



(51) International Patent Classification:

G06F 17/30 (2006.01)

(21) International Application Number:

PCT/CN2017/082728

(22) International Filing Date:

02 May 2017 (02.05.2017)

(25) Filing Language:

English

(26) Publication Language:

English

(71) Applicant: ALIBABA GROUP HOLDING LIMITED

[—/CN]; Fourth Floor, One Capital Place, P.O. Box 847, George Town, Grand Cayman (KY).

(72) Inventors: ZOU, Youzhi; San Francisco, California (US).

HUANG, Haikuan; Hangzhou, Zhejiang (CN). SI, Luo; Seattle, Washington (US). LIU, Wenqi; Hangzhou, Zhejiang (CN). WU, Chen; Hangzhou, Zhejiang (CN).

(74) Agent: BEIJING TSINGHUAYUAN LAW FIRM; Xue

Yan Building, East Gate of Tsinghua University, Beijing 100084 (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, 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, 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).

Published:

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

(54) Title: METHOD AND APPARATUS FOR QUERY AUTO-COMPLETION

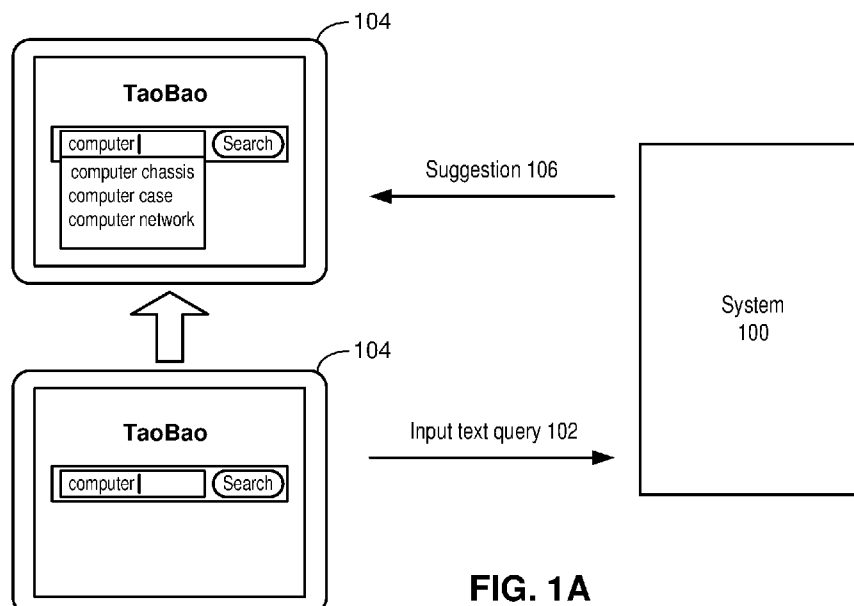


FIG. 1A

(57) **Abstract:** An apparatus for facilitating acquisition of information is described. The apparatus is configured to: receive, acquire a predetermined set of historical input text queries based on an input text query received from a user device; acquire a predetermined set of suggested query candidates; determine similarity measurements between the predetermined set of suggested query candidates and the predetermined set of historical input text queries; determine a ranking of the predetermined set of suggested query candidates based on the similarity measurements; and provide data of at least some of the predetermined set of suggested query candidates and the associated ranking to the user device for displaying.



METHOD AND APPARATUS FOR QUERY AUTO-COMPLETION

TECHNICAL FIELD

[001] The present disclosure generally relates to the field of computer software, and more particularly, to a method and an apparatus for query auto-completion.

BACKGROUND

[002] In an information retrieval system (e.g., a search engine, a content server, etc.), a user transmits a query to the system in order to request certain information. In a case where the query is text-based, the system typically determines a semantic relationship between the text query and the information stored in the system (e.g., a title of a document, a media file, etc.), and provides the stored information based on the semantic relationship. As an illustrative example, a user who wishes to access websites of cosmetic sellers may provide a text query of “cosmetics” to the system. The system may then provide a list of websites that are related to the word “cosmetics,” and predict that the list of websites includes the websites the user is requesting for (or interested in). The accuracy of such a prediction may improve with the specificity of the query. As the query contain more information and become more specific, the system can then do a better job in predicting what information the user actually wants. For example, referring to the example above, in response to the text query of “cosmetics,” the system may provide a list of websites that are related to a particular brand of cosmetics, the user reviews of that brand of cosmetics, etc., but not cosmetics sellers, which may not be what the user intends to look for. In contrast, if the user provides a text query of “cosmetics sellers,” it is more likely that the system will provide a list of websites operated by cosmetic sellers.

[003] Query auto-completion can provide suggestions to help the user include more

information in the query (to “complete the query”), so as to improve the likelihood that the system provides information relevant to the user’s needs. For example, after detecting that the user types in the word “cosmetics,” the system may provide a list of suggested queries such as, for example, “cosmetics sellers in my local area,” “cosmetics sellers specialized in eyeliner,” etc., that adds specificity to the query. The user may then select one of the suggested queries, and provide the selected query to the system to retrieve information. With such arrangements, it becomes more likely that the system can provide information relevant to the user’s needs, which also improve the utilization of the computing and networking resources that support the information retrieval system.

[004] Current technologies provide a number of approaches to generate the suggested query. One approach is to generate the suggested query based on a history of selection of a particular suggested query for a particular input text query. For example, in response to the input text query of “cosmetics,” the system may provide, as a suggested query, “cosmetics sellers in my local area,” based on a determination that this particular suggested query was selected most often when provided in response to the input text query of “cosmetics.” Another approach is to generate the suggested query based on information specific to the user, such as gender, a record of historical actions taken by the user, etc.

[005] The major short-coming of these approaches is that they do not take into account potential complex and entangled correlations in language semantics. If a large number of popular suggested queries is mapped to a small input text query, and/or that user’s record includes lots of texts which are associated with different semantic modes, the lack of consideration of entangled correlations in language semantics makes it difficult for the system to accurately predict the user’s intent based on input text query.

[006] As an illustrative example, the system may map the input text query “cosmetics” to two popular suggested queries “cosmetics sellers in my local area” and “cosmetics sellers specialized in eyeliner.” Both suggested queries have been selected for the same number of times by different users for the input text query of “cosmetics.” Moreover, the record of the user may also reveal that she has watched a movie titled “Eyeliner,” and that she has searched for local sellers of clothing. Now the system receives the input text query of “cosmetics.” In this example, the system may incorrectly disregard the user’s history of searching for local sellers of clothing (for example, because the meaning of the word “clothing” is very different from the word “eyeliner”). The system may also, based on the existence of the word “eyeliner” in the user’s record, incorrectly predicts that the user’s intent is to search for “cosmetics sellers specialized in eyeliner,” when in fact the user wants suggestions for cosmetics sellers in her local area, just as she has searched for local sellers of clothing. As a result, the system provides information that the user does not need, which leads to waste of computing and network resources.

SUMMARY

[007] Embodiments of the present disclosure provide an apparatus for facilitating acquisition of information, the apparatus comprising: a classifier and filter module configured to acquire a predetermined set of historical input text queries based on an input text query received from a user device, and acquire a predetermined set of suggested query candidates; a semantic similarity determination module configured to determine a set of similarity measurements between the predetermined set of suggested query candidates and the predetermined set of historical input text queries; a ranking determination module configured to determine a ranking of the predetermined set of suggested query candidates based on the set of similarity measurements; and a suggestion generation module configured to provide data of at least some

of the predetermined set of suggested query candidates and the associated ranking to the user device for displaying the at least some of the predetermined set of suggested query candidates based on the ranking.

[008] Embodiments of the present disclosure also provide a method of facilitating acquisition of information, the method comprising: acquiring a predetermined set of historical input text queries based on an input text query received from a user device; acquiring a predetermined set of suggested query candidates; determining a set of similarity measurements between the predetermined set of suggested query candidates and the predetermined set of historical input text queries; determining a ranking of the predetermined set of suggested query candidates based on the set of similarity measurements; and transmitting data of at least some of the predetermined set of suggested query candidates and the associated ranking to the user device for displaying the at least some of the predetermined set of suggested query candidates based on the ranking.

[009] Embodiments of the present disclosure also provide a non-transitory computer readable medium that stores a set of instructions that is executable by at least one hardware processor of an apparatus to cause the apparatus to perform a method of facilitating acquisition of information, the method comprising: acquiring a predetermined set of historical input text queries based on an input text query received from a user device; acquiring a predetermined set of suggested query candidates; determining a set of similarity measurements between the predetermined set of suggested query candidates and the predetermined set of historical input text queries; determining a ranking of the predetermined set of suggested query candidates based on the set of similarity measurements; and transmitting data of at least some of the predetermined set of suggested query candidates and the associated ranking to the user device for

displaying the at least some of the predetermined set of suggested query candidates based on the ranking.

[010] Additional objects and advantages of the disclosed embodiments will be set forth in part in the following description, and in part will be apparent from the description, or may be learned by practice of the embodiments. The objects and advantages of the disclosed embodiments may be realized and attained by the elements and combinations set forth in the claims.

[011] It is to be understood that both the foregoing general description and the following detailed description are exemplary and explanatory only and are not restrictive of the disclosed embodiments, as claimed.

BRIEF DESCRIPTION OF THE DRAWINGS

[012] **FIGs. 1A-1E** are block diagrams illustrating an exemplary system for facilitating acquisition of information consistent with embodiments of the present disclosure.

[013] **FIG. 2** is a diagram illustrating an exemplary method of updating a model used for semantic similarity determination consistent with embodiments of the present disclosure.

[014] **FIG. 3** is a block diagram illustrating an exemplary system for facilitating acquisition of information consistent with embodiments of the present disclosure.

[015] **FIG. 4** is a flowchart illustrating an exemplary method of facilitating acquisition of information consistent with embodiments of the present disclosure.

[016] **FIG. 5** is a block diagram illustrating an exemplary computer system on which embodiments described herein can be implemented.

