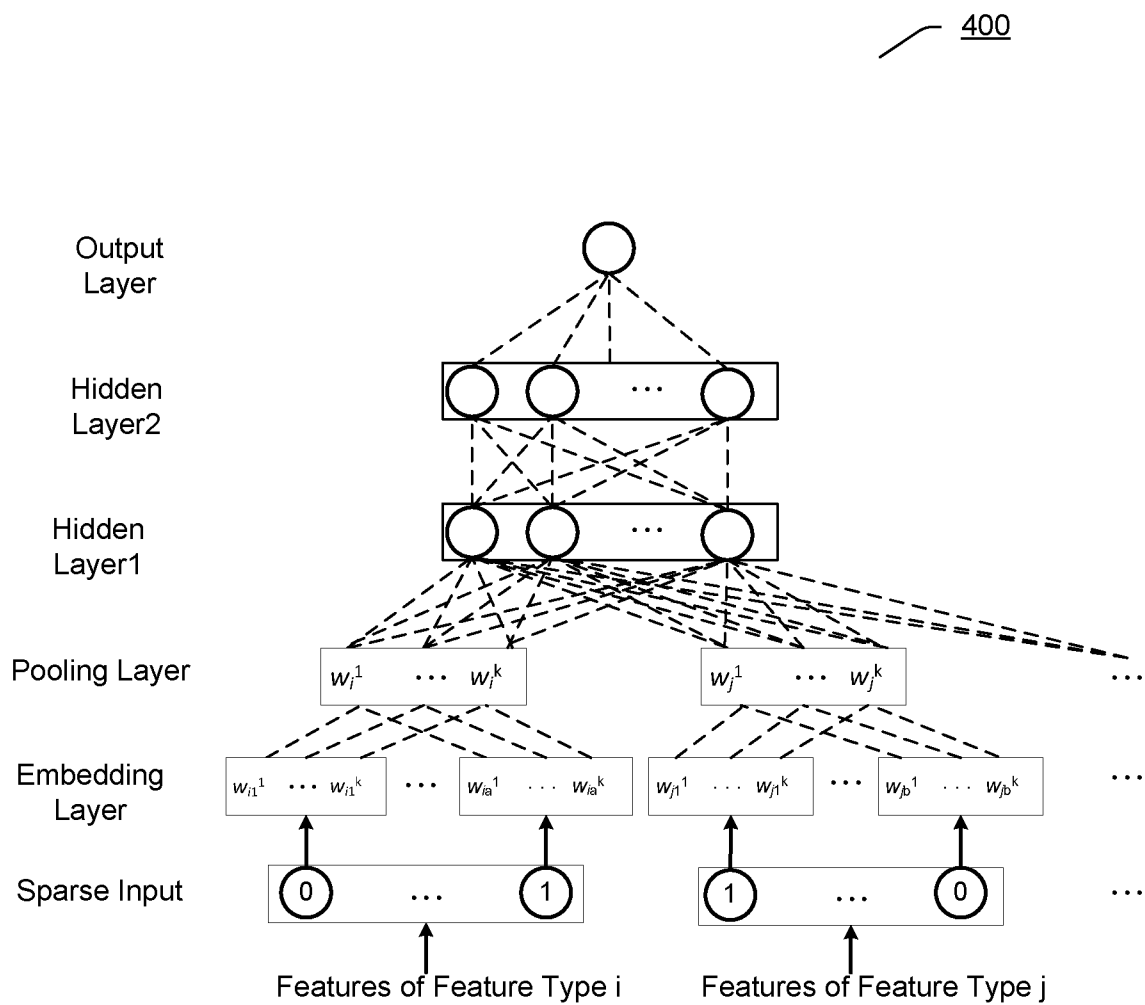
**FIG 2**

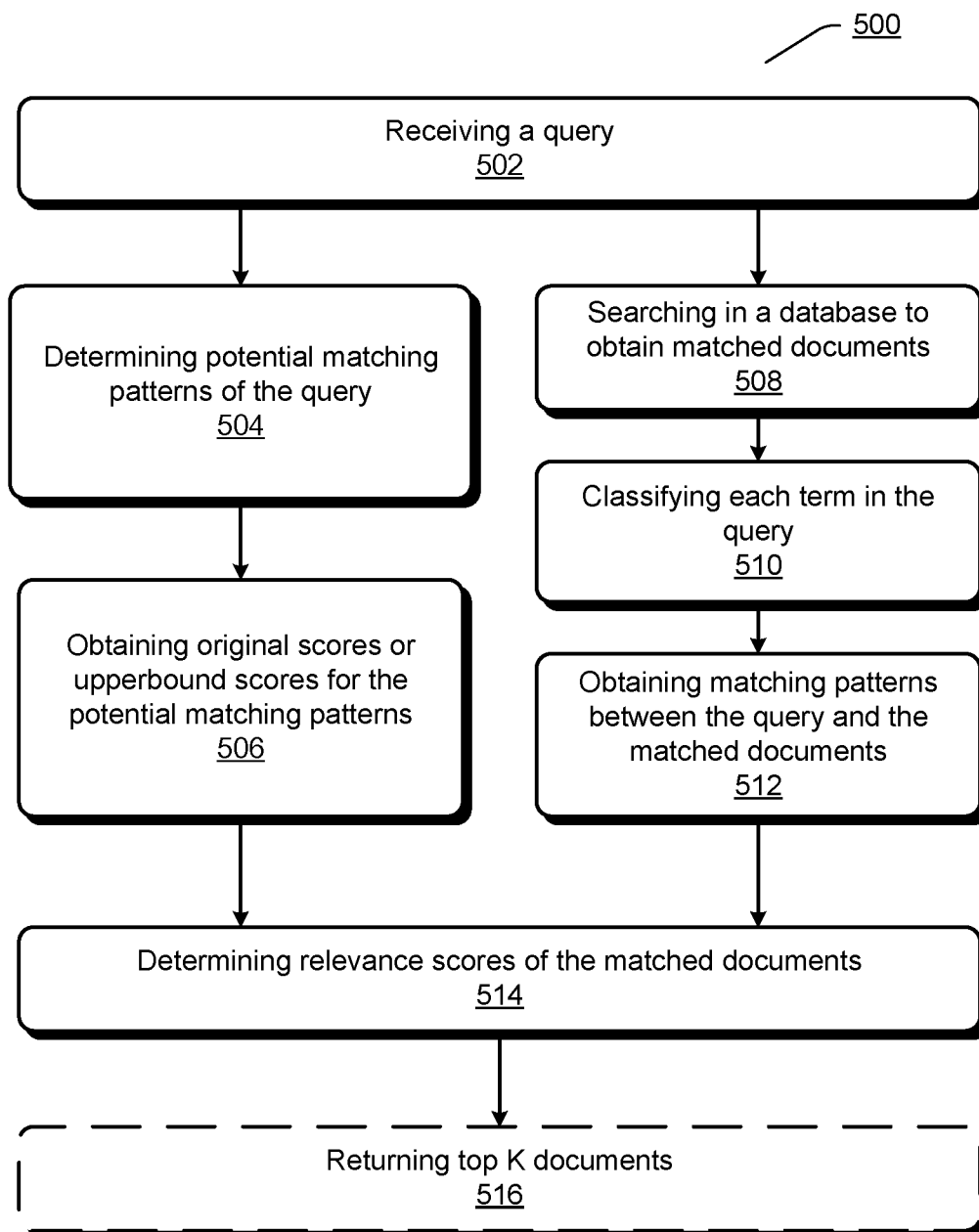
300

	Doc 1	Doc 3	Doc 7	Doc 8	Doc 9	Doc 10	Doc 11	Doc 13
Term1 (home)		√	√		√			√
Term2 (depot)	√		√			√	√	
Term3 (wood)			√			√	√	
Term4 (floor)	√	√		√				

