(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
25 July 2019 (25.07.2019)

WIPO | PCT

(10) International Publication Number
WO 2019/140863 A1

(51) International Patent Classification:
G06F 17/21 (2006.01)

(21) International Application Number:
PCT/CN2018/091822

(22) International Filing Date:
19 June 2018 (19.06.2018)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
201810060942.7 22 January 2018 (22.01.2018) CN

(71) Applicant: **BOE TECHNOLOGY GROUP CO., LTD.** [CN/CN]; No. 10 Jiuxianqiao Rd., Chaoyang District, Beijing 100015 (CN).

(72) Inventor: **ZHANG, Zhenzhong**; No. 9 Dize Rd., BDA, Beijing 100176 (CN).

(74) Agent: **TEE & HOWE INTELLECTUAL PROPERTY ATTORNEYS**; CHEN, Yuan, 10th Floor, Tower D, Minsheng Financial Center, 28 Jianguomennei Avenue, Dongcheng District, Beijing 100005 (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, 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 OF CALCULATING RELEVANCY, APPARATUS FOR CALCULATING RELEVANCY, DATA QUERY APPARATUS, AND NON-TRANSITORY COMPUTER-READABLE STORAGE MEDIUM

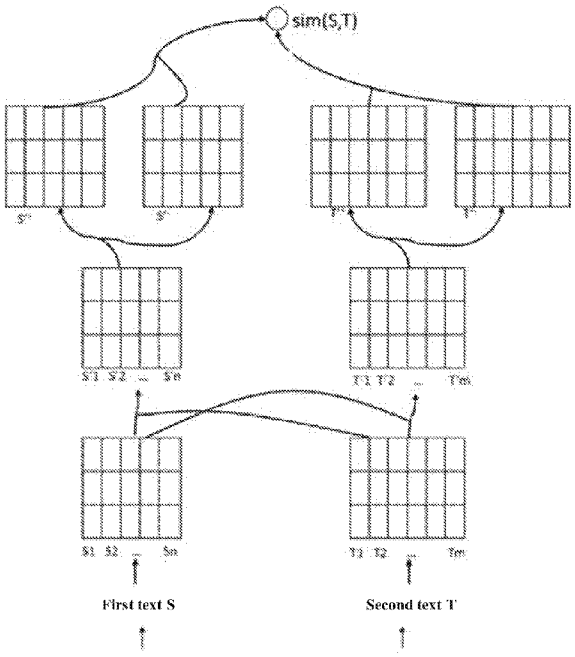


FIG. 1

(57) **Abstract:** The present application provides a method of calculating relevancy between a first text and a second text. The method includes respectively mapping the first text and the second text to vectors; determining a similar part and a dissimilar part between the first text and the second text based on the vectors; and calculating the relevancy between the first text and the second text using both the similar part and the dissimilar part.

WO 2019/140863 A1

METHOD OF CALCULATING RELEVANCY, APPARATUS FOR CALCULATING RELEVANCY, DATA QUERY APPARATUS, AND NON-TRANSITORY COMPUTER-READABLE STORAGE MEDIUM

CROSS-REFERENCE TO RELATED APPLICATION

[0001] This application claims priority to Chinese Patent Application No. 201810060942.7, filed January 22, 2018, the contents of which are incorporated by reference in the entirety.

TECHNICAL FIELD

[0002] The present invention relates to data processing technology, more particularly, to a method of calculating relevancy, an apparatus for calculating relevancy, a data query apparatus, and a non-transitory computer-readable storage medium.

BACKGROUND

[0003] With the development of internet and big data technology, a huge amount of information becomes available. Various search engines have been developed to facilitate a user to search for useful information based on a keyword entry. For example, a medical professional can search various medical literatures, books, and websites on the internet based on a search query. A patient may seek health-related information on health information websites.

SUMMARY

[0004] In one aspect, the present invention provides a method of calculating relevancy between a first text and a second text, comprising respectively mapping the first text and the second text to vectors; determining a similar part and a dissimilar part between the first text and the second text based on the vectors; and calculating the relevancy between the first text and the second text using both the similar part and the dissimilar part.

[0005] Optionally, determining the similar part and the dissimilar part comprises determining a similar part and a dissimilar part of the first text; and determining a similar part and a dissimilar part of the second text.

[0006] Optionally, subsequent to respectively mapping the first text and the second text to vectors, further comprising performing dimension reduction on the vectors, thereby representing the first text and the second text respectively using low-dimension dense vectors.

[0007] Optionally, dimension reduction is performed using one or more methods selected from a group consisting of Word2Vec, Sentence2Vec, and Doc2Vec.

[0008] Optionally, determining the similar part and dissimilar part comprises performing a semantic content analysis based on the first text and the second text.

[0009] Optionally, the semantic content analysis comprises performing a sematic matching process between the first text and the second text thereby obtaining a sematic overlay between the first text and the second text.

[0010] Optionally, performing the sematic matching process comprises reconstructing a vector of a word in the first text using word vectors of the second text; and calculating a semantic overlay based on a result of the reconstructing.

[0011] Optionally, determining the similar part and dissimilar part comprises performing a semantics decomposition on the first text and the second text.

[0012] Optionally, the semantic overlay is calculated based on Equation (1):

$$\min(\left\| S_i - \sum_{j=1}^{m} \alpha_{i,j} \times T_j \right\|^2 + \lambda \sum_{j=1}^{m} {\alpha_{i,j}}^2) \quad (1);$$

[0013] wherein S_i is a d×1 column vector of the first text; T_j is a d×1 column vector of the second text; $\alpha_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number.

[0014] Optionally, the similar part of the first text is calculated based on $S_i^{'} = \sum_{j=1}^{m} A_{i,j} \times T_j$; wherein $S_i^{'}$ represents the similar part of the first text, and $A_{i,j}$ is a matrix of $\alpha_{i,j}$; and S_i - $S_i^{'}$ represents the dissimilar part of the first text.

[0015] Optionally, the semantic overlay is calculated based on Equation (2):

$$\min(\left\| T_i - \sum_{j=1}^{m} \beta_{i,j} \times S_j \right\|^2 + \lambda \sum_{j=1}^{m} {\beta_{i,j}}^2) \quad (2);$$

[0016] wherein T_i is a d×1 column vector of the second text; S_j is a d×1 column vector of the first text; $\beta_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number.

[0017] Optionally, the similar part of the second text is calculated based on $T_j^{'} = \sum_{j=1}^{m} A_{i,j} \times S_j$; wherein $T_j^{'}$ represents the similar part of the second text, and $A_{i,j}$ is a matrix of $\beta_{i,j}$; and $T_j - T_j^{'}$ represents the dissimilar part of the second text.

[0018] Optionally, the relevancy between the first text and the second text is calculated using a Recurrent Neural Network using the similar part and the dissimilar part between the first text and the second text as input.

[0019] Optionally, the input comprises a similar part and a dissimilar part of the first text; and a similar part and a dissimilar part of the second text.

[0020] Optionally, the method further comprises training the Recurrent Neural Network using a sample data represented as (Ss, Ts, L), wherein Ss stands for a first sample text, Ts stands for a second sample text, and L represents a sample relevancy.

[0021] Optionally, training the Recurrent Neural Network further comprises assigning a first value of granularity to a first relevancy in a first range and a second value of granularity to a second relevancy in a second range.

[0022] In another aspect, the present invention provides an apparatus for calculating relevancy between a first text and a second text, comprising a memory; and one or more processors; wherein the memory and the one or more processors are connected with each other; the memory stores computer-executable instructions for controlling the one or more processors to respectively map the first text and the second text to vectors; determine a similar part and a dissimilar part between the first text and the second text based on the vectors; and calculate the relevancy between the first text and the second text using both the similar part and the dissimilar part.

[0023] Optionally, the memory stores computer-executable instructions for controlling the one or more processors to determine a similar part and a dissimilar part of the first text; and determine a similar part and a dissimilar part of the second text.

[0024] Optionally, the memory further stores computer-executable instructions for controlling the one or more processors to, subsequent to respectively mapping the first text and the second text to vectors, perform dimension reduction on the vectors, thereby representing the first text and the second text respectively using low-dimension dense vectors.

[0025] Optionally, dimension reduction is performed using one or more methods selected from a group consisting of Word2Vec, Sentence2Vec, and Doc2Vec.

[0026] Optionally, the memory further stores computer-executable instructions for controlling the one or more processors to perform a semantic content analysis based on the first text and the second text thereby determining the similar part and dissimilar part.

[0027] Optionally, the memory further stores computer-executable instructions for controlling the one or more processors to perform a sematic matching process between the first text and the second text thereby obtaining a sematic overlay between the first text and the second text.

[0028] Optionally, the memory further stores computer-executable instructions for controlling the one or more processors to reconstruct a vector of a word in the first text using word vectors of the second text, and calculate a semantic overlay based on a result of reconstruction.

[0029] Optionally, the memory further stores computer-executable instructions for controlling the one or more processors to perform a semantics decomposition on the first text and the second text thereby determining the similar part and dissimilar part.

[0030] Optionally, the semantic overlay is calculated based on Equation (1):

$$\min\left(\left\| S_i - \sum_{j=1}^{m} \alpha_{i,j} \times T_j \right\|^2 + \lambda \sum_{j=1}^{m} \alpha_{i,j}^2\right) \qquad (1);$$

[0031] wherein S_i is a d×1 column vector of the first text; T_j is a d×1 column vector of the second text; $\alpha_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number.

[0032] Optionally, the similar part of the first text is calculated based on $S_i^{'} = \sum_{j=1}^{m} A_{i,j} \times T_j$; wherein $S_i^{'}$ represents the similar part of the first text, and $A_{i,j}$ is a matrix of $\alpha_{i,j}$; and S_i - $S_i^{'}$ represents the dissimilar part of the first text.

[0033] Optionally, the semantic overlay is calculated based on Equation (2):

$$\min\left(\left\| T_i - \sum_{j=1}^{m} \beta_{i,j} \times S_j \right\|^2 + \lambda \sum_{j=1}^{m} \beta_{i,j}{}^2\right) \qquad (2);$$

[0034] wherein T_i is a d×1 column vector of the second text; S_j is a d×1 column vector of the first text; $\beta_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number.