DESCRIPTION OF THE EMBODIMENTS

[017] Reference will now be made in detail to exemplary embodiments, examples of

which are illustrated in the accompanying drawings. The following description refers to the accompanying drawings in which the same numbers in different drawings represent the same or similar elements unless otherwise represented. The implementations set forth in the following description of exemplary embodiments do not represent all implementations consistent with the invention. Instead, they are merely examples of apparatuses and methods consistent with aspects related to the invention as recited in the appended claims.

[018] According to some embodiments, the operations, techniques, and/or components described herein can be implemented by an electronic device, which can include one or more special-purpose computing devices. The special-purpose computing devices can be hard-wired to perform the operations, techniques, and/or components described herein, or can include digital electronic devices such as one or more application-specific integrated circuits (ASICs) or field programmable gate arrays (FPGAs) that are persistently programmed to perform the operations, techniques and/or components described herein, or can include one or more hardware processors programmed to perform such features of the present disclosure pursuant to program instructions in firmware, memory, other storage, or a combination. Such special-purpose computing devices can also combine custom hard-wired logic, ASICs, or FPGAs with custom programming to accomplish the technique and other features of the present disclosure. The special-purpose computing devices can be desktop computer systems, portable computer systems, handheld devices, networking devices, or any other device that incorporates hard-wired and/or program logic to implement the techniques and other features of the present disclosure.

[019] The one or more special-purpose computing devices can be generally controlled and coordinated by operating system software, such as iOS, Android, Blackberry, Chrome OS, Windows XP, Windows Vista, Windows 7, Windows 8, Windows Server, Windows CE, Unix,

Linux, SunOS, Solaris, VxWorks, or other compatible operating systems. In other embodiments, the computing device can be controlled by a proprietary operating system. Operating systems control and schedule computer processes for execution, perform memory management, provide file system, networking, I/O services, and provide a user interface functionality, such as a graphical user interface (“GUI”), among other things.

[020] Embodiments of the present disclosure take into consideration the user’s prior usage of the historical input queries and the suggested query candidate for information acquisition, the determination of the semantic similarities can factor in the intents and preferences of the user. As a result, the determined semantic similarities can provide a better representation of relevancy to that user. Moreover, embodiments of the present disclosure can filter out outlier historical input queries and suggested query candidates that are unrelated to the input query text, the accuracy of semantic similarity (and relevancy) determination can be improved, and the likelihood that the system provides relevant suggestions to the user can also be improved.

[021] Reference is now made to **FIG. 1A**, which illustrates an exemplary system 100 for facilitating acquisition of information, consistent with embodiments of the present disclosure. System 100 can include one or more computer servers that forms a part of a cloud-based data processing platform. As shown in **FIG. 1A**, system 100 may receive an input text query 102 (“computer”) generated by a mobile app operating on a user device 104. Although **FIG. 1A** illustrates that input text query 102 consists of a complete English word (“computer”), it is understood that input text query 102 may include incomplete words as well (e.g., “com”).

[022] System 100 can then provide suggestion 106 based on input text query 102, where the suggestion can include a set of suggested queries system 100 determines to be relevant to

input text query 102. As shown in **FIG. 1A**, suggestion 106 may include, for example, “computer chassis,” “computer case,” and “computer network.” Each of the set of suggested queries may also be associated with a ranking that reflects a predetermined degree of relevancy, and the suggested queries can be displayed in user device 104 based on the rankings. For example, as shown in **FIG. 1A**, the suggested query “computer chassis” is determined to be the most relevant and ranked highest, and is displayed as the first suggestion, followed by “computer case” and “computer network.”

[023] In some embodiments, system 100 may determine the set of suggested queries, and their associated ranking, by determining semantic similarities between a predetermined set of historical input text queries of the user (which may also include input text query 102) and a predetermined set of suggested query candidates, both of which can be determined based on input text query 102 received from user device 104. The mobile app operating on user device 104 can receive a selection 107 of one of the suggested query, and then transmit the selection to system 100, which can then forward suggested query 109 to a search engine 111. Search engine 111 can then generate and provide a set of search result back to user device 104.

[024] System 100 may also employ a model, which may include a deep neural network, a set of natural language processing semantic algorithms, etc., to process the predetermined sets of historical input text queries and suggested query candidates. Based on the processing result, system 100 can determine semantic similarities between the members of the two predetermined sets. The model can be trained based on prior acquisition of information by the user after submitting the historical input text queries, as well as submitting one of the suggested query candidates.

[025] As an illustrative example, the user provided one of the historical input text

queries to system 100 and was provided with a set of suggested queries. The user selected one of the set of suggested queries and provided the selected query to a search engine, which returned a search result including a list of documents. The user then accessed one of the documents. The selection of a document can provide an indication of, for example, that the user considered the selected suggested query and the historical input text query are semantically similar, to the extent that they both lead to a document the user is interested in. On the other hand, the non-selection of some other documents in the search result may also reflect that the user did not treat those documents as relevant to the selected suggested query and the historical input text query. In the training process, various parameters of the model can be updated, based on the selection and non-selection of a set of documents, to cause the updated model to indicate a high degree of semantic similarity between that particular historical input text query and from the selected suggested query.

[026] By taking into consideration the user's prior usage of the historical input queries and the suggested query candidate for information acquisition, the determination of the semantic similarities can factor in the intents and preferences of the user. As a result, the determined semantic similarities can provide a better representation of relevancy to that user. The likelihood that system 100 provides relevant suggestions to the user can also be further improved.

[027] Reference is now made to **FIGs. 1B-1C**, which illustrate the exemplary components of system 100 consistent with embodiments of the present disclosure. As shown in **FIG. 1B**, system 100 includes a classification database 105 and a historical input text query database 108. Historical input text query database 108 may store a set of historical input text queries previously submitted by a user (or multiple users). In one example, as shown in **FIG. 1B**, historical input text query database 108 stores the texts "monitor," "mouse," and "charger." In

another example, as shown in **FIG. 1C**, historical input text query database 108 stores the texts “make-up,” “eyeliner,” and “ear-ring,” and “ornament.” Classification database 105 associates the set of historical input text queries with a set of categories based on a predetermined semantic scheme. In the example of **FIG. 1B**, texts “monitor” and “mouse” are associated with the category “computer,” and the text “charger” is associated with the category “phone.” In the example of **FIG. 1C**, texts “make-up” and “eyeliner” are associated with the category “cosmetics,” and the texts “ear-ring” and “ornament” are associated with the category “accessories.” System 100 also includes a suggested query candidates database 110. In the example of **FIG. 1B**, suggested query candidates database 110 stores a set of suggested query candidates “computer chassis,” “computer case,” and “computer network.” In the example of **FIG. 1C**, suggested query candidates database 110 stores a set of suggested query candidates “perfume bottled table-top,” “jewelry,” and “shirt.” System 100 further includes a classifier and filter module 112, a semantic similarity determination module 116, a ranking determination module 122, a suggestion generation module 124, and a suggested query forwarding module 126.

[028] In general, the word “module,” as used herein, can be a packaged functional hardware unit designed for use with other components (e.g., portions of an integrated circuit) or a part of a program (stored on a computer readable medium) that performs a particular function of related functions. The module can have entry and exit points and can be written in a programming language, such as, for example, Java, Lua, C or C++. A software module can be compiled and linked into an executable program, installed in a dynamic link library, or written in an interpreted programming language such as, for example, BASIC, Perl, or Python. It will be appreciated that software modules can be callable from other modules or from themselves, and/or can be invoked in response to detected events or interrupts. Software modules

configured for execution on computing devices can be provided on a computer readable medium, such as a compact disc, digital video disc, flash drive, magnetic disc, or any other non-transitory medium, or as a digital download (and can be originally stored in a compressed or installable format that requires installation, decompression, or decryption prior to execution). Such software code can be stored, partially or fully, on a memory device of the executing computing device, for execution by the computing device. Software instructions can be embedded in firmware, such as an EPROM. It will be further appreciated that hardware modules can be comprised of connected logic units, such as gates and flip-flops, and/or can be comprised of programmable units, such as programmable gate arrays or processors. The modules or computing device functionality described herein are preferably implemented as software modules, but can be represented in hardware or firmware. Generally, the modules described herein refer to logical modules that can be combined with other modules or divided into sub-modules despite their physical organization or storage.

[029] Classifier and filter module 112 can use the association information stored in classification database 105 to determine an objective semantic relationships of the set of historical input text queries with the current input query text 102 and, based on the objective semantic relationships, determine which of the set of historical input text queries for semantic similarity determination. Referring to the example of **FIG. 1B**, if classifier and filter module 112 determines that input text query 102 includes texts that are associated with the category “computer,” it may determine that only the historical input text queries associated with the same category “computer” in classification database 105 (which include the texts “monitor” and “mouse”) are to be provided for semantic similarity determination, which may cause the system to provide query candidates related to computer. Also, referring to the example of **FIG. 1C**, if

classifier and filter module 112 determines that input text query 102 includes texts that are associated with the category “cosmetics,” it may determine that only the historical input text queries associated with the same category “cosmetics” in classification database 105 (which include the texts “make-up” and “eyeliner”) are to be provided for semantic similarity determination, which may cause the system to provide candidates related to cosmetics (e.g., “perfume bottled table-top” rather than candidates that are not related to cosmetics (e.g., “jewelry.”). Moreover, classifier and filter module 112 may also determine the identity of the user (e.g., based on an identifier of user device 104 from which system 100 receives input text query 102), and filter out historical input text queries that are not generated by the user. Classifier and filter module 112 may also acquire the set of suggested query candidates from suggested query candidates database 110 based on, for example, a determination that each of these candidates share the same prefix (e.g., “com”) as the current input query text 102 (“computer”). Classifier and filter module 112 can then provide the determined set of historical input text queries (which may also include the current input query text 102), as well as suggested query candidates, to semantic similarity determination module 116 for semantic similarity determination.

[030] By filtering out outlier historical input queries and suggested query candidates that are unrelated to the input query text, the accuracy of semantic similarity (and relevancy) determination can be improved, and the likelihood that system 100 provides relevant suggestions to the user can also be improved.