310

FIG 3

**FIG 4**

**FIG 5**

610

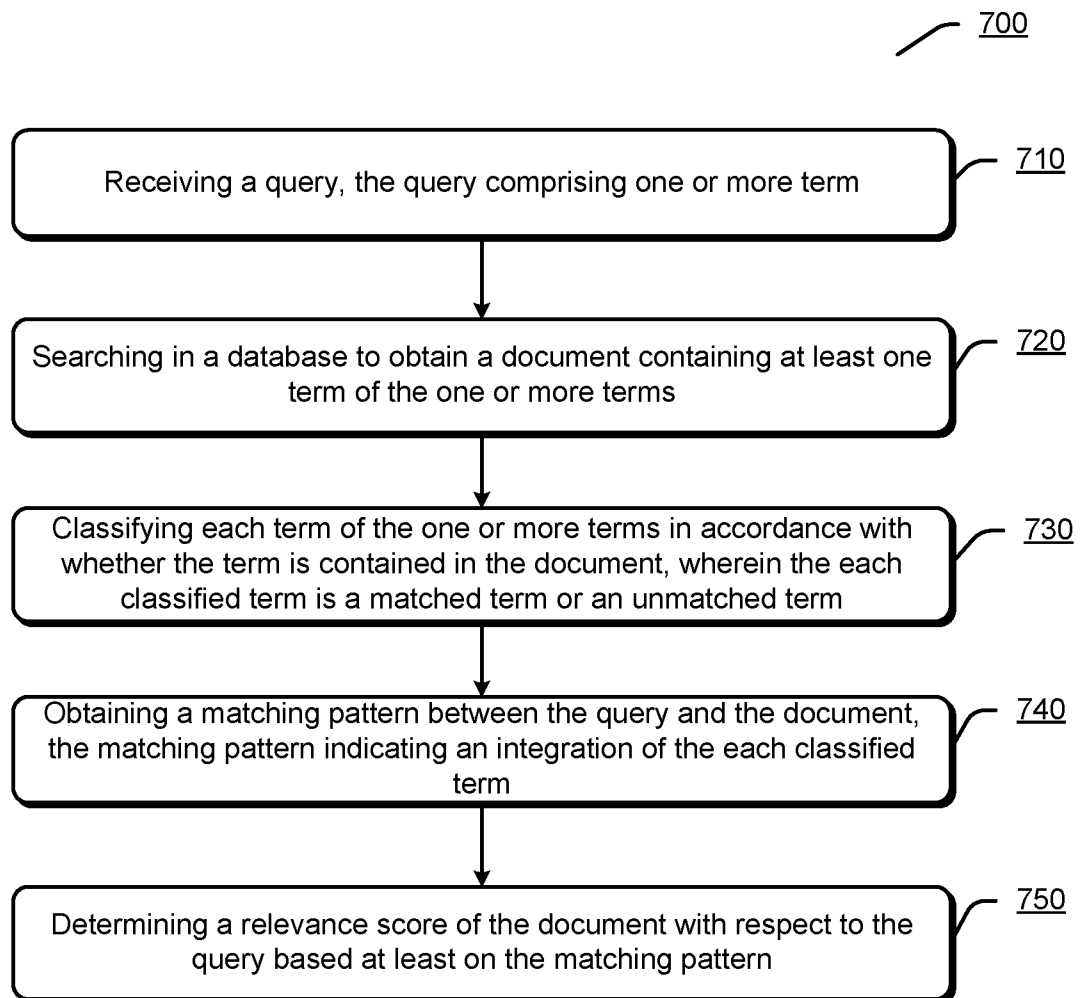
Term in the Query	DocID of Matched Documents				
Term 1	<1>	<4> ← Cursor1	...	<10>	...
Term 2	<1>	<3> ← Cursor2	...	<7>	...
Term 3	<2>	<7> ← Cursor3	...	<9>	...