[0035] Optionally, the similar part of the second text is calculated based on $T_j^{'} = \sum_{j=1}^{m} A_{i,j} \times S_j$; wherein $T_j^{'}$ represents the similar part of the second text, and $A_{i,j}$ is a matrix of $\beta_{i,j}$; and $T_j - T_j^{'}$ represents the dissimilar part of the second text.

[0036] Optionally, the apparatus further comprises a Recurrent Neural Network using the similar part and the dissimilar part of the first text and the second text as input to calculate the relevancy between the first text and the second text.

[0037] Optionally, the input comprises a similar part and a dissimilar part of the first text; and a similar part and a dissimilar part of the second text.

[0038] Optionally, the memory further stores computer-executable instructions for controlling the one or more processors to train the Recurrent Neural Network using a sample data represented as (Ss, Ts, L), wherein Ss stands for a first sample text, Ts stands for a second sample text, and L represents a sample relevancy.

[0039] Optionally, the memory further stores computer-executable instructions for controlling the one or more processors to assign a first value of granularity to a first relevancy in a first range and a second value of granularity to a second relevancy in a second range.

[0040] In another aspect, the present invention provides a data query apparatus, comprising the apparatus for calculating relevancy described herein.

[0041] In another aspect, the present invention provides a data query method, comprising calculating relevancy between a first text and a second text according to the method of calculating relevancy described herein, the second text being a text selected from a plurality of potential target texts; ranking the plurality of potential target texts based on calculation of relevancy between the plurality of potential target texts and the first text; and selecting a target text out of the plurality of potential target texts based on a resulting of ranking the plurality of potential target texts.

[0042] In another aspect, the present invention provides a non-transitory computer-readable storage medium storing computer-readable instructions, the computer-readable instructions

being executable by a processor to cause the processor to perform respectively map a first text and a second text to vectors; determine a similar part and a dissimilar part between the first text and the second text based on the vectors; and calculate a relevancy between a first text and a second text using both the similar part and the dissimilar part.

BRIEF DESCRIPTION OF THE FIGURES

[0043] The following drawings are merely examples for illustrative purposes according to various disclosed embodiments and are not intended to limit the scope of the present invention.

[0044] FIG. 1 is a flow chat illustrating a method of calculating relevancy in some embodiments according to the present disclosure.

[0045] FIG. 2 is a schematic diagram illustrating the structure of an apparatus for calculating relevancy in some embodiments according to the present disclosure.

[0046] FIG. 3 is a schematic diagram illustrating the structure of an apparatus for calculating relevancy in some embodiments according to the present disclosure.

[0047] FIG. 4 is a schematic diagram illustrating the structure of a data query apparatus in some embodiments according to the present disclosure.

[0048] FIG. 5 is a schematic diagram illustrating an electronic apparatus in some embodiments according to the present disclosure.

DETAILED DESCRIPTION

[0049] The disclosure will now be described more specifically with reference to the following embodiments. It is to be noted that the following descriptions of some embodiments are presented herein for purpose of illustration and description only. It is not intended to be exhaustive or to be limited to the precise form disclosed.

[0050] In a conventional method of calculating relevancy, typically similarity between texts is considered, but not the dissimilarity between texts. In the present disclosure, it is discovered for the first time that dissimilarity between texts also provides important semantic information that facilitates accurate information matching in the process of calculating relevancy. By also considering dissimilarity between texts, a better and more accuracy result can be obtained in calculating relevancy, rendering information that is more useful in response to a user query.

[0051] Accordingly, the present disclosure provides, inter alia, a method of calculating relevancy, an apparatus for calculating relevancy, a data query apparatus, an electronic apparatus that substantially obviate one or more of the problems due to limitations and disadvantages of the related art. In one aspect, the present disclosure provides a method of calculating relevancy. In some embodiments, the method includes respectively mapping the first text and the second text to vectors; determining a similar part and a dissimilar part between the first text and the second text based on the vectors; and calculating the relevancy between the first text and the second text using both the similar part and the dissimilar part. In one example, the method includes mapping the first text into a first set of vectors, and mapping the second text into a second set of vectors. Optionally, the step of determining the similar part and the dissimilar part includes determining a similar part and a dissimilar part of the first text; and determining a similar part and a dissimilar part of the second text.

[0052] As used herein, the term "similar part" refers to a part of a text having a similarity measure higher than or equal to a threshold value, and the term "dissimilar part" refers to a part of a text having a similarity measure lower than the threshold value. Optionally, a similar part of a first text refers to a part (e.g., a word, a sentence, a document) of a first text having an identical occurrence in the second text or having a sematic overlay between the first text and the second text, and a similar part of a second text refers to a part (e.g., a word, a sentence, a document) of a second text having an identical occurrence in the first text or having a sematic overlay between the first text and the second text. Optionally, a dissimilar part of a first text refers to a part (e.g., a word, a sentence, a document) of a first text that does not have an identical occurrence in the second text and does not have a sematic overlay between the first text and the second text, and a dissimilar part of a second text refers to a part (e.g., a word, a sentence, a document) of a second text that does not have an identical occurrence in the first text and does not have a sematic overlay between the first text and the second text.

[0053] As used herein, the term "relevancy" refers to a measure of a degree of relevancy between two texts, e.g., how pertinent, connected, or applicable a text is to another text (e.g., a query text). Optionally, values of relevancy may be expressed in granularity, and may not have a continuous relationship (e.g., having a 0, 0.5, or 1 relationship). Optionally, the relevancy is calculated by a relevancy algorithm. Optionally, the relevancy algorithm may be trained using a sample data to calculate relevancy scores. Optionally, the relevancy is

determined based on a plurality of criteria, e.g., considering both the similar part and the dissimilar part of the texts.

[0054] In one example, a user enters a query: "introduction on cold." Three results are found in the database. Text 1: This article introduces sinusitis. Text 2: This article introduces influenza, not the common cold. Text 3: This article introduces influenza, not the pneumonia. In one example, a function sim(.,.) used to represent the similarities between texts. When the similarities between the query text entered by the user and the three texts found in the database are calculated, it can be relatively easily determined that sim("Query Text", "Text 2") > sim("Query Text", "Text 1") and sim("Query Text", "Text 3") > sim("Query Text", "Text 1"). However, it is relatively more difficult to compare between sim("Query Text", "Text 2") and sim("Query Text", "Text 3"). When the dissimilarities between texts are also considered, however, a comparison between sim("Query Text", "Text 2") and sim("Query Text", "Text 3") can be performed. For example, Text 2 emphasizes more on the difference between the influenza and the common cold, e.g., the unique characteristics of the influenza, which are not commonly shared by the common cold. On the other hand, Text 3 emphasizes the difference between the influenza and the pneumonia. Because the influenza and the pneumonia have much less commonly shared characteristics, Text 3 may contain more information about the common characteristics of cold. Thus, it follows that Text 3 is more relevant to the Query Text as compared to Text 2.

[0055] In another example, a user enters a query asking for information on a genus A (having a plurality of species, e.g., species i and species ii). Three results are found in the database. Text 1 contains information on a closely related genus B or a species in a closely related genus B. Text 2 contain information on difference between species i and species ii. Text 3 contain information on difference between species i and a related genus C. In one example, a function sim(.,.) used to represent the similarities between texts. When the similarities between the query text entered by the user and the three texts found in the database are calculated, it can be relatively easily determined that sim("Query Text", "Text 2") > sim("Query Text", "Text 1") and sim("Query Text", "Text 3") > sim("Query Text", "Text 1"). However, it is relatively more difficult to compare between sim("Query Text", "Text 2") and sim("Query Text", "Text 3"). When the dissimilarities between texts are also considered, however, a comparison between sim("Query Text", "Text 2") and sim("Query Text", "Text 3") can be performed. For example, Text 2 emphasizes more on the difference between species i and species ii, e.g., the unique characteristics of species i, which are not

commonly shared by the species ii. On the other hand, Text 3 emphasizes the difference between species i and the related genus C. Because species i and the related genus C have much less commonly shared characteristics, Text 3 may contain more information about the common characteristics of the genus A. Thus, it follows that Text 3 is more relevant to the Query Text as compared to Text 2.

[0056] In yet another example, a user enters a query asking for information on a "cancer pain" (having various types such as cancer nerve pain and cancer bone pain). Three results are found in the database. Text 1 contains information on migraine. Text 2 contain information on difference between cancer nerve pain and cancer bone pain. Text 3 contain information on how to tell the difference between cancer bone pain and arthritis pain. In one example, a function sim(.,.) used to represent the similarities between texts. When the similarities between the query text entered by the user and the three texts found in the database are calculated, it can be relatively easily determined that sim("Query Text", "Text 2") > sim("Query Text", "Text 1") and sim("Query Text", "Text 3") > sim("Query Text", "Text 1"). However, it is relatively more difficult to compare between sim("Query Text", "Text 2") and sim("Query Text", "Text 3"). When the dissimilarities between texts are also considered, however, a comparison between sim("Query Text", "Text 2") and sim("Query Text", "Text 3") can be performed. For example, Text 2 emphasizes more on the difference between cancer nerve pain and cancer bone pain, e.g., the unique characteristics of cancer nerve pain, which are not commonly shared by cancer bone pain. On the other hand, Text 3 emphasizes the difference cancer bone pain and arthritis pain. Because cancer bone pain and arthritis pain are caused by two completely different pathological conditions, Text 3 may contain more information about the common characteristics of cancer pain. Thus, it follows that Text 3 is more relevant to the Query Text as compared to Text 2.

[0057] FIG. 1 is a flow chat illustrating a method of calculating relevancy in some embodiments according to the present disclosure. Referring to FIG. 1, a method of calculating relevancy in some embodiments includes at least obtaining a first text S and obtaining a second text T; respectively mapping the first text S and the second text T to vectors; determining a similar part and a dissimilar part between the first text S and the second text T; and calculating the relevancy between the first text S and the second text T using the similar part and the dissimilar part. Optionally, the step of determining the similar part and the dissimilar part includes determining a similar part and a dissimilar part of the first text; and determining a similar part and a dissimilar part of the second text.