[031] Semantic similarity determination module 116 includes a semantic analytic module 118 and a similarity module 120 for determining semantic similarity between the determined sets of historical input text queries and suggested query candidates. For example,

semantic vector module 118 may determine a set of vectors that represent each of the determined sets of historical input text queries and suggested query candidates in a predetermined semantic space. Similarity module 120 may then provide an indication of semantic similarity based on the set of vectors.

[032] In some embodiments, the semantic space can refer to a multi-dimensional coordinates system that can provide a measurement of a semantic relationship. The semantic space can be defined by a number of reference texts, each of which can represent one dimension in the semantic space. A set of texts (e.g., an input text query, a suggested query candidate, etc.) can be represented by a multi-dimensional vector defined by the semantic space. The vector can include a set of numbers, with each number reflecting a number of occurrences of a reference text in the set of texts.

[033] Reference is now made to **FIG. 1D**, which illustrates an exemplary semantic space consistent with embodiments of the present disclosure. As shown in **FIG. 1D**, table 150 illustrates that a first set of texts includes three occurrences of the reference text “Apple,” zero occurrence of the reference text “phone,” and four occurrences of the reference text “computer.” Also, a second set of texts includes one occurrence of the reference text “Apple,” two occurrences of the reference text “phone,” and three occurrences of the reference text “computer.” Based on these information, the first set of texts can be associated with a vector [3, 0, 4], and the second set of texts can be associated with a vector [1, 2, 3]. Graph 160 provides an illustration of these vectors in the semantic space defined by the reference texts “Apple,” “phone,” and “computer.” Although table 150 and graph 160 illustrates that a semantic space has three dimensions, it is understood that a semantic space typically include more than three dimensions and can be defined by more than three reference texts.

[034] The semantic similarity between the two sets of texts can be determined based on a mathematic relationship between the two vectors. Such a semantic similarity measurement can be based on a premise that that when two sets of texts include same or similar numbers of certain texts, the two sets of texts are likely to be semantically similar to each other. The semantic similarity measurement can be based on, for example, a cosine distance between the two vectors (denoted as y_1 and y_2 below) according to the following exemplary expression:

$$\text{Cosine}([y_1], [y_2]) = \frac{[y_1]^T [y_2]}{\|y_1\| \|y_2\|} \quad (\text{Expression 1})$$

Here, y_1 can be a 1x3 matrix of [3, 0, 4] that represents the first set of texts, and y_2 can be a 1x3 matrix of [1, 2, 3] that represents the second set of texts. $[y_1]^T [y_2]$ can refer to the dot-product between the two matrices. $\|y_1\|$ and $\|y_2\|$ may represent the magnitude of each vector. With Expression 1, a number can be computed to represent the cosine distance between the vectors y_1 and y_2 , which can also provide an indication of semantic similarity between the two sets of texts respectively presented by the two vectors.

[035] Referring back to **FIG. 1B**, semantic vector module 118 may store a set of reference texts that define the dimensions of a semantic space. Semantic vector module 118 may then determine a vector for each of the set of historical input text queries and suggested query candidates provided by classifier and filter module 112 based on, for example, a number of occurrences of each reference text. Semantic vector module 118 can then create a set of vector pairs. Each vector pair may associate a vector representing a historical input text query with a vector representing a suggested query candidate. Similarity module 120 can then determine a semantic similarity measurement (e.g., based on cosine distance according to Expression 1) for each vector pair, and provide the determined semantic similarity measurements to ranking determination module 122.

[036] Ranking determination module 122 can determine a ranking for each of the suggested query candidates based on the semantic similarity measurements. In some embodiments, ranking determination module 122 can determine a ranking score for a particular suggested query candidate based on the semantic similarity measurements of all vector pairs that include that suggested query candidate, and higher values of semantic similarity measurements can lead to a higher ranking score. In some embodiments, the ranking score can also be determined based on a weighted average of the semantic similarity measurements and some other data, such as prior activities of the user (e.g., contents of information previously acquired by the user). In some embodiments, other ranking algorithms, such as logistic regression, support vector machines, or cascaded boosting classifiers can also be used by ranking determination module 122 to determine the ranking.

[037] Suggestion generation module 124 can generate suggestion 106 including at least some of the suggested query candidates and their associated ranking scores. For example, Suggestion generation module 124 may include a predetermined number (e.g., five) of the suggested query candidates associated with the highest ranking scores in suggestion 106, and transmit the suggestion to user device 104 for displaying. Suggestion generation module 124 may also transmit the ranking scores associated with the suggested query candidates, and/or transmit the suggested query candidates following a particular order according to the ranking scores, to allow user device 104 to display the suggested query candidates in a manner than reflects the ranking.

[038] System 100 may further include a suggested query forwarding module 126 configured to receive selection 107 from the user of one of the suggested query candidates included in suggestion 106, and forward suggested query 109 to an information acquisition

system (e.g., search engine 111) to acquire information.

[039] In some embodiments, semantic vector module 118 may also apply a model, which may include a deep neural network, a set of natural language processing semantic algorithms, etc., to generate the aforementioned vectors for representing the predetermined sets of historical input text queries and suggested query candidates. Semantic vector module 118 may store a set of parameters representing the model. System 100 further comprises a training module 130 and a prior user activity database 132. Prior user activity database 132 can store data related to, for example, prior acquisition of information by the user after submitting certain historical input text queries and/or certain suggested query candidates. Training module 130 can update the parameters of the model stored in semantic vector module 118 based on the prior user activity data stored in prior user activity database 132. As discussed above, with such arrangements, the determination of the semantic similarities can factor in the intents and preferences of the user. As a result, the determined semantic similarities can provide a better representation of relevancy to that user. The likelihood that system 100 provides relevant suggestions to the user can also be further improved.

[040] Reference is now made to **FIG. 1E**, which illustrates an exemplary deep neural network model 170 which can be used by semantic vector module 118 for vector generation. As shown in **FIG. 1E**, model 170 includes multiple layers, such as a layer 172, a layer 174, a layer 176, and a layer 178. Each layer includes a predetermined number of elements, which can correspond to a set of reference texts for vector generation. For example, layer 172 includes 300k – 1 million elements, which can correspond to 300k to 1 million reference texts numbered between 300k to 1 million. Also, each of layers 174, 176, and 178 includes 100-500 elements, which can correspond to a 100-500 reference texts.

[041] Each element of a layer (except layer 172) may store a value determined based on values from elements of a preceding layer. For example, as shown in **FIG. 1E**, the value stored in element 174-n of layer 174 can be determined based on the values stored in each element of layer 172, and a weighing matrix $W[174-n]$ and a bias matrix $B[174-n]$ associated with element 174-n. The value stored in element 176-n of layer 176 can also be determined based on the values stored in each element of layer 174, and a weighing matrix $W[176-n]$ and a bias matrix $B[176-n]$ associated with element 176-n. Further, the value stored in element 178-n of layer 178 can also be determined based on the values stored in each element of layer 176, and a weighing matrix $W[178-n]$ and a bias matrix $B[178-n]$ associated with element 178-n. The values stored in each element of layer 172 can be based on a result of projecting the input texts into a semantic space defined by the 300k – 1 million reference texts as depicted in **FIG. 1D**.

[042] Moreover, each element can also be associated with an activation function. An exemplary activation function can be as follows:

$$\phi(x) = \frac{1}{(1+e^{-x})} \quad (\text{Expression 2})$$

[043] The value stored in a subsequent layer element can be determined based on the activation function, the weight-bias pair of a preceding layer element, and the value stored in that preceding layer element. As an illustrative example, the value stored in element 174-n (denoted as E_{174_n}) can be determined based on a weighted average of the values stored in the preceding layer 172 following exemplary expression:

$$E_{174_n} = \phi(\sum_{i=1} (w_{ni} \times E_{172_i} + b_{ni})) \quad (\text{Expression 3})$$

[044] Here, w_{ni} and b_{ni} are the matrix values of $W[174-n]$ and $B[174-n]$, and the weighted sum is evaluated using $\phi(x)$ according to Expression 2. Semantic vector module 118 can execute a set of software instructions according to expressions 2 and 3 to determine the

values stored in each of the layers with historical input text query and a suggested query candidate as inputs. Semantic vector module 118 can then obtain the vectors based on the values stored in layer 178.

[045] The weight and bias matrices of model 170 can be updated by training module 130 based on information about prior acquisition of information by the user after submitting certain historical input text queries and/or certain suggested query candidates, which may allow the determination of the semantic similarities to factor in the intents and preferences of the user. Reference is now made to **FIG. 2**, which illustrates an exemplary method of updating model 170. As shown in **FIG. 2**, a user submitted an input text query “phone,” to system 100, and then selected the suggested query “phone wholesale outlet” provided by system 100. The user then provided the suggested query to a search engine. The search engine returns a list of links, and the user selected one of the links (“John’s phone marketplace”) to acquire more information. System 100 can store this history in prior user activity database 132. For example, system 100 can store a mapping between the input text query “phone” and information related to the selected link (e.g., the title “John’s phone marketplace”) in the database. System 100 may also store a mapping between the selected suggested query “phone wholesale outlet” with the title “John’s phone marketplace” in the database as well.

[046] Based on these mappings stored in prior user activity database 132, training module 130 can determine that when the user types in the input text query “phone,” there is a high likelihood that the user is meant to type in “phone wholesale outlet.” For example, the updating of the weight and bias matrices can be based on a gradient-based optimization algorithm that maximizes the likelihood of model 170 generating an output that leads the user to be provided with a particular link (e.g., the link “John’s phone marketplace”), after receiving as

inputs a particular input text query (“phone”) and a particular suggested query candidate (“phone wholesale outlet”). After executing the optimization algorithm, training module 130 may update the weight and bias matrices such that model 170 can generate two vectors for the text “phone” and “phone wholesale outlet” that have small cosine distance in between (which can indicate high semantic similarity value). Moreover, training module 130 can also update the weight and bias matrices such that suggested query candidates that led to other users making a different selection (e.g., selecting “First marketplace”) have large cosine distance with respect to the text “phone.” In some embodiments, the updating of model 170, as well as the generation of the vectors for the historical input text queries and suggested query candidates, can be performed by system 100 in an off-line process that took place at scheduled maintenance periods of system 100.

