Computers in Biology and Medicine 101 (2018) 39–50

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/compbiomed

Exploiting the semantic graph for the representation and retrieval of medical documents

Qing Zhao, Yangyang Kang[1], Jianqiang Li*, Dan Wang

Faculty of Information Technology, Beijing University of Technology, Beijing, China

ARTICLE INFO

Keywords:
Semantic information retrieval
Medical search
Document ranking
Electronic medical records

ABSTRACT

Objective: The objective of this study was to propose a graph-based semantic search approach by addressing the inherent complexity and ambiguity of medical terminology in queries and clinical text for enhanced medical information retrieval.

Methods: The supportive use of a medical domain ontology exploits the light-weight semantics discovered from queries and documents for enhanced document ranking. First, the implicit information regarding concepts and the relations between them is discovered in the documents and queries and is used to evaluate the relevance of the query-document; then, the semantic linkages between concepts distributed in target documents and reference documents are built and used to score the document's popularity; finally, the above two evaluations are integrated to produce the final ranking list for document ranking.

Results: Empirical experiments are conducted on two different datasets. The results demonstrate that the proposed graph-based approach significantly outperforms the baselines. For example, the average performance improvement on two datasets of the best variant of GSRM compared to the best baseline achieve 7.2% and 7.9% in terms of *P@20* and *NDCG@20*, respectively, which illustrates the effectiveness of the proposed approach.

1. Introduction

With the wide adoption of Electronic Medical Records (EMRs) systems and the continued increase in medical documentation being provided online, medical information retrieval is becoming a hot research topic since it is critical for enabling users to find useful patient information rapidly and effectively in large medical and clinical dataset [1].

Traditional information retrieval models mainly use two factors for document ranking, query-document relevance and document popularity. Query-document relevance measures how well a document matches the query submitted by the user, which is important for achieving a good search result since acquiring relevant results is the users' basic requirement. Many methods, such as vector space models [2–4], probabilistic rank models [7–9], and learning-based rank models [10–12,15], support relevance computing. Document popularity acts as a complementary factor that is also significant for ranking because there are usually so many documents that match the query that it becomes impossible for users to review them all; therefore, only the popular (important or authoritative) documents will be reviewed. The representative approaches comprise PageRank [5], HITS [6] and others.

However, until now, it has been quite clear that the traditional information retrieval technologies perform poorly when they are employed in healthcare. This finding is mainly caused by the complex and inherent ambiguity of the data or information in the medical domain. There are some characteristics in medical information retrieval that determine the following [1,16,18,19,24]: (1) A query's expression is fuzzy when the information inquirers are parents or non-medical professional users; (2) Query terms (such as anatomy and morphology) are ambiguous by themselves because most users have little medical knowledge. For example, a user feels pain in her abdomen, and as a result, she/he submits a query about 'pain in the abdomen'. In this case, the term 'pain' is ambiguous, which could mean 'stabbing pain', 'distending pain', 'labour pain', and so on. In another example, a user entered a query ‘eye infection’, which also has two meanings, ‘bacterial eye infections’ and ‘fungal eye infections’. Thus, finding an appropriate ranking approach to use in a medical search is a central problem.

In this paper, we propose a Graph-based Semantic Ranking Model (GSRM) for practical medical searches. Basically, our method uses domain knowledge as support to exploit the lightweight semantics

* Corresponding author.
E-mail addresses: zhaoqing1025@emails.bjut.edu.cn (Q. Zhao), kangyangemail@163.com (Y. Kang), lijianqiang@bjut.edu.cn (J. Li), wangdan@bjut.edu.cn (D. Wang).
[1] Co-first author.

https://doi.org/10.1016/j.compbiomed.2018.08.009
Received 10 May 2018; Received in revised form 1 August 2018; Accepted 5 August 2018

discovered from the query and documents, to enhance the document ranking. Our model is different from other ranking strategies because GSRM not only uses concepts defined in an ontology but also mines the implicit relations between them from the document queries, employing them for document understanding and relevance computing. In addition, without using hyperlinks, our model develops a novel method to compute the document's popularity, which will be used in conjunction with the document relevance to further improve the accuracy of the ranking results. We incorporate an open dataset and an internal dataset to demonstrate the performance of the GSRM. The experimental results clearly show that compared with the existing models, our proposed novel ranking approach GSRM performs better in terms of the ranking accuracy.