[0058] In some embodiments, the first text S or the second text T is a sentence including a plurality of words. Optionally, the first text S or the second text T is a paragraph including a plurality of sentences. Optionally, the first text S or the second text T is a document including a plurality of paragraphs. Optionally, the first text S or the second text T includes a huge number of documents.

[0059] In some embodiments, the first text S and the second text T is obtained from a single data source. In some embodiments, the first text S and the second text T is obtained from a plurality of data sources. Optionally, the source of the first text S and the second text T includes texts stored in or outputted by the local devices (e.g., a cellphone, a pad, and a computer). Optionally, the source of the first text and the second text includes texts transmitted through internet.

[0060] In some embodiments, the first text S and the second text T are respectively mapped to separate vectors. For example, the first text S is mapped to S1, S2, …, Sn. The second text T is mapped to T1, T2, …, Tm. Therefore, S1-Sn is an array of words forming the first text S. For example, S1-Sn is a vector corresponding to the first text S. In one example, S="symptom of cold", S1="symptom", S2="of", S3="cold". Similarly, T1-Tn is an array of words forming the second text T, T1-Tn is a vector corresponding to the second text T. In one example, T="symptom of influenza", T1="symptom", T2="of", T3= "influenza."

[0061] Various appropriate methods may be used to vectorize text. Examples of appropriate method for vectorizing text include Vector Space Model (VSM) and distributional word representation. Various appropriate computational models may be used to map a text to a vector by distributional word representation, including Latent Semantic Analysis (LSA), Probabilistic Latent Semantic Analysis (PLSA), and Latent Dirichlet Allocation (LDA).

[0062] In some embodiments, subsequent to mapping the first text S or the second text T to vectors, the method further includes performing dimension reduction on the vectors, thereby representing the text using low-dimension dense vectors. Optional, dimension reduction is performed by Principal component analysis (PCA). If a multivariate data set is visualized as a set of coordinates in a high-dimensional data space (one axis per variable), PCA can supply the user with a lower-dimensional picture, a projection of this object when viewed from its most informative viewpoint. This is done by using only the first few principal components so that the dimensionality of the transformed data is reduced. Therefore, PCA can linearly

transform multiple variances and obtain a few principal variances. Optionally, dimension reduction is performed using methods such as Word2Vec, Sentence2Vec, and Doc2Vec.

[0063] In some embodiments, a word embedding process is used to perform dimension reduction. The word embedding process may be conducted using methods such as Word2Vec. In one example, subsequent to dimension reduction, the vector S1-Sn of the first text S and the vector T1-Tm of the second text T are respectively mapped to two sets of low-dimension vectors. When the number of dimensions is d, each word of a text corresponds to a d×1 column vector. The first text S corresponds to a d×n matrix (i.e. matrix P), and the second text T corresponds to a d×m matrix (i.e. matrix Q).

[0064] In some embodiments, the similar part and dissimilar part between the first text and the second text are determined. Optionally, the similar part and dissimilar part are determined using a textual encoding method such as Simhash, Shingling, and Minhash. In one example, Simhash is used to binary encode the first text and the second text, and a Hamming distance between the first text and the second text is determined by performing XOR operation (i.e. execute XOR operation on the simhash result of those two texts). In the result of the XOR operation, if the number of result "1: exceeds three, the two texts are determined to be dissimilar (e.g., the dissimilar part); if the number of result "1" is less than or equal to three, the two texts are determined to be similar (e.g., the similar part).

[0065] In some embodiments, the step of determining a similar part and a dissimilar part between the first text S and the second text T includes performing a semantic content analysis based on the first text S and the second text T. Optionally, the semantic content analysis includes performing a sematic matching process between the first text and the second text thereby obtaining a sematic overlay between the first text S and the second text T.

[0066] In some embodiments, to calculate the similar part between the first text S and the second text T, it is necessary to know whether every word in S(T) can be semantically overlaid by the words in T(S). For example, the word "cold" in the Query Text can semantically overlay the word "influenza" in Text 3, but can hardly overlay the word "sinusitis" in Text 1. Therefore, sim("Query Text", "Text 3") > sim("Query Text", "Text 1").

[0067] In some embodiments, the step of determining the semantic overlay between the first text S and the second text T includes reconstructing a vector of a word in the first text S (e.g., a word Si) using word vectors of the second text T; and calculating the semantic

overlay based on a result of the reconstruction. Optionally, Equation (1) is used to calculate the semantic overlay of the words of the first text S over the second text T:

[0068] $$\min(\left\| S_i - \sum_{j=1}^{m} \alpha_{i,j} \times T_j \right\|^2 + \lambda \sum_{j=1}^{m} \alpha_{i,j}^2) \quad (1);$$

[0069] wherein each of S_i and T_j is a d×1 column vector; each of S_i and T_j is a word vector obtained from the word embedding process; $\alpha_{i,j}$ is parameter. $\lambda > 0$, and λ is a positive real number. Optionally, S_i is a d×1 column vector of the first text; T_j is a d×1 column vector of the second text; $\alpha_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number.

[0070] For example, the parameter $\alpha_{i,j}$ can be solved by the following method:

[0071] First, a convex function f_i is provided to satisfy $f_i = \left\| S_i - \sum_{j=1}^{m} \alpha_{i,j} \times T_j \right\|^2 + \lambda \sum_{j=1}^{m} \alpha_{i,j}^2$.

To obtain a minimum value for f_i, $\alpha_{i,j}$ should meet the condition of $\frac{\partial f_i}{\partial \alpha_{i,j}} = 0$.

[0072] Accordingly, it can be derived that

$$\frac{\partial f_i}{\partial \alpha_{i,j}} = -2(S_i - \sum_{k=1}^{m} \alpha_{i,k} T_k)^T T_j + 2\lambda \alpha_{i,j}$$

$$= -2\left[S_i^T T_j - \sum_{k=1}^{m} \alpha_{i,k} T_k^T T_j - \lambda \alpha_{i,j} \right] = 0 \quad (2);$$

$$\Rightarrow S_i^T T_j - \sum_{k=1}^{m} \alpha_{i,k} T_k^T T_j - \lambda \alpha_{i,j} = 0$$

[0073] wherein a superscript T stands for a transpose of an original matrix. Because the above formula is equally applicable when i=1,2,…,n and when j=1,2,…,m. A total number of n×m equations can be generated to solve values of n×m parameters.

[0074] To illustrate the process of solving the values of n×m parameters, the equation $S_i^T T_j - \sum_{k=1}^{m} \alpha_{i,k} T_k^T T_j - \lambda \alpha_{i,j} = 0$ can be better represented in matrix format. First, an n×m matrix R is provided, with an element $R_{i,j}$ in i-th row and j-th column to be $R_{i,j} = S_i^T T_j$. The matrix R can be represented as $R = P^T Q$, wherein P and Q are corresponding to matrixes

formed after word embedding. Second, a m×m matrix U is provided, with an element $U_{k,j}$ in k-th row and j-th column to be $U_{k,j}=T_k^T T_j$. The matrix U can be represented as $U=Q^TQ$, wherein Q corresponds to a matrix formed after word embedding. Third, an n×m matrix A is provided, with an element in i-th row and j-th column to be $\alpha_{i,j}$. Fourth, using the matrices R, U, and A, the equation $S_i^T T_j - \sum_{k=1}^{m} \alpha_{i,k} T_k^T T_j - \lambda \alpha_{i,j} = 0$ can be represented as

[0075]
$$\begin{aligned} & R - AU - \lambda A = 0 \\ & \Rightarrow R = AU + \lambda A \\ & \quad = A(U + \lambda I) \\ & \Rightarrow A = R(U + \lambda I)^{-1} \end{aligned}$$
(3); wherein, I is a m×m unit matrix.

[0076] During the process of solving equation (2), the matrix (U+ λ I) should be a non-singular matrix, i.e. an inverse matrix of the matrix (U+ λ I) exists. Because $U_{k,j}= T_k^T T_j$, U is a m×m real symmetric matrix. A diagonal matrix Λ similar to the matrix U exists, i.e. $V^{-1}UV = \Lambda$. Accordingly, it can be derived that $V^{-1}(U+\lambda I)V = \Lambda + \lambda I$. As long as λ is not equal to the negative eigenvalue of the matrix U, the eigenvalue of the matrix (U+ λ I). (i.e., the elements along the diagonal of Λ+ λ I) will not be zero, and the matrix (U+ λ I) is a non-singular matrix. The eigenvalue of matrix U can be any real number in real number field. Provided that the distribution function of matrix U is F(x), and because the value of x is continuous, for any provided real number a, a's probability value F(a)=0. Therefore, without *a priori* knowledge, the probability of pre-determining the value of λ to be equal to the negative eigenvalue of the matrix U is substantially zero, i.e., an inverse matrix of the matrix (U+ λ I) exists.

[0077] Similarly, using the same method, a semantic overlay of the words of the second text T over the first text S can also be obtained. Optionally, Equation (3) is used to calculate the semantic overlay of the words of the second text T over the first text S:

[0078]
$$\min\left(\left\| T_i - \sum_{j=1}^{m} \beta_{i,j} \times S_j \right\|^2 + \lambda \sum_{j=1}^{m} \beta_{i,j}{}^2\right) \quad (3);$$

[0079] wherein each of T_i and S_j is a d×1 column vector; each of T_i and S_j is a word vector obtained from the word embedding process; $\beta_{i,j}$ is parameter. $\lambda > 0$, and λ is a positive real

number. Optionally, T_i is a d×1 column vector of the first text; S_j is a d×1 column vector of the second text; $\beta_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number.

[0080] Various alternative methods may be used for solve the Equation (1) and the Equation (3) above. Examples of appropriate methods include numerical methods and algebraic methods.

[0081] In some embodiments, the semantics decomposition can be used to obtain the similar part and the dissimilar part between the first text S and the second text T (e.g., the similar part and dissimilar part of the first text S when comparing the first text S to the second text T, and the similar part and dissimilar part of the second text T when comparing the second text T to the first text S.