[047] Reference is now made to **FIG. 3**, which illustrates an exemplary system 300 for facilitating acquisition of information, consistent with embodiments of the present disclosure. As shown in **FIG. 3**, system 300 includes an online distributed server 302 and an online database 304. Online distributed server 302 includes a category classifier module 306, a similarity determination module 308, and a ranking module 310. Online database 304 stores a vector lookup table 312 and a vector lookup table 314. Vector lookup table 312 can store a mapping between a set of keys and a set of vectors that represent historical input text queries stored in historical input text query database 108. Vector lookup table 314 can store a mapping between a set of keys and a set of vectors that represent suggested query candidates stored in suggested query candidates database 110. The key can also be generated based on, for example, a combination of the vector and additional information associated with the historical input text query/suggested query candidate associated with the vector. The additional information may

include, for example, a category classification, an identity of user associated with the input text query, etc. The key can then be used to retrieve a vector from the online database.

[048] The vectors can be generated by an offline processing system 320 at scheduled maintenance periods of system 300. Offline processing system 320 can include vector modules 322 and 324, which can generate a vector from, respectively, historical input text queries and suggested query candidates. Each of vector modules 322 and 324 may include a set of software routines that represent, for example, model 170 of **FIG. 1D** and semantic vector module 118 of **FIG. 1B**. For example, vector modules 322 and 324 can store the parameters related to the model, which may include, for example, the weighing and bias matrices, the values of each element at each layer, etc., and determine a similarity measure (e.g., cosine distance) between vectors generated by the vector modules. Offline processing system 320 can also perform the functionality of training module 130 and update the parameters of the model based on information from prior user activity database 132.

[049] The online distributed server 302 can receive input text query 102 and generate suggestion 106 including a ranked list of suggested query candidates. Category classifier module 306 can perform similar functionalities as classifier and filter module 112 of **FIG. 1B**, and can determine a category of input text query 102 based on, for example, information stored in classification database 105.

[050] Similarity determination module 308 may determine to obtain a number of pairings for similarity determination. For example, similarity determination module 308 may determine to obtain pairs of vectors representing a single historical text query (which may include input text query 102) of a user and vectors representing suggested query candidates. Similarity determination module 308 may also determine to obtain pairs of vectors representing a

set of historical text queries of a user and vectors representing suggested query candidates.

Similarity determination module 308 may also determine to obtain pairs of vectors representing a set of historical text queries of a user and is specific to a certain topic or category determined based on input text query 102 and vectors representing suggested query candidates. Based on the determined pairing, similarity determination module 308 can generate keys to retrieve the vectors from vector lookup table 312 and 314, and then determine semantic similarity measurement (e.g., based on cosine distance) between the retrieved vectors, in a similar way as similarity module 120 of **FIG. 1B**. The semantic similarity measurements for each pairing comprising the suggested query candidates can be provided to ranking module 310, which can determine a ranking score for each suggested query candidates based on the semantic similarity measurements (as well as other information), and include the suggested query candidates in suggest 106 based on their associated ranking scores. Ranking module 310 may perform similarities similar to those of ranking determination module 122 and suggestion generation module 124 of **FIG. 1B**.

[051] **FIG. 4** is a flowchart representing an exemplary method 400 for facilitating acquisition of information, consistent with embodiments of the present disclosure. It will be readily appreciated that the illustrated procedure can be altered to delete steps or further include additional steps. Method 400 can be performed by a server (e.g., systems 100 of **FIG. 1B**, system 300 of **FIG. 3**, etc.) that communicates with a user device (e.g. user device 104).

[052] After an initial start, the server receives an input text query from a user device, in step 402. The input text query may be generated by a user when operating a mobile app. The input text query may include complete words (e.g., “computer”) or incomplete words (e.g., “com”).

[053] The server can then determine a category associated with the input text query, in step 404. The determination can be based on, for example, a predetermined category of the input text query based on its objective meaning and stored in classification database 105. In step 406, the server can acquire historical input text queries based on the determined category from a first database. For example, after determining that the input text query received in step 402 is associated with the category “computer,” the server can acquire a set of historical input text queries from the first database (e.g., historical input text query database 108) that is associated with that specific category, based on information stored in classification database 105. The server can also filter out historical input text queries based on other criteria, such as historical input text queries that are not generated from the same user device, or not from a particular user. In step 408, the server can acquire a set of suggested query candidates from a second database. The acquisition can be based on, for example, prefix-matching between the set of suggested query candidates and the input text query. In some embodiments, steps 402 to 406 can be performed by, for example, classifier and filter module 112 of **FIG. 1B**.

[054] In step 410, the server can determine a first set of vectors for the set of historical input text queries and a second set of vectors for the set of suggested query candidates. The vectors can be determined by projecting the historical input text queries and the suggested query candidates in a multi-dimensional semantic space defined by a set of reference texts. The determination of the vectors can also be based on a model similar to model 170 of FIG. 1D according to Expressions 2-3, and the model can be trained based on prior acquisition of information by the user after submitting the historical input text queries, as well as submitting one of the suggested query candidates, in a similar manner as depicted in FIG. 2 The model can include, for example, a deep neural network, a set of natural language processing semantic

algorithms, etc. In some embodiments, step 410 can be performed by, for example, semantic vector module 118 of **FIG. 1B**.

[055] In step 412, the server may determine semantic similarity measurements between the first and second sets of vectors. For example, the server may generate pairing between each of the first and second set of vectors, and for each pairing, determine a cosine distance between the pair of vectors, based on Expression 1. The server may then generate a set of semantic similarity measurements based on the cosine distances. In some embodiments, step 412 can be performed by, for example, similarity module 120 of **FIG. 1B**.

[056] In step 414, the server may determine a ranking score for each of the set of suggested query candidates. The determination can be based on, for example, the semantic similarity measurements determined in step 412. The determination can also be based on other factors, such as prior user activities, can may include logistic regression, support vector machines, or cascaded boosting classifiers. In some embodiments, step 414 can be performed by, for example, ranking determination module 122 of **FIG. 1B**.

[057] In step 416, the server may provide a set of suggested query candidates, as well as their associated ranking, back to the user device for displaying, which allows the mobile app to receive a selection of one of the suggested query candidates and provide the selected query to an information acquisition system (e.g., search engine 111 of **FIG. 1A**) to acquire information.

[058] **FIG. 5** is a block diagram of an exemplary computer system 500 with which embodiments described herein can be implemented. Computer system 500 includes a bus 502 or other communication mechanism for communicating information, and one or more hardware processors 504 (denoted as processor 504 for purposes of simplicity) coupled with bus 502 for processing information. Hardware processor 504 can be, for example, one or microprocessors.

[059] Computer system 500 also includes a main memory 506, such as a random access memory (RAM) or other dynamic storage device, coupled to bus 502 for storing information and instructions to be executed by processor 504. Main memory 506 also can be used for storing temporary variables or other intermediate information during execution of instructions to be executed by processor 504. Such instructions, after being stored in non-transitory storage media accessible to processor 504, render computer system 500 into a special-purpose machine that is customized to perform the operations specified in the instructions.

[060] Computer system 500 further includes a read only memory (ROM) 508 or other static storage device coupled to bus 502 for storing static information and instructions for processor 504. A storage device 510, such as a magnetic disk, optical disk, or USB thumb drive (Flash drive), etc., is provided and coupled to bus 502 for storing information and instructions.

[061] Computer system 100 can be coupled via bus 502 to a display 512, such as a cathode ray tube (CRT), an liquid crystal display (LCD), or a touch screen, for displaying information to a computer user. An input device 514, including alphanumeric and other keys, is coupled to bus 502 for communicating information and command selections to processor 504. Another type of user input device is cursor control 516, such as a mouse, a trackball, or cursor direction keys for communicating direction information and command selections to processor 504 and for controlling cursor movement on display 512. The input device typically has two degrees of freedom in two axes, a first axis (for example, x) and a second axis (for example, y), that allows the device to specify positions in a plane. In some embodiments, the same direction information and command selections as cursor control may be implemented via receiving touches on a touch screen without a cursor.

[062] Computing system 500 can include a user interface module to implement a

graphical user interface (GUI) that can be stored in a mass storage device as executable software codes that are executed by the one or more computing devices. This and other modules can include, by way of example, components, such as software components, object-oriented software components, class components and task components, processes, functions, fields, procedures, subroutines, segments of program code, drivers, firmware, microcode, circuitry, data, databases, data structures, tables, arrays, and variables. The modules may include, for example, components of system 100 of FIG. 1B and system 300 of FIG. 3.

[063] Computer system 500 can implement the techniques described herein using customized hard-wired logic, one or more ASICs or FPGAs, firmware and/or program logic which in combination with the computer system causes or programs computer system 500 to be a special-purpose machine. According to some embodiments, the operations, functionalities, and techniques and other features described herein are performed by computer system 500 in response to processor 504 executing one or more sequences of one or more instructions contained in main memory 506. Such instructions can be read into main memory 506 from another storage medium, such as storage device 510. Execution of the sequences of instructions contained in main memory 506 causes processor 504 to perform the method steps (e.g., method 400 of FIG. 4) described herein. In alternative embodiments, hard-wired circuitry can be used in place of or in combination with software instructions.

[064] The term “non-transitory media” as used herein refers to any non-transitory media storing data and/or instructions that cause a machine to operate in a specific fashion. Such non-transitory media can comprise non-volatile media and/or volatile media. Non-volatile media can include, for example, optical or magnetic disks, such as storage device 510. Volatile media can include dynamic memory, such as main memory 506. Non-transitory media include, for

example, a floppy disk, a flexible disk, hard disk, solid state drive, magnetic tape, or any other magnetic data storage medium, a CD-ROM, any other optical data storage medium, any physical medium with patterns of holes, a RAM, a PROM, and EPROM, a FLASH-EPROM, NVRAM, flash memory, register, cache, any other memory chip or cartridge, and networked versions of the same.