The remainder of this study is organized as follows: Section 2 summarizes the related works. Section 3 introduces details about our semantic-based synthetic rank model. The experiments are reported in Section 4, where the setup of the experiment is illustrated and the test results are discussed. Section 5 discusses the previous study and the similar wok in the literature, and the limitations of this study. Section 6 concludes with a summary of our research and directions for future work.

2. Related work

Traditional Information Retrieval (IR) strategies employ statistics for words in document text to calculate query-document relevance on which to base the rank. For example, the classic vector space model [2] constructs vectors based on term statistical information to represent documents and queries, and it calculates the vector, including the angle cosine, to evaluate the document relevance; probabilistic rank models [7–9] exploit the term distribution probability in a document set to estimate the document relevancy probability with which to decide the ranking list; the learning-based rank models [10–12] employ machine learning methods to construct ranking functions from training data, which help to compute the document relevance.

Later, as the web matured, an enormous number of medical documents of varying quality in traditional IR became available, and researchers found that the classic relevance-only retrieval strategies performed poorly in this environment [5,6,17]; thus, the document importance was introduced for the IR ranking. The earliest work was Google's PageRank algorithm, which computes the page importance by analysing the hyperlink structure of the web, with the assumption that the larger the number of recommendation hyperlinks a web page has, the more authoritative it is [5]. Another representative work is the Hyperlink-Induced Topic Search algorithm [6], which considers not only the authority of a page but also its role as a hub. Additionally, there are many similar methods that have been proposed by various researchers, such as topic-sensitive PageRank [20], Hilltop [22], and stochastic approach for link structure analysis (SALSA) [21], among others.

An ontology is a powerful knowledge representation and reasoning tool that has attracted researchers' attention in recent years [26]. The rich prior knowledge that it describes is considered to be a good candidate for directing and optimizing semantic similarity computing for IR [16,18,19,24]. With the support of standard terminologies or domain ontologies, such as the International Classification of Disease (ICD), Unified Medical Language System (UMLS), and Medical Subject Headings (MeSH) [32], semantic-based IR approaches are widely used for medical information retrieval. For example, the work in Ref. [42] proposes an approach for measuring the semantic similarity between words by using WordNet and other information sources. The vector-space model is suitable for the exploitation of ontological concepts that are recognized from both queries and documents to improve the document ranking [49]. References [25,31] propose approaches to compute the document relevance by determining the semantic relatedness between the words or concepts defined in the corresponding ontologies. By abstracting free-text content into semantic graphs, several papers [27,28,34,40] report work on using graph matching for document ranking. Considering queries as concepts and documents as instances, ontological reasoning is adopted to estimate the document relevance [33]. The work in Ref. [30] is similar to our work, which utilizes semantic relations between concepts for query-page matching, but it relies on the premise that a document has a complete semantic annotation graph that describes its content, which is only satisfied by annotated pages in Semantic Web, and not by documents or EMRs in the medical domain.

Complementary to the existing work, the contribution of this paper is as follows: (1) With the support of an ontology that describes the background knowledge on the medical domain, a novel approach to compute the relevance between queries and documents is proposed, where lightweight semantics (i.e. concepts and the relations between them) identified in queries and documents are exploited for calculating query-dependent scores; (2) Through building the semantic linkages between documents from two different sources, we propose an approach to calculate the document popularity for a medical search. (3) The evaluations derived from the query-document relevance and document popularity are combined for the document ranking, on which basis the experiments on the real-world datasets are reported. The present empirical study clearly demonstrates the effectiveness of the proposed approaches for medical information retrieval.

3. Graph-based semantic rank model

As shown in Fig. 1, the pipeline of the GSRM approach contains three modules, i.e., (1) query-document relevance evaluation; (2) document popularity calculation; and (3) synthetic document ranking. In the following, we will introduce each of those modules in detail.