[0082] In some embodiments, the matrix A (see equation (2)) is based on how every word in the first text S is represented or overlaid in the second text T, wherein the i-th row of the matrix A shows how word S_i is represented or overlaid in text T. Provided that $S_i^{'} = \sum_{j=1}^{m} A_{i,j} \times T_j$, $S_i^{'}$ represents the similar part of the first text S, e.g., a part of word S_i of the first text S that can be overlaid by the second text T. Then, S_i - $S_i^{'}$ represents the dissimilar part of the first text S. The first text S is represented by two parts, similar part (represented by a d×n matrix, refer to the similar part S'^{+} in FIG. 1) and dissimilar part (represented by a d ×n matrix, refer to the dissimilar part S'^{-} in FIG. 2). Thus, the text can be represented by a d ×2n matrix instead of a d×n matrix. By adopting this method, the similar part and dissimilar part of the first text S when comparing the first text S and the second text T can be calculated.

[0083] Similarly, calculating the similar part and dissimilar part between the second text T and the first text S includes adopting the equation $T_j^{'} = \sum_{j=1}^{m} A_{i,j} \times S_j$, wherein $A_{i,j}$ is the matrix of $\beta_{i,j}$, $T_j^{'}$ represents the similar part of the second text T, and $T_j - T_j^{'}$ represents the dissimilar part of the second text T. The second text T includes two parts, a similar part (represented by a d×m matrix, refer to the similar part T'^{+} in FIG. 1) and a dissimilar part (represented by a d×m matrix, refer to the dissimilar part T'^{-} in FIG. 2). By adopting this

method, the similarity and dissimilarity of the second text T when comparing the first text S and the second text T can also be calculated.

[0084] In some embodiments, based on a determination of the similar part and the dissimilar part of each of the first text S and the second text T, the relevancy between the first text S and the second text T can be calculated. Optionally, calculating the relevancy includes determining a Jaccard similarity through textual encoding. Optionally, calculating the relevancy is performed by textual semantics calculation. Optionally, calculating the relevancy is performed by machine learning.

[0085] In some embodiments, the relevancy between the first text and the second text can be calculated using a Recurrent Neural Network (RNN). Optionally, using RNN includes using sample data to train RNN. In one example, the sample data is represented as (Ss, Ts, L), wherein Ss and Ts represent sample texts (e.g., Ss represent a first sample text, and Ts represent a second sample text), and L represents relevancy. Optionally, Ss and Ts are the originally inputted first text S and second text T. Optionally, Ss and Ts are sample texts different from the first text S and the second text T. Optionally, the granularity of the relevancy can be assigned. In one example, the granularity is 2, $L \in \{0,1\}$, wherein 0 represents irrelevant, and 1 represents relevant. In another example, the granularity is 3, $L \in \{0,0.5,1\}$, wherein 0 represents irrelevant, 0.5 represents somewhat relevant, 1 represents very relevant.

[0086] Optionally, in order to train the RNN with the sample data, the sample data is inputted in the RNN. A composite relevancy based on the similar part of the first text S, the similar part of the second text T, the dissimilar part of the first text S, and the dissimilar part of the second text T is then calculated. In one example, when the granularity is set to 2, a composite relevancy in a first range is defined to have a granularity of 1, and a composite relevancy in a second range different from the first range is defined to have a granularity of 0. In another example, when the granularity is set to 3, a composite relevancy in a first range is defined to have a granularity of 1, a composite relevancy in a second range is defined to have a granularity of 0.5, and a composite relevancy in a third range is defined to have a granularity of 0, the first range, the second range, and the third range being different from each other. Once the RNN is trained using the sample data, it can be used for calculating the relevancy on non-sample data. In a trained RNN, by inputting a $d \times 2n$ matrix which represents the first text S and a $d \times 2m$ matrix which represents the second text T (e.g.,

inputting the similar part and dissimilar part of the first text S and the second text T), the RNN can be used to calculate the relevancy based on input information.

[0087] In the present method of calculating the relevancy, the effects of both the similar part and the dissimilar part between two texts are considered. Consequently, the present method enables a relevancy calculation with a finer granularity classification, thereby enhancing the accuracy of textual matching, and rendering highly accurate search or query results.

[0088] In another aspect, the present disclosure also provides an apparatus for calculating relevancy. FIG. 2 and FIG. 3 are schematic diagrams illustrating the structure of an apparatus for calculating relevancy in some embodiments according to the present disclosure. Referring to FIG.2, the apparatus for calculating relevancy has a data acquirer 200 configured to obtain a first text and a second text; a mapping engine 210 configured to respectively map the first text and the second text to vectors; an relevancy calculator 220 configured to determine a similar part and a dissimilar part between the first text and the second text based on the vectors, and calculate the relevancy between the first text and the second text based on the similar part and the dissimilar part. Optionally, the relevancy calculator 220 configured to determine a similar part and a dissimilar part of the first text, and determining a similar part and a dissimilar part of the second text.

[0089] In some embodiments, referring to FIG. 3, the mapping engine 210 includes a dimension reduction processor 211 configured to reduce the dimension of vectors respectively corresponding to the first text and the second text.

[0090] In some embodiments, referring to FIG. 3, the relevancy calculator 220 includes a semantics matching processor 221 configured to perform semantic matching between the first text and the second text; a semantics decomposition processor 222 configured to determine the similar part and the dissimilar part between the first text and the second text; a calculator 223 configured to calculate the relevancy between the first text and the second text basing on the similar part and the dissimilar part. Optionally, the semantics decomposition processor 222 configured to determine a similar part and a dissimilar part of the first text, and determining a similar part and a dissimilar part of the second text.

[0091] Optionally, the semantics matching processor 221 is configured to determine semantic overlay between the first text and the second text by reconstructing the first text's

vector using the second text's vector, e.g., by reconstructing a vector of a word in the first text using word vectors of the second text.

[0092] In some embodiments, the semantics matching processor 221 is configured to use Equation (1), $\min(\left\| S_i - \sum_{j=1}^{m} \alpha_{i,j} \times T_j \right\|^2 + \lambda \sum_{j=1}^{m} \alpha_{i,j}^2)$, to calculate the semantic overlay of the words of the first text S over the second text T, wherein, S_i is a column vector (e.g., a d×1 column vector) of the first text; T_j is the column vector (e.g., a d×1 column vector) of the second text; $\alpha_{i,j}$ is a parameter for semantic overly; $\lambda > 0$, and λ is a positive real number. Optionally, S_i is a d×1 column vector of the first text; T_j is a d×1 column vector of the second text; $\alpha_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number.

[0093] In some embodiments, the semantics matching processor 221 is configured to use Equation (3), $$\min(\left\| T_i - \sum_{j=1}^{m} \beta_{i,j} \times S_j \right\|^2 + \lambda \sum_{j=1}^{m} \beta_{i,j}{}^2)$$, to calculate the semantic overlay of the words of the second text T over the first text S, wherein, T_i is a column vector (e.g., a d×1 column vector) of the second text; S_j is the column vector (e.g., a d×1 column vector) of the first text; $\beta_{i,j}$ is a parameter for semantic overly; $\lambda > 0$, and λ is a positive real number. Optionally, T_i is a d×1 column vector of the first text; S_j is a d×1 column vector of the second text; $\beta_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number.

[0094] Optionally, the semantics decomposition processor 222 is configured to obtain the similar part and the dissimilar part of the first text based on the equation of $S_i' = \sum_{j=1}^{m} A_{i,j} \times T_j$, wherein $A_{i,j}$ is the matrix of $\alpha_{i,j}$; S_i' stands for the similar part of the first text; and S_i - S_i' stands for the dissimilar part of the second text.

[0095] Optionally, the semantics decomposition processor 222 is configured to obtain the similar part and the dissimilar part of the second text based on the equation of $T_j' = \sum_{j=1}^{m} A_{i,j} \times S_j$, wherein $A_{i,j}$ is the matrix of $\beta_{i,j}$; T_j' stands for the similar part of the first text; and $T_j - T_j'$ stands for the dissimilar part of the second text.

[0096] In some embodiments, the apparatus for calculating relevancy includes a memory; and one or more processor. The memory and the at least one processor are connected with each other. The memory stores computer-executable instructions for controlling the one or more processors to respectively map the first text and the second text to vectors; determine a similar part and a dissimilar part between the first text and the second text based on the vectors; and calculate the relevancy between the first text and the second text using both the similar part and the dissimilar part. Optionally, the memory further stores computer-executable instructions for controlling the one or more processors to, subsequent to respectively mapping the first text and the second text to vectors, perform dimension reduction on the vectors, thereby representing the text using low-dimension dense vectors. Optionally, dimension reduction is performed using one or more methods selected from a group consisting of Word2Vec, Sentence2Vec, and Doc2Vec. Optionally, the memory stores computer-executable instructions for controlling the one or more processors to determine a similar part and a dissimilar part of the first text; and determine a similar part and a dissimilar part of the second text.

[0097] In some embodiments, the memory further stores computer-executable instructions for controlling the one or more processors to perform a semantic content analysis based on the first text and the second text thereby determining the similar part and dissimilar part. Optionally, the memory further stores computer-executable instructions for controlling the one or more processors to perform a sematic matching process between the first text and the second text thereby obtaining a sematic overlay between the first text and the second text. Optionally, the memory further stores computer-executable instructions for controlling the one or more processors to reconstruct a vector of a word in the first text S using word vectors of the second text, and calculate the semantic overlay based on a result of reconstruction. Optionally, the semantic overlay of the words of the first text S over the second text T is calculated based on Equation (1):

[0098] $$\min\left(\left\| S_i - \sum_{j=1}^{m} \alpha_{i,j} \times T_j \right\|^2 + \lambda \sum_{j=1}^{m} \alpha_{i,j}^2\right) \qquad (1);$$

[0099] wherein S_i is a d×1 column vector of the first text; T_j is a d×1 column vector of the second text; $\alpha_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number. Optionally, S_i is a d×1 column vector of the first text; T_j is a d×1 column vector of

the second text; $\alpha_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number.