[065] Non-transitory media is distinct from, but can be used in conjunction with, transmission media. Transmission media can participate in transferring information between storage media. For example, transmission media can include coaxial cables, copper wire and fiber optics, including the wires that comprise bus 502. Transmission media can also take the form of acoustic or light waves, such as those generated during radio-wave and infra-red data communications.

[066] Various forms of media can be involved in carrying one or more sequences of one or more instructions to processor 504 for execution. For example, the instructions can initially be carried on a magnetic disk or solid state drive of a remote computer. The remote computer can load the instructions into its dynamic memory and send the instructions over a telephone line using a modem. A modem local to computer system 500 can receive the data on the telephone line and use an infra-red transmitter to convert the data to an infra-red signal. An infra-red detector can receive the data carried in the infra-red signal and appropriate circuitry can place the data on bus 502. Bus 502 carries the data to main memory 506, from which processor 504 retrieves and executes the instructions. The instructions received by main memory 506 can optionally be stored on storage device 510 either before or after execution by processor 504.

[067] Computer system 500 can also include a communication interface 518 coupled to bus 502. Communication interface 518 can provide a two-way data communication coupling to a

network link 520 that can be connected to a local network 522. For example, communication interface 518 can be an integrated services digital network (ISDN) card, cable modem, satellite modem, or a modem to provide a data communication connection to a corresponding type of telephone line. As another example, communication interface 518 can be a local area network (LAN) card to provide a data communication connection to a compatible LAN. Wireless links can also be implemented. In any such implementation, communication interface 518 can send and receive electrical, electromagnetic or optical signals that carry digital data streams representing various types of information.

[068] Network link 520 can typically provide data communication through one or more networks to other data devices. For example, network link 520 can provide a connection through local network 522 to a host computer 524 or to data equipment operated by an Internet Service Provider (ISP) 526. ISP 526 in turn can provide data communication services through the world wide packet data communication network now commonly referred to as the “Internet” 528. Local network 522 and Internet 528 both use electrical, electromagnetic or optical signals that carry digital data streams. The signals through the various networks and the signals on network link 520 and through communication interface 518, which carry the digital data to and from computer system 500, can be example forms of transmission media.

[069] Computer system 500 can send messages and receive data, including program code, through the network(s), network link 520 and communication interface 518. In the Internet example, a server 530 can transmit a requested code for an application program through Internet 528, ISP 526, local network 522 and communication interface 518.

[070] The received code can be executed by processor 504 as it is received, and/or stored in storage device 510, or other non-volatile storage for later execution. In some

embodiments, server 530 can provide information for being displayed on a display.

[071] It will be appreciated that the present invention is not limited to the exact construction that has been described above and illustrated in the accompanying drawings, and that various modifications and changes can be made without departing from the scope thereof. It is intended that the scope of the invention should only be limited by the appended claims.

WHAT IS CLAIMED IS:

1. An apparatus for facilitating acquisition of information, the apparatus comprising:

a classifier and filter module configured to:

 acquire a predetermined set of historical input text queries based on an input text query received from a user device, and

 acquire a predetermined set of suggested query candidates;

a semantic similarity determination module configured to determine a set of similarity measurements between the predetermined set of suggested query candidates and the predetermined set of historical input text queries;

a ranking determination module configured to determine a ranking of the predetermined set of suggested query candidates based on the set of similarity measurements; and

a suggestion generation module configured to provide data of at least some of the predetermined set of suggested query candidates and the associated ranking to the user device for displaying the at least some of the predetermined set of suggested query candidates based on the ranking.
2. The apparatus of claim 1, further comprising a suggested query forwarding module configured to:

 receive a selection of one of the at least some of the predetermined set of suggested query candidates; and

provide the selected query candidate to an information acquisition system for acquisition of information.

3. The apparatus of claim 1, wherein the classifier and filter module is further configured to:

determine a category associated with the input text query; and

acquiring the predetermined set of historical input text queries based on the determined category.
4. The apparatus of claim 1, wherein the determination of the set of similarity measurements comprises the semantic similarity determination module being further configured to:

determine, based on a model trained with selected data, a first set of vectors associated with the predetermined set of historical input text queries and a second set of vectors associated with the predetermined set of suggested query candidates, and

determine the set of similarity measurements between the first and second set of vectors.
5. The apparatus of claim 4, wherein the determination of a set of similarity measurements comprises the semantic similarity determination module being configured to:

determine a set of cosine distances between the first and second set of vectors; and

determine a set of similarity measurements based on the set of cosine distances.

6. The apparatus of claim 5, wherein the ranking is determined based on the set of cosines distances.
7. The apparatus of claim 4, wherein the first and second sets of vectors are determined from one or more lookup tables.
8. The apparatus of claim 7, wherein the one or more lookup tables store a result of projecting the historical input text queries and the suggested query candidates in a semantic space using a neural network comprising a set of parameters;

wherein the parameters are updated based on history data related to prior acquisition of information after submitting at least some of the historical input text queries and at least some one of the suggested query candidates.
9. The apparatus of claim 1, wherein the predetermined set of suggested query candidates is determined based on a prefix of the input text query.
10. A method of facilitating acquisition of information, the method comprising:

acquiring a predetermined set of historical input text queries based on an input text query received from a user device;

acquiring a predetermined set of suggested query candidates;

determining a set of similarity measurements between the predetermined set of suggested query candidates and the predetermined set of historical input text queries;

determining a ranking of the predetermined set of suggested query candidates based on the set of similarity measurements; and

providing data of at least some of the predetermined set of suggested query candidates and the associated ranking to the user device for displaying the at least some of the predetermined set of suggested query candidates based on the ranking.

11. The method of claim 10, further comprising:

receiving a selection of one of the at least some of the predetermined set of suggested query candidates; and

providing the selected query candidate to an information acquisition system for acquisition of information.

12. The method of claim 10, further comprising: determining a category associated with the

input text query; wherein the predetermined set of historical input text queries is acquired based on the determined category.

13. The method of claim 10, wherein determining a set of similarity measurements comprises:

determining, based on a model trained with selected data, a first set of vectors associated with the predetermined set of historical input text queries and a second set of vectors associated with the predetermined set of suggested query candidates, and

determining a set of cosine distances between the first set of vectors and the second set of vectors; and

determining a set of similarity measurements based on the set of cosine distances.

14. The method of claim 13, wherein the ranking is determined based on the set of cosines distances.

15. The method of claim 13, wherein the first and second sets of vectors are determined from one or more lookup tables;

wherein the one or more lookup tables store a result of projecting the historical input text queries and the suggested query candidates in a semantic space using a neural network comprising a set of parameters; and

wherein the parameters are updated based on history data related to prior acquisition of information after submitting at least some of the historical input text queries and at least some one of the suggested query candidates.

16. The method of claim 10, wherein the predetermined set of suggested query candidates is determined based on a prefix of the input text query.

17. A non-transitory computer readable medium that stores a set of instructions that is executable by at least one hardware processor of an apparatus to cause the apparatus to perform a method of facilitating acquisition of information, the method comprising:

- acquiring a predetermined set of historical input text queries based on an input text query received from a user device;

- acquiring a predetermined set of suggested query candidates;

- determining a set of similarity measurements between the predetermined set of suggested query candidates and the predetermined set of historical input text queries;

- determining a ranking of the predetermined set of suggested query candidates based on the set of similarity measurements; and

- providing data of at least some of the predetermined set of suggested query candidates and the associated ranking to the user device for displaying the at least some of the predetermined set of suggested query candidates based on the ranking.

18. The non-transitory computer readable medium of claim 17, wherein determining a set of similarity measurements comprises the set of instructions that is executable by at least one hardware processor of an apparatus to cause the apparatus to perform:

determining, based on a model trained with selected data, a first set of vectors associated with the predetermined set of historical input text queries and a second set of vectors associated with the predetermined set of suggested query candidates, and

determining a set of cosine distances between the first set of vectors and the second set of vectors; and

determining a set of similarity measurements based on the set of cosine distances.

19. The non-transitory computer readable medium of claim 17, further storing the set of instructions that is executable by at least one hardware processor of an apparatus to cause the apparatus to perform:

receiving a selection of one of the at least some of the predetermined set of suggested query candidates; and

providing the selected query candidate to an information acquisition system for acquisition of information.

20. The non-transitory computer readable medium of claim 17, further storing the set of instructions that is executable by at least one hardware processor of an apparatus to cause the apparatus to perform:

determining a category associated with the input text query; wherein the predetermined set of historical input text queries is acquired based on the determined category.