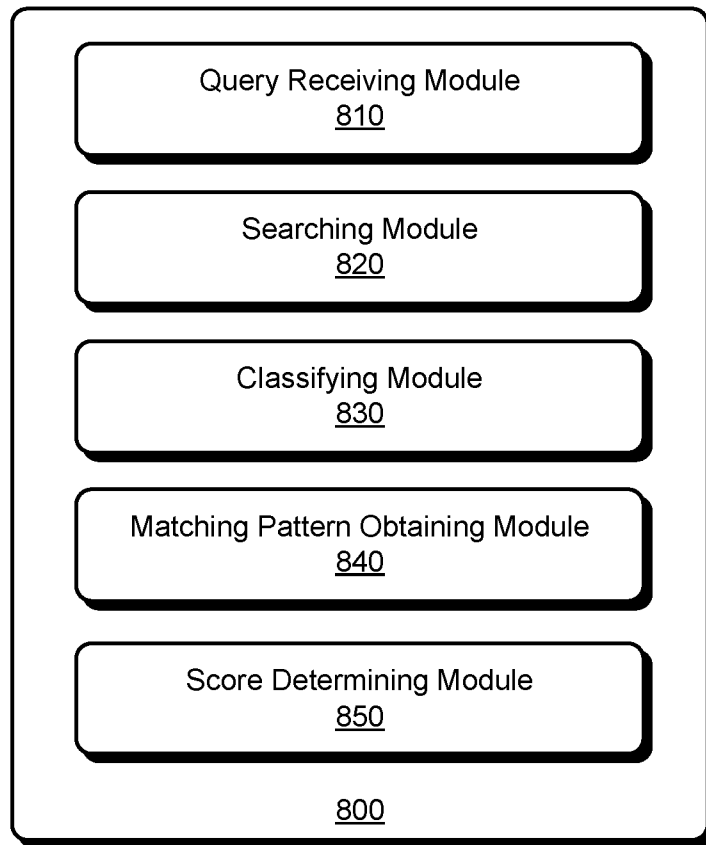
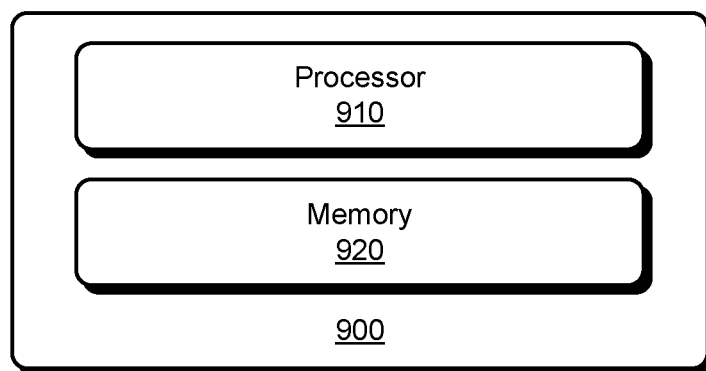
FIG 6(A)

620

Term in the Query	DocID of Matched Documents				
Term 1	<1>	<4>	...	<10> ← Cursor1	...
Term 2	<1>	<3>	...	<7> ← Cursor2	...
Term 3	<2>	<7> ← Cursor3	...	<9>	...

FIG 6(B)

**FIG 7**

**FIG 8****FIG 9**

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2019/073262

A. CLASSIFICATION OF SUBJECT MATTER

G06F 16/00(2019.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNPAT;CNKI;WPI;EPODOC;GOOGLE:information, document, retrieval, search+, quer+, term?, vector, match, score, potential, subset

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	MANNING, Christopher D. et al. "An Introduction to Information Retrieval[online]." <i>Online edition. Cambridge University Press</i> [retrieved on 2019-10-12], 01 April 2009 (2009-04-01), Retrieved from <The Stanford Natural Language Processing Group, https://nlp.stanford.edu/IR-book/pdf/irbookprint.pdf > Chapter 1, 6, pages1-3, 113-114, ISBN: 0521865719	1, 2, 7-13, 19, 20
A	CN 106919565 A (AISINO CORPORATION INC.) 04 July 2017 (2017-07-04) the whole document	1-20
A	US 2017235736 A1 (SWAMINATHAN, ASHWIN ET AL.) 17 August 2017 (2017-08-17) the whole document	1-20
A	CN 105814564 A (MICROSOFT TECHNOLOGY LICENSING, LLC) 27 July 2016 (2016-07-27) the whole document	1-20

☐ Further documents are listed in the continuation of Box C.☒ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

12 October 2019

Date of mailing of the international search report

25 October 2019

Name and mailing address of the ISA/CN

National Intellectual Property Administration, PRC
6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing
100088
China

Authorized officer

LI,Di

Facsimile No. (86-10)62019451

Telephone No. 86-(10)-53961300

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2019/073262

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
CN	106919565	A	04 July 2017	None			
US	2017235736	A1	17 August 2017	None			
CN	105814564	A	27 July 2016	US	2015169582	A1	18 June 2015
				WO	2015088995	A1	18 June 2015
				EP	3080721	A1	19 October 2016