[0100] Optionally, the semantic overlay of the words of the second text T over the first text S is calculated based on Equation (3),

$$\min(\left\| T_i - \sum_{j=1}^{m} \beta_{i,j} \times S_j \right\|^2 + \lambda \sum_{j=1}^{m} \beta_{i,j}{}^2) \quad (3);$$

[0101] wherein, T_i is a column vector (e.g., a d×1 column vector) of the second text; S_j is the column vector (e.g., a d×1 column vector) of the first text; $\beta_{i,j}$ is a parameter for semantic overly; $\lambda > 0$, and λ is a positive real number. Optionally, T_i is a d×1 column vector of the first text; S_j is a d×1 column vector of the second text; $\beta_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number.

[0102] Optionally, the similar part of the first text is calculated based on $S_i^{'} = \sum_{j=1}^{m} A_{i,j} \times T_j$; $S_i^{'}$ represents the similar part of the first text, and $A_{i,j}$ is a matrix of $\alpha_{i,j}$. Optionally, S_i - $S_i^{'}$ represents the dissimilar part of the first text.

[0103] Optionally, the similar part of the second text is calculated based on $T_j^{'} = \sum_{j=1}^{m} A_{i,j} \times S_j$; $T_j^{'}$ represents the similar part of the second text, and $A_{i,j}$ is a matrix of $\beta_{i,j}$. Optionally, $T_j - T_j^{'}$ represents the dissimilar part of the second text.

[0104] In some embodiments, the memory further stores computer-executable instructions for controlling the one or more processors to perform a semantics decomposition on the first text and the second text thereby determining the similar part and dissimilar part.

[0105] In some embodiments, the apparatus further includes a Recurrent Neural Network using the similar part and the dissimilar part of the first text and the second text as input to calculate the relevancy between the first text and the second text. Optionally, the memory further stores computer-executable instructions for controlling the one or more processors to train the Recurrent Neural Network using a sample data represented as (Ss, Ts, L), wherein Ss stands for a first sample text, Ts stands for a second sample text, and L represents a sample relevancy. Optionally, Ss and Ts are the originally inputted first text S and second text T.

Optionally, Ss and Ts are sample texts different from the first text S and the second text T. Optionally, the memory further stores computer-executable instructions for controlling the one or more processors to assign a first value of granularity to a first relevancy in a first range and a second value of granularity to a second relevancy in a second range.

[0106] In another aspect, the present disclosure also provides a non-transitory computer-readable storage medium storing computer-readable instructions. In some embodiments, the computer-readable instructions are executable by a processor to cause the processor to perform one or more steps of the method described herein. In some embodiments, the computer-readable instructions are executable by a processor to cause the processor to respectively map the first text and the second text to vectors; determine a similar part and a dissimilar part between the first text and the second text based on the vectors; and calculate the relevancy between the first text and the second text using both the similar part and the dissimilar part. Optionally, the computer-readable instructions are executable by a processor to further cause the processor to, subsequent to respectively mapping the first text and the second text to vectors, perform dimension reduction on the vectors, thereby representing the text using low-dimension dense vectors. Optionally, dimension reduction is performed using one or more methods selected from a group consisting of Word2Vec, Sentence2Vec, and Doc2Vec. Optionally, the computer-readable instructions are executable by a processor to cause the processor to determine a similar part and a dissimilar part of the first text; and determine a similar part and a dissimilar part of the second text.

[0107] In some embodiments, the computer-readable instructions are executable by a processor to cause the processor to perform a semantic content analysis based on the first text and the second text thereby determining the similar part and dissimilar part. Optionally, the computer-readable instructions are executable by a processor to cause the processor to perform a sematic matching process between the first text and the second text thereby obtaining a sematic overlay between the first text and the second text. Optionally, the computer-readable instructions are executable by a processor to cause the processor to reconstruct a vector of a word in the first text S using word vectors of the second text, and calculate the semantic overlay based on a result of reconstruction. Optionally, the semantic overlay of the words of the first text S over the second text T is calculated based on Equation (1):

[0108] $$\min(\left\| S_i - \sum_{j=1}^{m} \alpha_{i,j} \times T_j \right\|^2 + \lambda \sum_{j=1}^{m} \alpha_{i,j}^2) \quad (1);$$

[0109] wherein S_i is a d×1 column vector of the first text; T_j is a d×1 column vector of the second text; $\alpha_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number. Optionally, S_i is a d×1 column vector of the first text; T_j is a d×1 column vector of the second text; $\alpha_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number.

[0110] Optionally, the semantic overlay of the words of the second text T over the first text S is calculated based on Equation (3):

$$\min(\left\| T_i - \sum_{j=1}^{m} \beta_{i,j} \times S_j \right\|^2 + \lambda \sum_{j=1}^{m} \beta_{i,j}^2) \quad (3);$$

[0111] wherein, T_i is a column vector (e.g., a d×1 column vector) of the second text; S_j is the column vector (e.g., a d×1 column vector) of the first text; $\beta_{i,j}$ is a parameter for semantic overly; $\lambda > 0$, and λ is a positive real number. Optionally, T_i is a d×1 column vector of the first text; S_j is a d×1 column vector of the second text; $\beta_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number.

[0112] Optionally, the similar part of the first text is calculated based on $S_i^{'} = \sum_{j=1}^{m} A_{i,j} \times T_j$; $S_i^{'}$ represents the similar part of the first text, and $A_{i,j}$ is a matrix of $\alpha_{i,j}$. Optionally, S_i - $S_i^{'}$ represents the dissimilar part of the first text.

[0113] Optionally, the similar part of the second text is calculated based on $T_j^{'} = \sum_{j=1}^{m} A_{i,j} \times S_j$; $T_j^{'}$ represents the similar part of the second text, and $A_{i,j}$ is a matrix of $\beta_{i,j}$. Optionally, $T_j - T_j^{'}$ represents the dissimilar part of the second text.

[0114] In some embodiments, the computer-readable instructions are executable by a processor to cause the processor to perform a semantics decomposition on the first text and the second text thereby determining the similar part and dissimilar part.

[0115] In another aspect, the present disclosure also provides a data query apparatus. In some embodiments, the data query apparatus includes the apparatus for calculating relevancy as described herein. FIG. 4 is a schematic diagram illustrating the structure of a data query apparatus in some embodiments according to the present disclosure. Referring to FIG. 4, the data query apparatus in some embodiments includes a human-computer interaction interface 401. Optionally, the human-computer interaction interface 401is a medical question answering platform, which can receive patients' questions and display the corresponding answers. Optionally, the human-computer interaction interface 401is a medical literature searching system, which can receive the content the user is looking for and display the corresponding literature.

[0116] In some embodiments, the data query apparatus further includes a content matching processor 402 and a database 403. Optionally, the content matching processor 402 includes the relevancy calculator 220 described herein.

[0117] In some embodiments, when the user enters the content he wants to search on the human-computer interaction interface 401, the content matching processor 402 collects the potential target contents from the database 403. Optionally, the content matching processor 402 adopts the relevancy calculating methods described herein to calculate the relevancy between the potential target contents and the content entered by the user. Optionally, the content matching processor 402 selects the content the user looks for from the potential target contents based on a ranking of relevancy.

[0118] In another aspect, the present disclosure also provides a data query method. In some embodiments, the data query method includes calculating relevancy between a first text and a second text according to the method of calculating relevancy described herein. The second text is a text selected from a plurality of potential target texts. Optionally, the data query method further includes ranking the plurality of potential target texts based on calculation of relevancy between the plurality of potential target texts and the first text; and selecting a target text out of the plurality of potential target texts based on a resulting of ranking the plurality of potential target texts. Optionally, the method further includes outputting the target text as a result of data query to a user.

[0119] In another aspect, the present disclosure further provides an electronic apparatus 500. FIG. 5 is a schematic diagram illustrating the electronic apparatus in some embodiments according to the present disclosure. Optionally, the electronic apparatus 500 includes a

processor 502. Optionally, the processor 502 is configured to run computer instructions to perform at least one step of the relevancy calculation described herein.

[0120] Optionally, the electronic apparatus 500 includes at least a memory 501 connected to at least one the processor 502. Optionally, the memory 501 is configured to store computer instructions.

[0121] In some embodiment, the electronic apparatus 500 is a local end apparatus, which means the electronic apparatus 500 can calculate the relevancy on the user end. In some embodiments, the electronic apparatus 500 is a local and remote interaction apparatus, i.e. the electronic apparatus 500 obtains at least a first text and a second text at the user end apparatus (i.e. local end), and the remote internet processor (i.e. remote end) receives the first text and the second text and perform the similarity calculation.

[0122] In some embodiments, the electronic apparatus 500 can include a plurality of user end apparatus and a plurality of remote internet processor connecting to the user end apparatus. Optionally, the plurality of the user end apparatus upload information containing at least a first text and a second text to the remote internet processor. Optionally, the information uploaded is the query text entered or stored by the user. Optionally, the remote internet processor collects the information uploaded by every user end apparatus. Optionally, the remote internet processor then performs the relevancy calculation.

[0123] In some embodiments, the memory 501 can be formed by any type of volatile storage devices or non-volatile storage devices or the combination of both the volatile and non-volatile storage devices. The types of volatile storage devices or non-volatile storage devices include Static Random-Access Memory (SRAM), Electrically Erasable Programmable Read-only Memory (EEPROM), Erasable Programmable Read-only Memory (EPROM), Programmable Read-only Memory (PROM), Read-only Memory (ROM), magnetic memory, flash memory, disk or CD.

[0124] In some embodiments, the processor 502 is a logical operation apparatus having data processing capabilities and/or program execution capabilities, such as Central Processing Unit (CPU), Field-Programmable Gate Array (FPGA), Microcontroller Unit (MCU), Digital Signal Processor (DSP), Application-Specific Integrated Circuit (ASIC), graphics processing unit (GPU). Optionally, one or a plurality of the processor can be configured to execute the relevancy calculation concurrently with a parallel processor. Optionally, one or a plurality of

the processor can be configured to execute part of the relevancy calculation. Optionally, other processors can be configured to execute the remaining part of the relevancy calculation.