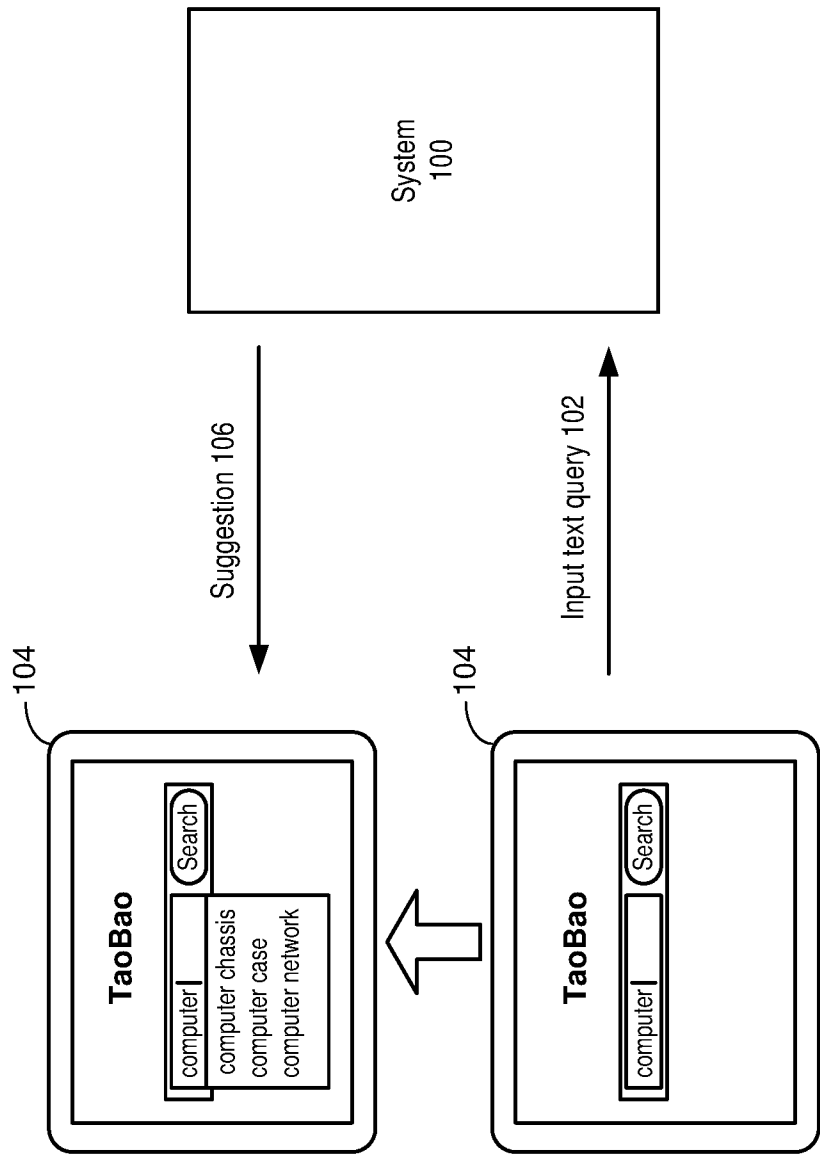


FIG. 1A

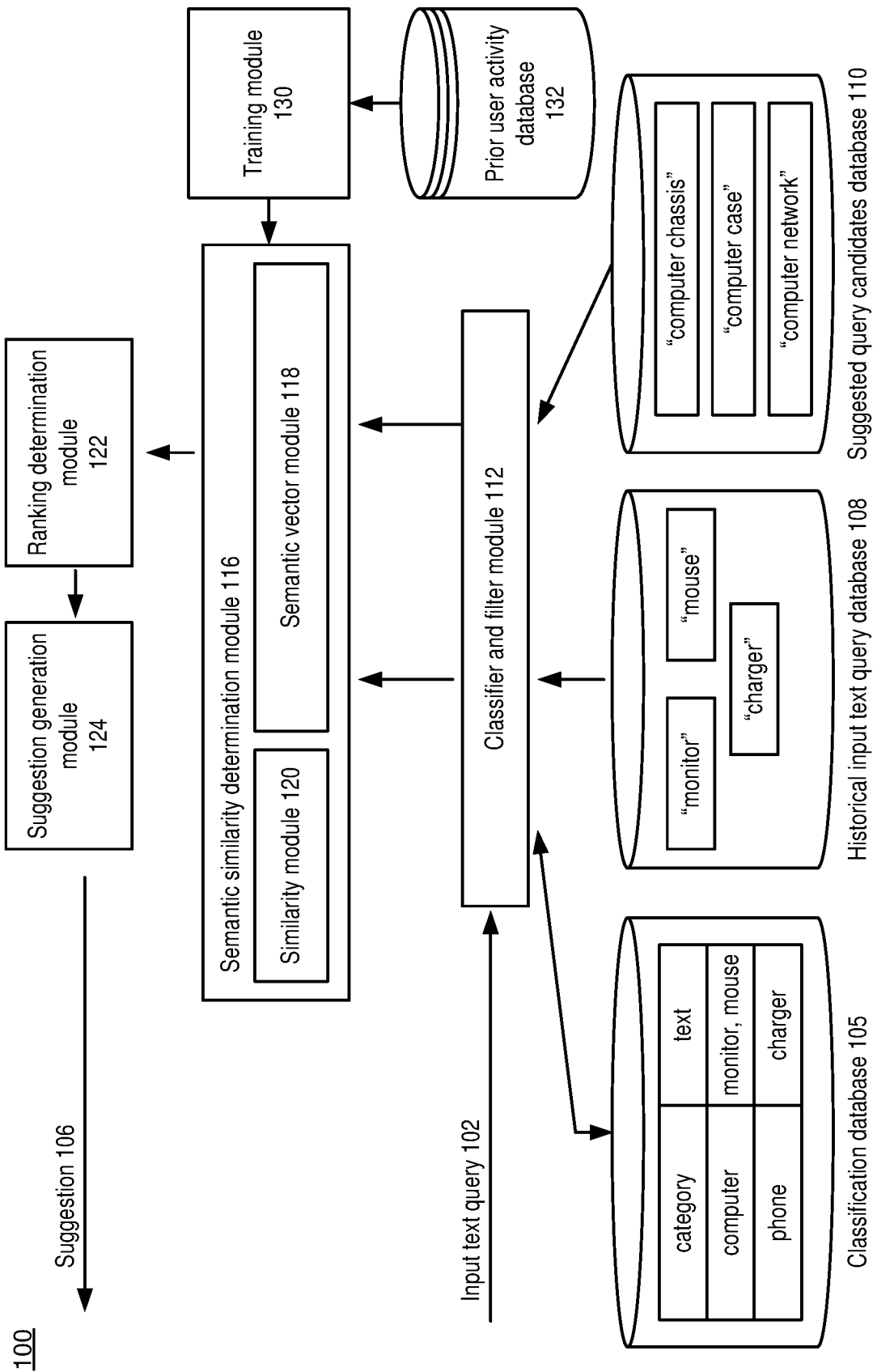


FIG. 1B

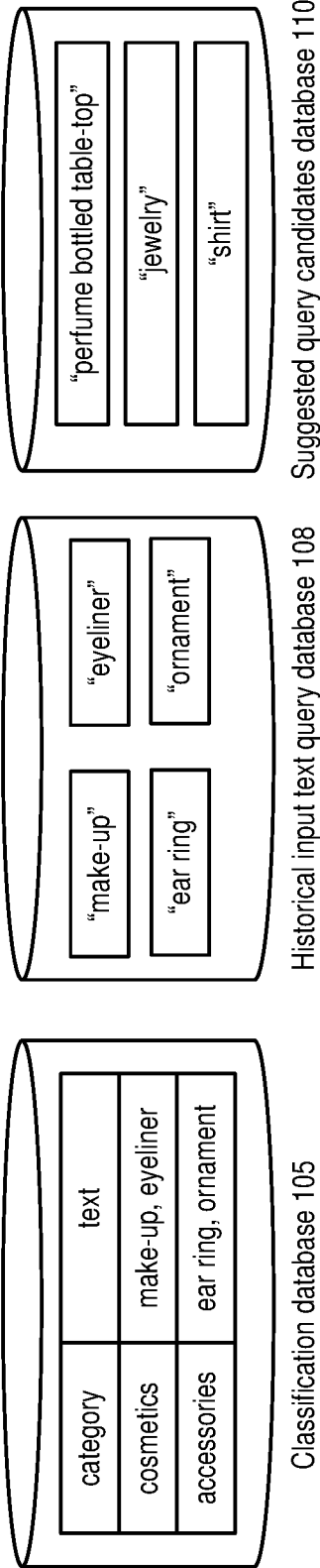
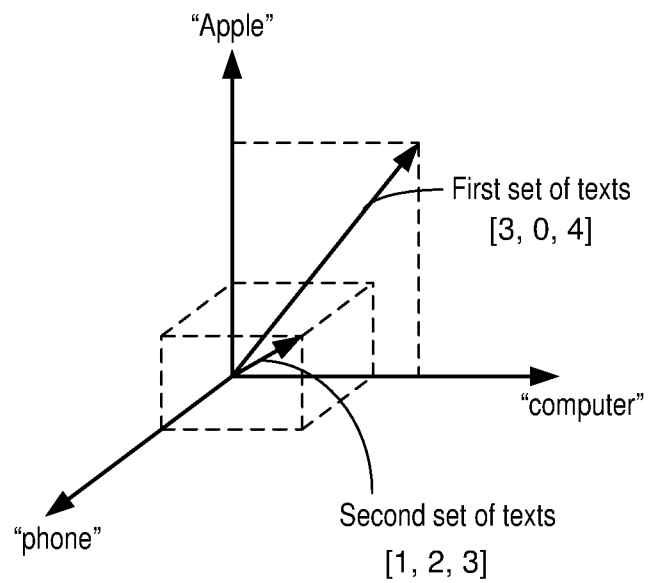