[0125] In some embodiments, the computer instructions include at least one processor's operation defined by the instruction set corresponding to the processor. Optionally, the computer instruction can be logically contained and represented by at least one computer programs.

[0126] In some embodiments, the electronic apparatus 500 can be connected to input apparatus such as a user interface, a keyboard. Optionally, the electronic apparatus 500 can be connected to output apparatus such as a speaker. Optionally, the electronic apparatus can be connected to the apparatus performing the interaction between user and the electronic apparatus 500, such as a display device.

[0127] In some embodiments, the electronic apparatus 500 can be connected to various input apparatus, output apparatus or other interaction apparatus through networks including wireless network, wired network, and/or any combination of wireless and wired network. Optionally, the network includes a local area network, the internet, a telecommunication network, an internet of thins based on the internet and/or a telecommunications network., and any combination of the above-mentioned network. Optionally, the wired network can use different materials to communicate, including twisted pair, coaxial able, and fiber. Optionally, the wireless network can also use different materials to communicate, including a 3G/4G/5G mobile communication network, Bluetooth, Zigbee, and Wi-Fi.

[0128] The present disclosure also provides a computer readable storage medium, which is configured to store computer instructions. When the computer instructions are executed, at least one of the step of the relevancy calculation is performed.

[0129] The foregoing description of the embodiments of the invention has been presented for purposes of illustration and description. It is not intended to be exhaustive or to limit the invention to the precise form or to exemplary embodiments disclosed. Accordingly, the foregoing description should be regarded as illustrative rather than restrictive. Obviously, many modifications and variations will be apparent to practitioners skilled in this art. The embodiments are chosen and described in order to explain the principles of the invention and its best mode practical application, thereby to enable persons skilled in the art to understand the invention for various embodiments and with various modifications as are suited to the particular use or implementation contemplated. It is intended that the scope of the invention

be defined by the claims appended hereto and their equivalents in which all terms are meant in their broadest reasonable sense unless otherwise indicated. Therefore, the term “the invention”, “the present invention” or the like does not necessarily limit the claim scope to a specific embodiment, and the reference to exemplary embodiments of the invention does not imply a limitation on the invention, and no such limitation is to be inferred. The invention is limited only by the spirit and scope of the appended claims. Moreover, these claims may refer to use “first”, “second”, etc. following with noun or element. Such terms should be understood as a nomenclature and should not be construed as giving the limitation on the number of the elements modified by such nomenclature unless specific number has been given. Any advantages and benefits described may not apply to all embodiments of the invention. It should be appreciated that variations may be made in the embodiments described by persons skilled in the art without departing from the scope of the present invention as defined by the following claims. Moreover, no element and component in the present disclosure is intended to be dedicated to the public regardless of whether the element or component is explicitly recited in the following claims.

WHAT IS CLAIMED IS:

1. A method of calculating relevancy between a first text and a second text, comprising:

respectively mapping the first text and the second text to vectors;

determining a similar part and a dissimilar part between the first text and the second text based on the vectors; and

calculating the relevancy between the first text and the second text using both the similar part and the dissimilar part.

2. The method of claim 1, wherein determining the similar part and the dissimilar part comprises:

determining a similar part and a dissimilar part of the first text; and

determining a similar part and a dissimilar part of the second text.

3. The method of claim 1, subsequent to respectively mapping the first text and the second text to vectors, further comprising performing dimension reduction on the vectors, thereby representing the first text and the second text respectively using low-dimension dense vectors.

4. The method of claim 3, wherein dimension reduction is performed using one or more methods selected from a group consisting of Word2Vec, Sentence2Vec, and Doc2Vec.

5. The method of any one of claims 1 to 4, wherein determining the similar part and dissimilar part comprises performing a semantic content analysis based on the first text and the second text.

6. The method of claim 5, wherein the semantic content analysis comprises performing a sematic matching process between the first text and the second text thereby obtaining a sematic overlay between the first text and the second text.

7. The method of claim 6, wherein performing the sematic matching process comprises reconstructing a vector of a word in the first text using word vectors of the second text; and

calculating a semantic overlay based on a result of the reconstructing.

8. The method of any one of claims 1 to 7, wherein determining the similar part and dissimilar part comprises performing a semantics decomposition on the first text and the second text.

9. The method of claim 7, wherein the semantic overlay is calculated based on Equation (1):

$$\min(\left\| S_i - \sum_{j=1}^{m} \alpha_{i,j} \times T_j \right\|^2 + \lambda \sum_{j=1}^{m} \alpha_{i,j}{}^2) \quad (1);$$

wherein S_i is a d×1 column vector of the first text; T_j is a d×1 column vector of the second text; $\alpha_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number.

10. The method of claim 9, wherein the similar part of the first text is calculated based on $S_i^{'} = \sum_{j=1}^{m} A_{i,j} \times T_j$;

wherein $S_i^{'}$ represents the similar part of the first text, and $A_{i,j}$ is a matrix of $\alpha_{i,j}$; and

S_i - $S_i^{'}$ represents the dissimilar part of the first text.

11. The method of claim 7, wherein the semantic overlay is calculated based on Equation (2):

$$\min(\left\| T_i - \sum_{j=1}^{m} \beta_{i,j} \times S_j \right\|^2 + \lambda \sum_{j=1}^{m} \beta_{i,j}{}^2) \quad (2);$$

wherein T_i is a d×1 column vector of the second text; S_j is a d×1 column vector of the first text; $\beta_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number.

12. The method of claim 11, wherein the similar part of the second text is calculated based on $T_j^{'} = \sum_{j=1}^{m} A_{i,j} \times S_j$;

wherein $T_j^{'}$ represents the similar part of the second text, and $A_{i,j}$ is a matrix of $\beta_{i,j}$; and

$T_j - T_j^{'}$ represents the dissimilar part of the second text.

13. The method of any one of claims 1 to 12, wherein the relevancy between the first text and the second text is calculated using a Recurrent Neural Network using the similar part and the dissimilar part between the first text and the second text as input.

14. The method of claim 13, wherein the input comprises:

a similar part and a dissimilar part of the first text; and

a similar part and a dissimilar part of the second text.

15. The method of claim 13, further comprising training the Recurrent Neural Network using a sample data represented as (Ss, Ts, L), wherein Ss stands for a first sample text, Ts stands for a second sample text, and L represents a sample relevancy.

16. The method of claim 15, wherein training the Recurrent Neural Network further comprises:

assigning a first value of granularity to a first relevancy in a first range and a second value of granularity to a second relevancy in a second range.

17. An apparatus for calculating relevancy between a first text and a second text, comprising:

a memory; and

one or more processors;

wherein the memory and the one or more processors are connected with each other;

the memory stores computer-executable instructions for controlling the one or more processors to:

respectively map the first text and the second text to vectors;

determine a similar part and a dissimilar part between the first text and the second text based on the vectors; and

calculate the relevancy between the first text and the second text using both the similar part and the dissimilar part.

18. The apparatus of claim 17, wherein the memory stores computer-executable instructions for controlling the one or more processors to:

determine a similar part and a dissimilar part of the first text; and

determine a similar part and a dissimilar part of the second text.

19. The apparatus of claim 17, wherein the memory further stores computer-executable instructions for controlling the one or more processors to, subsequent to respectively mapping the first text and the second text to vectors, perform dimension reduction on the vectors, thereby representing the first text and the second text respectively using low-dimension dense vectors.

20. The apparatus of claim 19, wherein dimension reduction is performed using one or more methods selected from a group consisting of Word2Vec, Sentence2Vec, and Doc2Vec.

21. The apparatus of any one of claims 17 to 20, wherein the memory further stores computer-executable instructions for controlling the one or more processors to perform a semantic content analysis based on the first text and the second text thereby determining the similar part and dissimilar part.

22. The apparatus of claim 21, wherein the memory further stores computer-executable instructions for controlling the one or more processors to perform a sematic matching process between the first text and the second text thereby obtaining a sematic overlay between the first text and the second text.

23. The apparatus of claim 22, wherein the memory further stores computer-executable instructions for controlling the one or more processors to reconstruct a vector of a word in the first text using word vectors of the second text, and calculate a semantic overlay based on a result of reconstruction.

24. The apparatus of any one of claims 17 to 23, wherein the memory further stores computer-executable instructions for controlling the one or more processors to perform a semantics decomposition on the first text and the second text thereby determining the similar part and dissimilar part.

25. The apparatus of claim 23, wherein the semantic overlay is calculated based on Equation (1):

$$\min\left(\left\| S_i - \sum_{j=1}^{m} \alpha_{i,j} \times T_j \right\|^2 + \lambda \sum_{j=1}^{m} \alpha_{i,j}^2\right) \qquad (1);$$

wherein S_i is a d×1 column vector of the first text; T_j is a d×1 column vector of the second text; $\alpha_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number.

26. The apparatus of claim 24, wherein the similar part of the first text is calculated based on $S_i^{'} = \sum_{j=1}^{m} A_{i,j} \times T_j$;

wherein $S_i^{'}$ represents the similar part of the first text, and $A_{i,j}$ is a matrix of $\alpha_{i,j}$; and

S_i - $S_i^{'}$ represents the dissimilar part of the first text.

27. The apparatus of claim 23, wherein the semantic overlay is calculated based on Equation (2):

$$\min(\left\| T_i - \sum_{j=1}^{m} \beta_{i,j} \times S_j \right\|^2 + \lambda \sum_{j=1}^{m} \beta_{i,j}^{\ 2}) \quad (2);$$

wherein T_i is a d×1 column vector of the second text; S_j is a d×1 column vector of the first text; $\beta_{i,j}$ is parameter for semantic overlay calculation; $\lambda > 0$; and λ is a positive real number.

28. The apparatus of claim 27, wherein the similar part of the second text is calculated based on $T_j^{'} = \sum_{j=1}^{m} A_{i,j} \times S_j$;

wherein $T_j^{'}$ represents the similar part of the second text, and $A_{i,j}$ is a matrix of $\beta_{i,j}$; and

$T_j - T_j^{'}$ represents the dissimilar part of the second text.

29. The apparatus of any one of claims 17 to 28, further comprising a Recurrent Neural Network using the similar part and the dissimilar part of the first text and the second text as input to calculate the relevancy between the first text and the second text.

30. The apparatus of claim 29, the input comprises:

a similar part and a dissimilar part of the first text; and

a similar part and a dissimilar part of the second text.

31. The apparatus of claim 29, wherein the memory further stores computer-executable instructions for controlling the one or more processors to train the Recurrent Neural Network using a sample data represented as (Ss, Ts, L), wherein Ss stands for a first sample text, Ts stands for a second sample text, and L represents a sample relevancy.

32. The apparatus of claim 31, wherein the memory further stores computer-executable instructions for controlling the one or more processors to assign a first value of granularity to a first relevancy in a first range and a second value of granularity to a second relevancy in a second range.

33. A data query apparatus, comprising the apparatus for calculating relevancy of any one of claims 17 to 32.

34. A data query method, comprising calculating relevancy between a first text and a second text according to the method of any one of claims 1 to 16, the second text being a text selected from a plurality of potential target texts;

ranking the plurality of potential target texts based on calculation of relevancy between the plurality of potential target texts and the first text; and

selecting a target text out of the plurality of potential target texts based on a resulting of ranking the plurality of potential target texts.

35. A non-transitory computer-readable storage medium storing computer-readable instructions, the computer-readable instructions being executable by a processor to cause the processor to perform:

respectively map a first text and a second text to vectors;

determine a similar part and a dissimilar part between the first text and the second text based on the vectors; and

calculate a relevancy between a first text and a second text using both the similar part and the dissimilar part.

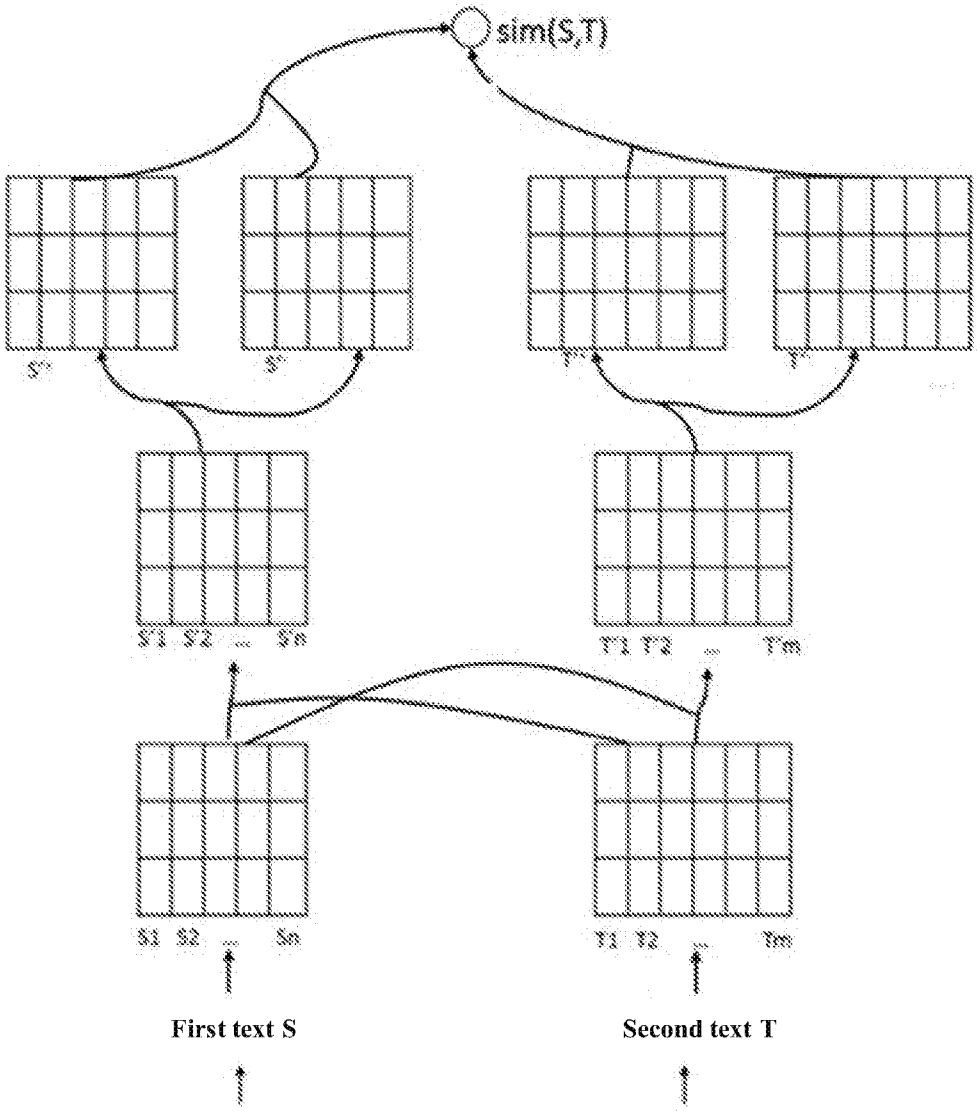

sim(S,T)
S1 S2 ... Sn
T1 T2 ... Tm
S1 S2 ... Sn
T1 T2 ... Tm
First text S
Second text T