FIG. 1C

150

	"Apple"	"phone"	"computer"
First set of texts	3	0	4
Second set of texts	1	2	3

160**FIG. 1D**

170

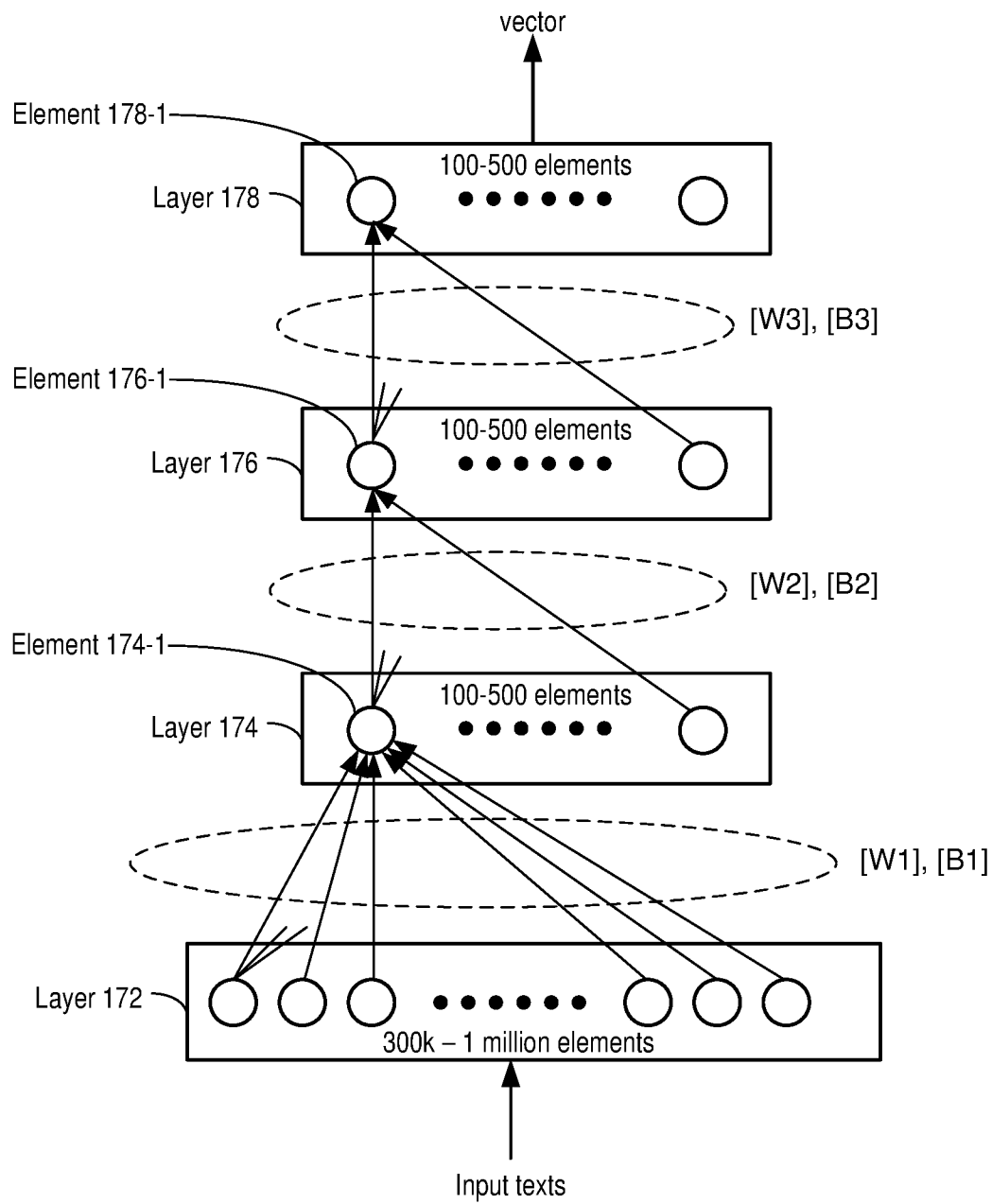


FIG. 1E

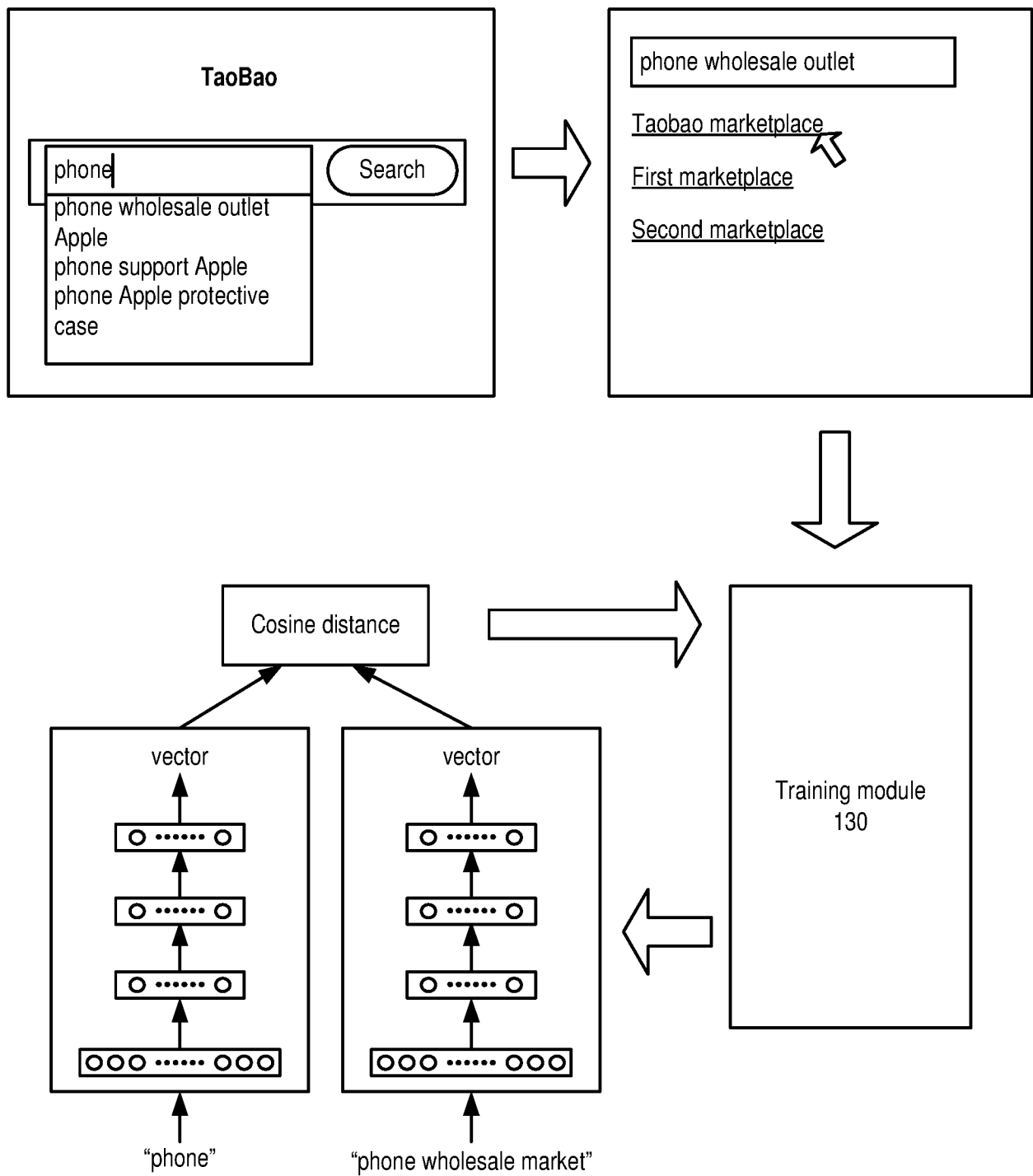


FIG. 2

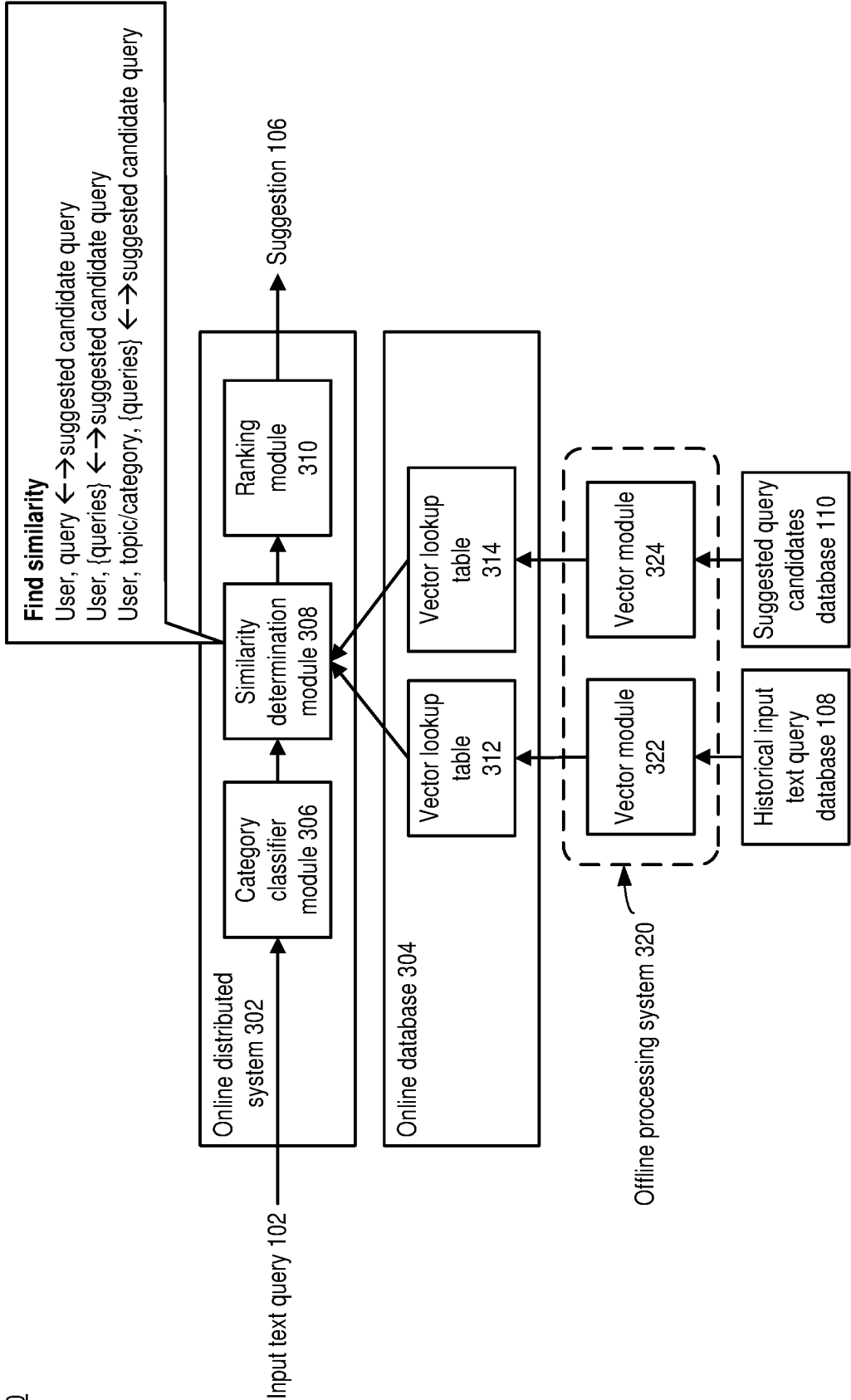
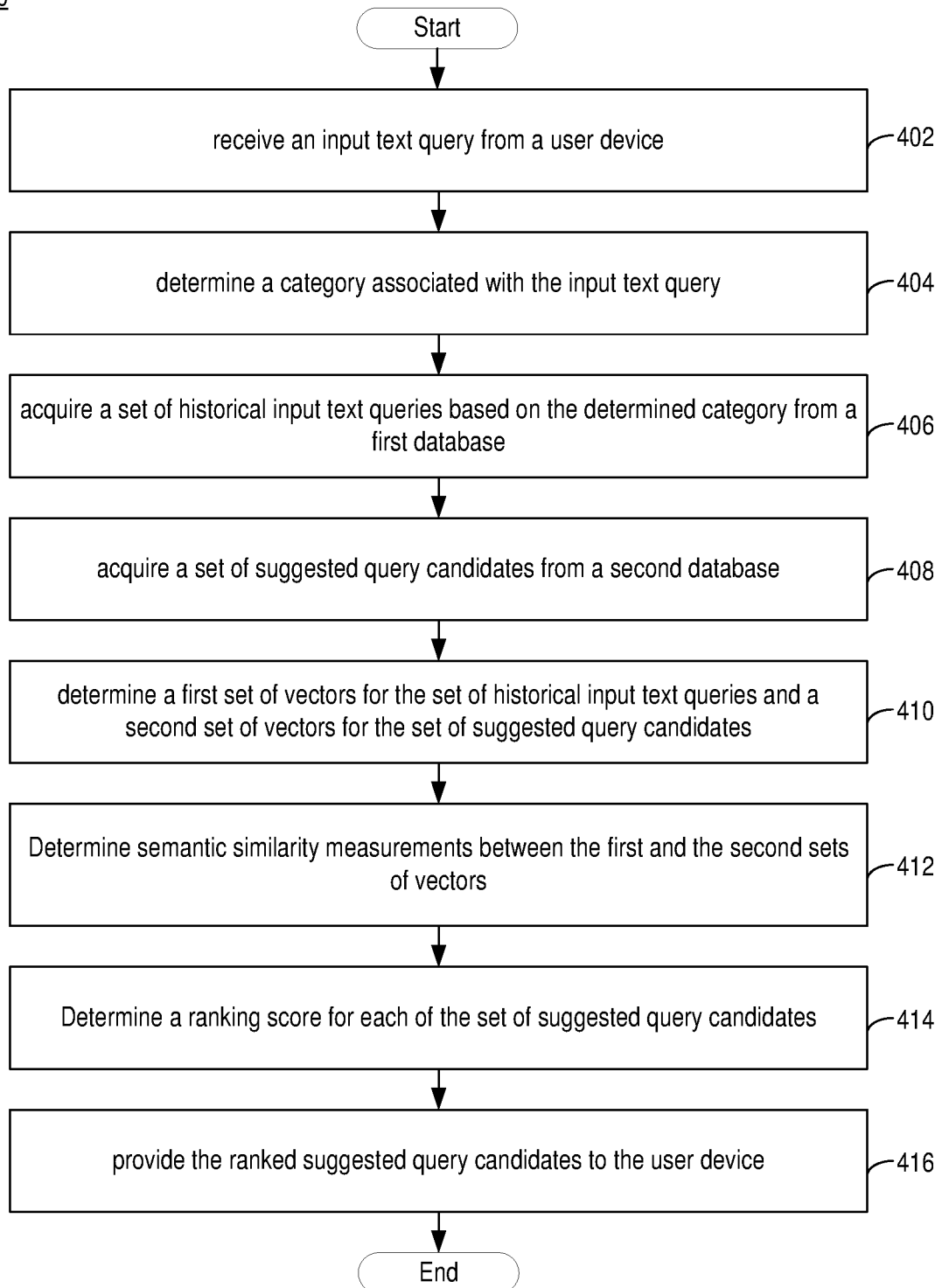


FIG. 3

400

**FIG. 4**

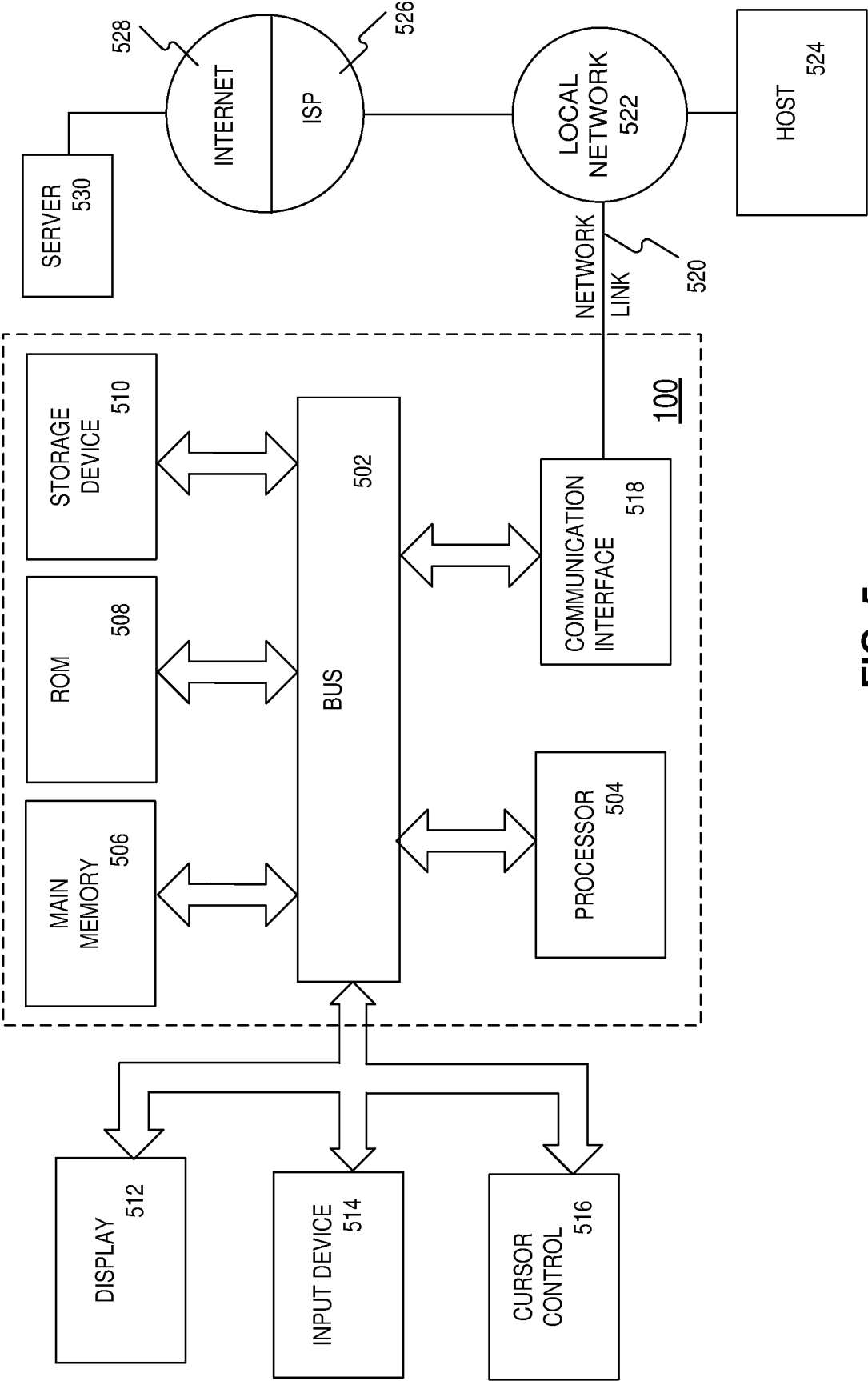


FIG. 5

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2017/082728

A. CLASSIFICATION OF SUBJECT MATTER

G06F 17/30(2006.01)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)

WPI, EPODOC, CNKI, CNPAT, IEEE: query, suggest+, intention, category, historical, text, semantic, similarity, calculat+, rank???, model, train???, vectors, cosine, distance, prefix

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	CN 106599278 A (BEIJING QIHOO TECHNOLOGY CO., LTD.) 26 April 2017 (2017-04-26) description, paragraphs [0006]-[0016], [0096]-[0104], [0149]-[0158], figures 1-3	1-3, 9-12, 16-17, 19-20
Y	CN 106599278 A (BEIJING QIHOO TECHNOLOGY CO., LTD.) 26 April 2017 (2017-04-26) description, paragraphs [0006]-[0016], [0096]-[0104], [0149]-[0158], figures 1-3	4-8, 13-15, 18
Y	CN 105045781 A (GUANGZHOU SHENMA MOBILE INFORMATION TECHNOLOGY CO., LTD.) 11 November 2015 (2015-11-11) description, paragraphs [0008]-[0021], figures 1-14	4-8, 13-15, 18
A	CN 104050235 A (ZHEJIANG UNIVERSITY) 17 September 2014 (2014-09-17) the whole document	1-20
A	US 2008313202 A1 (KAMEN, YAKOV) 18 December 2008 (2008-12-18) 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

19 January 2018

Date of mailing of the international search report

01 February 2018

Name and mailing address of the ISA/CN

**STATE INTELLECTUAL PROPERTY OFFICE OF THE
P.R.CHINA
6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing
100088
China**

Facsimile No. (86-10)62019451

Authorized officer

WANG,Jing

Telephone No. (86-10)53961317

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2017/082728

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
CN	106599278	A	26 April 2017	None			
CN	105045781	A	11 November 2015	WO	2017031996	A1	02 March 2017
CN	104050235	A	17 September 2014	None			
US	2008313202	A1	18 December 2008	None			