FIG. 1

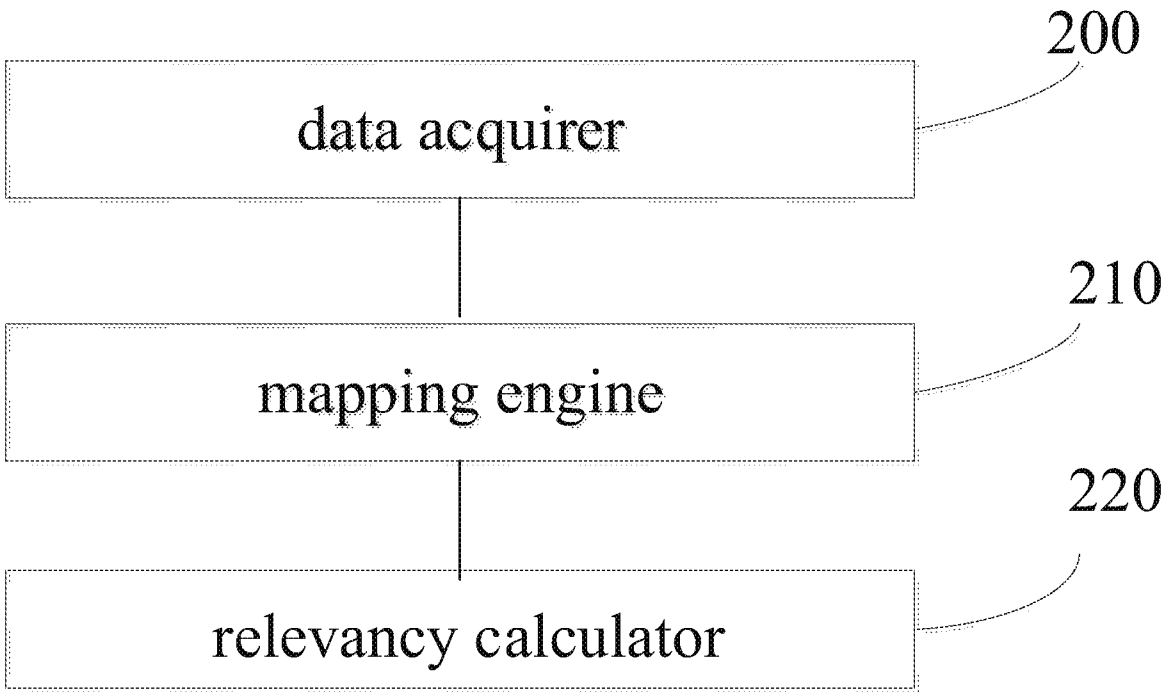

200
data acquirer
210
mapping engine
220
relevancy calculator

FIG. 2

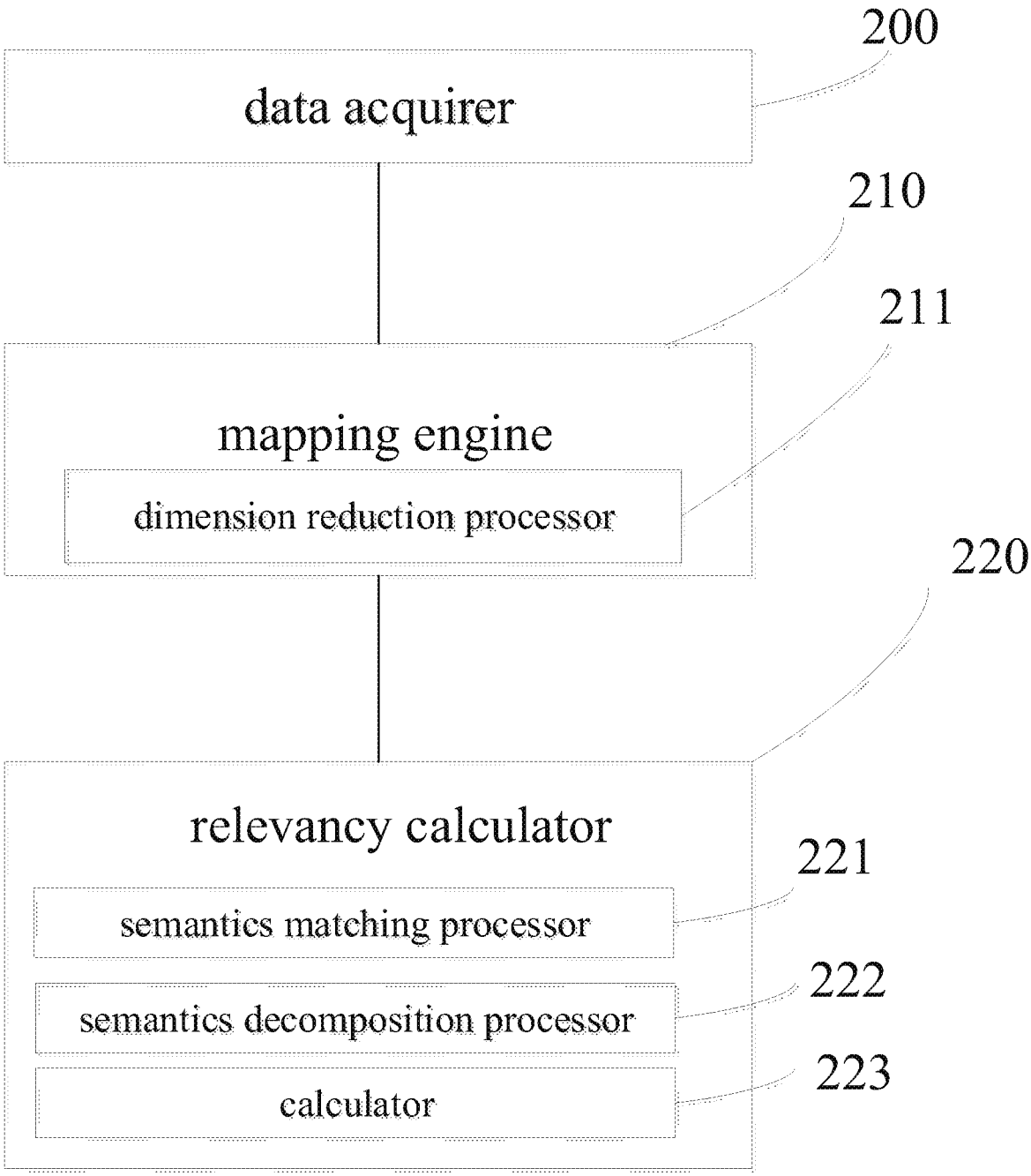

200
data acquirer
210
211
mapping engine
dimension reduction processor
220
relevancy calculator
221
semantics matching processor
222
semantics decomposition processor
223
calculator

FIG. 3

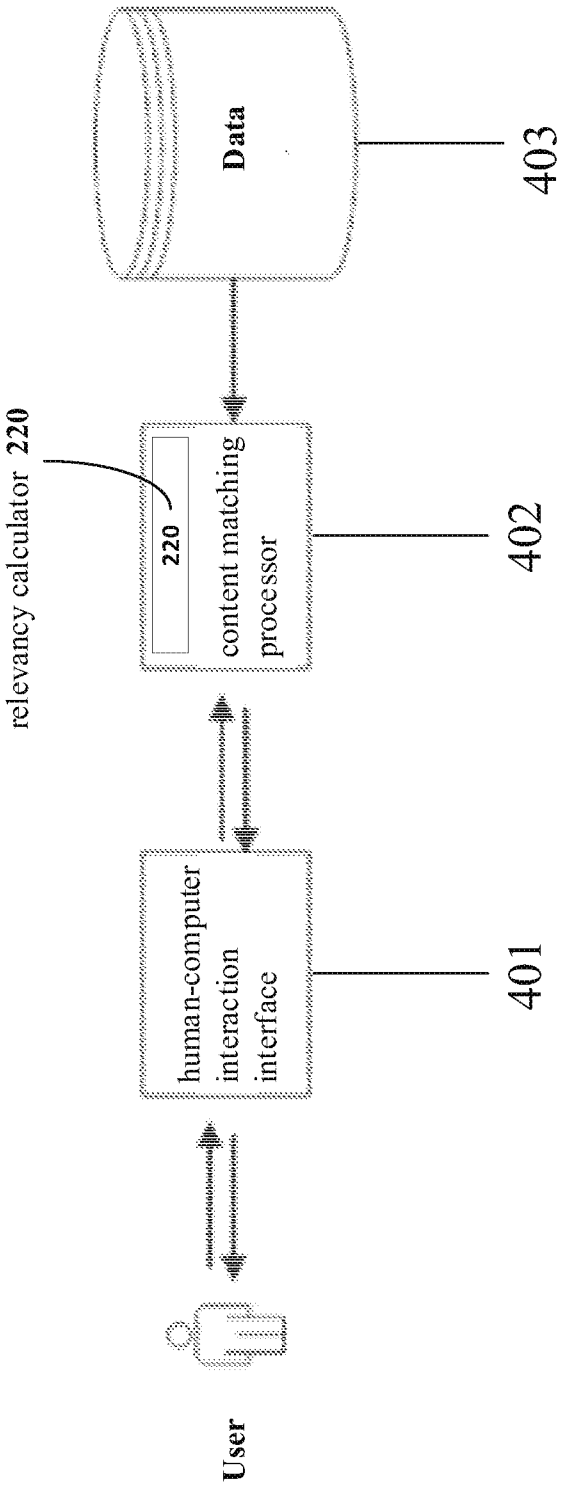

User
human-computer interaction interface
401
content matching processor
220
relevancy calculator 220
402
Data
403

FIG. 4

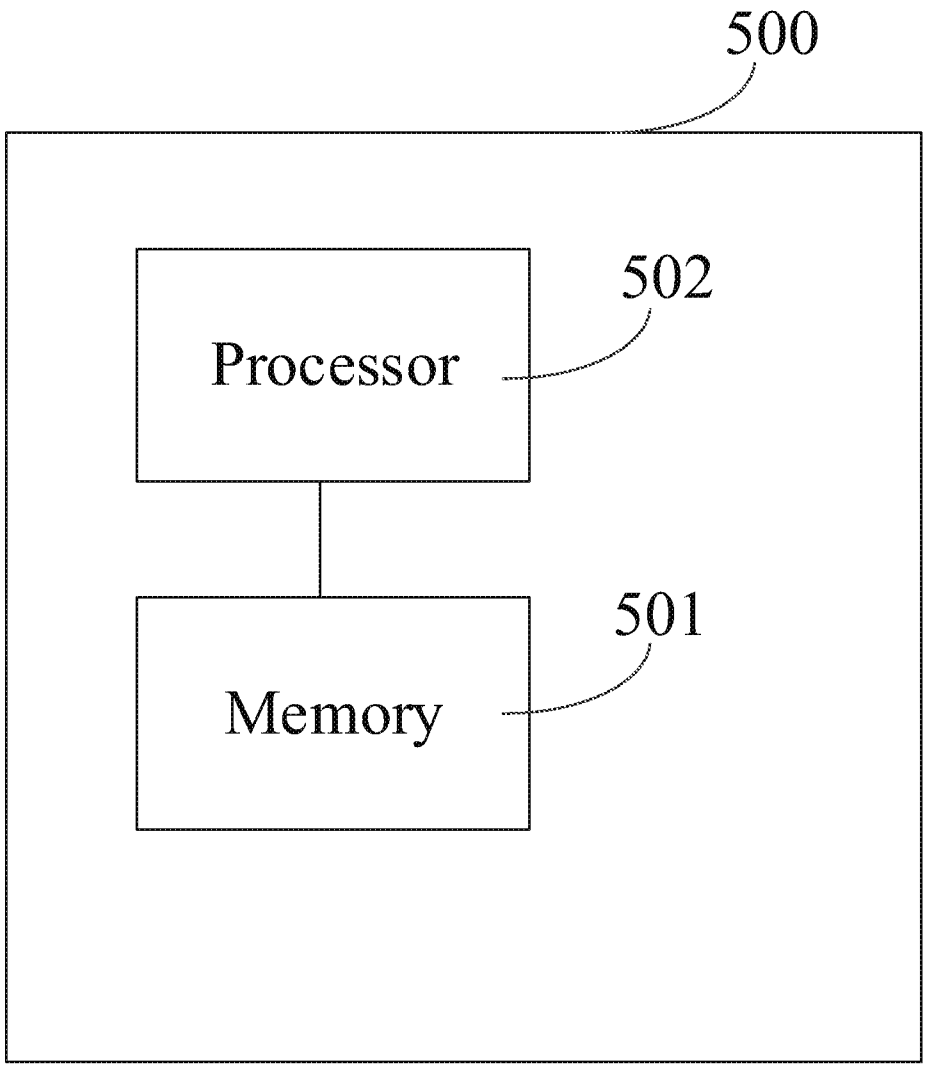


FIG. 5

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2018/091822

A. CLASSIFICATION OF SUBJECT MATTER

G06F 17/21(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, CNPAT, CNKI, IEEE: text, relevancy, dissimilar, similar, vector, semantic, match, overlay, dimension reduction

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
PX	CN 108170684 A (BOE TECHNOLOGY GROUP CO., LTD.) 15 June 2018 (2018-06-15) claims 1-21, description paragraphs [0055]-[0133]	1-35
A	CN 107220220 A (SONY CORPORATION) 29 September 2017 (2017-09-29) description paragraphs [0015]-[0029], [0044]-[0058], [0067]-[0084]	1-35
A	CN 107133202 A (BEIJING BAIDU NETCOM SCIENCE TECHNOLOGY CO., LTD.) 05 September 2017 (2017-09-05) the whole document	1-35
A	CN 107491547 A (BEIJING BAIDU NETCOM SCIENCE TECHNOLOGY CO., LTD.) 19 December 2017 (2017-12-19) the whole document	1-35
A	US 2014249799 A1 (MICROSOFT CORPORATION) 04 September 2014 (2014-09-04) the whole document	1-35

☐ 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	Date of mailing of the international search report
11 October 2018	**18 October 2018**

Name and mailing address of the ISA/CN	Authorized officer
STATE INTELLECTUAL PROPERTY OFFICE OF THE P.R.CHINA **6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing 100088** **China**	**GUO,Minghua**
Facsimile No. **(86-10)62019451**	Telephone No. **86-(10)-53961414**

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2018/091822

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
CN	108170684	A	15 June 2018	None			
CN	107220220	A	29 September 2017	CN	108475262	A	31 August 2018
				WO	2017162134	A1	28 September 2017
CN	107133202	A	05 September 2017	None			
CN	107491547	A	19 December 2017	None			
US	2014249799	A1	04 September 2014	WO	2014137854	A2	12 September 2014