3.1. Query-document relevance evaluation

Semantic relations among concepts matching the user's intention imply important information and are crucial for medical document ranking. It is intuitive that the more semantic relations between the keywords in a user's query and the concepts a document describes, the higher the probability the document satisfies the user's query. Taking the 'pain in the abdomen' as an example, if document A includes both 'stabbing pain' and 'distending pain' in the abdomen, whereas document B contains generalized 'pain in the abdomen', it is reasonable that we presume that document A is more relevant to the user's query than document B.

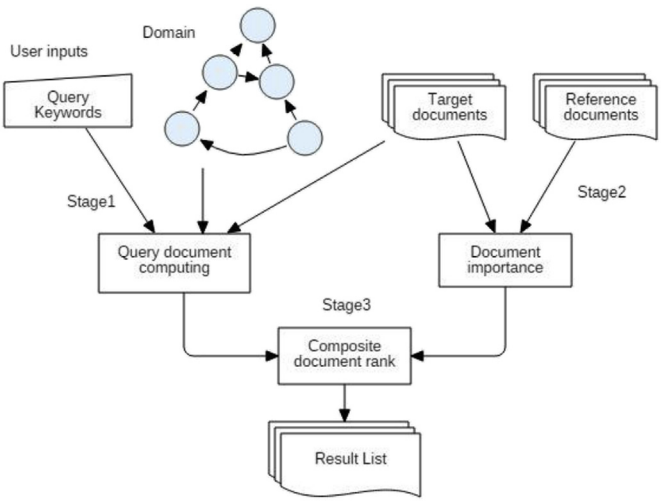


Fig. 1. GSRM architecture.

Given a query q, this module evaluates the relevance between q and each document d in the candidate medical document sets by comparing their contained semantic concepts and relations based on the domain ontology.

The relevance between q and d includes two parts, i.e., the relation-based relevance $score_R$ and the concept-based relevance $score_C$. This section first describes the process for calculating $score_R$ and then the computation of $score_C$, as well as their combination, to decide the final ranking of document d.

As illustrated in Fig. 2, the process of evaluating the query-document relevance includes three steps:

(1) Query intention understanding: The keywords that are contained in query q are mapped to the candidate concepts defined in the medical ontology. Then, a set of concepts is identified to represents the user's information needs. To conjecture all of the possible relations among the identified concepts that the user could have in mind, a set of *query semantic graphs* is constructed from the ontology based on the identified concepts, in which each *graph* represents a possible query intention. For example, 'eye infection' might refer to 'bacterial eye infections' or 'fungal eye infections', two graphs should be constructed for the query 'eye infection'.
(2) Document semantics understanding: This step recognizes the set of concepts that are mentioned in the medical document. Then, by utilizing the relations between the concepts defined in the medical ontology, a set of *document semantic graphs* is constructed for each document. This set of *graphs* can be deemed as the interpretation of the document content.
(3) Query-document relevance calculation: This calculation considers the following intuition the larger the number of semantic relation combinations that are covered by a document, the more likely it is that the real intention of the user is covered, and the more relevant the document is to the query. This step matches the *query semantic graph set* with the *document semantic graph set* to compute $score_R$. After the calculation of the concept-based relevance $score_C$, it is combined together with $score_R$ to obtain the final ranking of each document d.

More details of each step are given in the following sub-sections.

3.1.1. Query intention recognition

The goal of this step is to obtain a set of sub-graphs in the medical ontology to represent the user's intention. Its implementation mainly includes the following: (1) candidate concept mapping; (2) target concept set identification; and (3) intention graph construction, as described below:

(1) *Candidate concept mapping*: given a query $q = \{k_1, k_2, \ldots, k_m\}$, a set of candidate concepts $SC_i = \{C_{i1}, C_{i2}, \ldots, C_{ini}\}$ for each keyword k_i $(1 \le i \le m)$ is found in the medical ontology by fuzzy matching [13]. For each candidate concept C_{ij}, there is a score r_{ij} that denotes the confidence that the keyword k_i truly refers to C_{ij}. Since each k_i could be mapped to multiple concepts, there are multiple concept combinations that are implied by a given query, where each concept combination corresponds to a concept cluster in the ontology. Assuming that each k_i represents a concept, there should be an ideal concept combination for each given query that truly represents the user's query intention.
(2) *Target concept combination identification*: This component attempts to understand the user's information needs by identifying the ideal concept combination. As illustrated in Fig. 3, each concept combination corresponds to a cluster in the medical ontology, where each red node denotes a candidate concept, and each red ellipse represents a candidate concept combination. Intuitively, when a user uses the query to describe a set of concepts to represent the information needed, these concepts should be correlated closely with one another. Therefore, the intuition that the graph with a more compact structure has a higher possibility of denoting the user's intention is adopted for the target concept identification. More concretely, we use the concept combination that has the corresponding sub-graph with the minimum number of edges linking these concept nodes together to denote the user's information needs.

The sub-graph construction algorithm in Ref. [14] is adopted here to extract from the medical ontology all of the sub-graphs that contains the minimum number of edges, to connect all of the candidate concepts in a combination together, where the number of edges $W_{g\text{-}min}$ denotes the penalty [15,16,17] of the corresponding sub-graph. Obviously, the set of minimum sub-graphs has the same penalty value $W_{g\text{-}min}$. The set relates all query concepts in the closest way and is considered to be one of the best reflections of the user's mind.

Step (2) of the Target concept combination identification covers the disambiguation functionality. For example, given a query, there are three concept clusters, i.e., ⓐ, ⓑ, and ⓒ, that are obtained from in the ontology (as illustrated Fig. 3), by step (1), Candidate concept identification. Since the concept cluster ⓑ is the most compact, we use its contained concepts to denote the user's information need. In this process, these keywords in the query construct mutually a context for

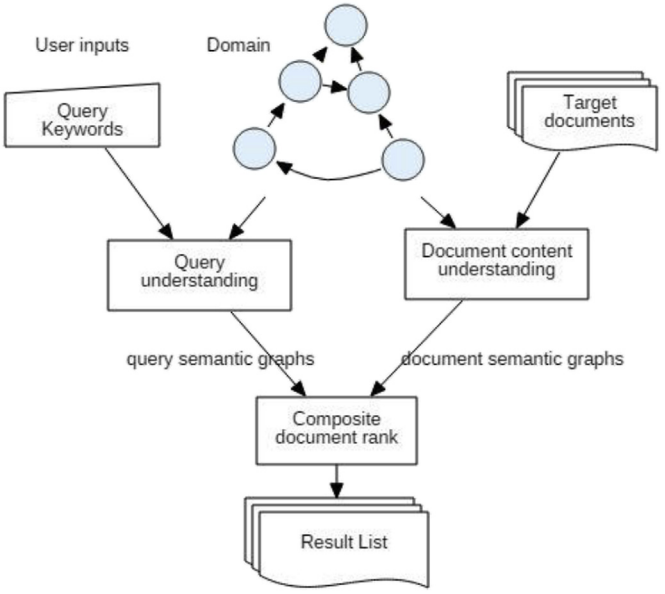


Fig. 2. $Score_R$ calculation process.

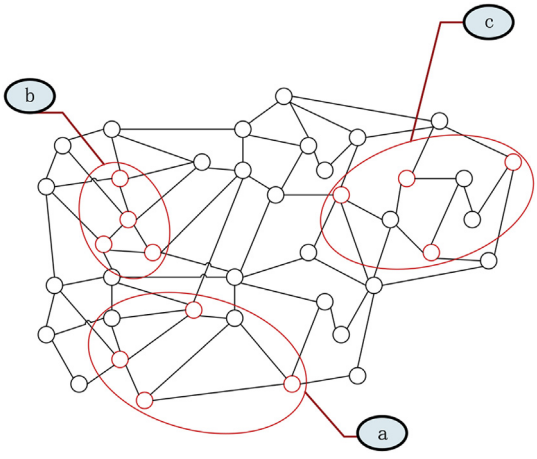


Fig. 3. The schematic diagram of concept mapping and disambiguation.

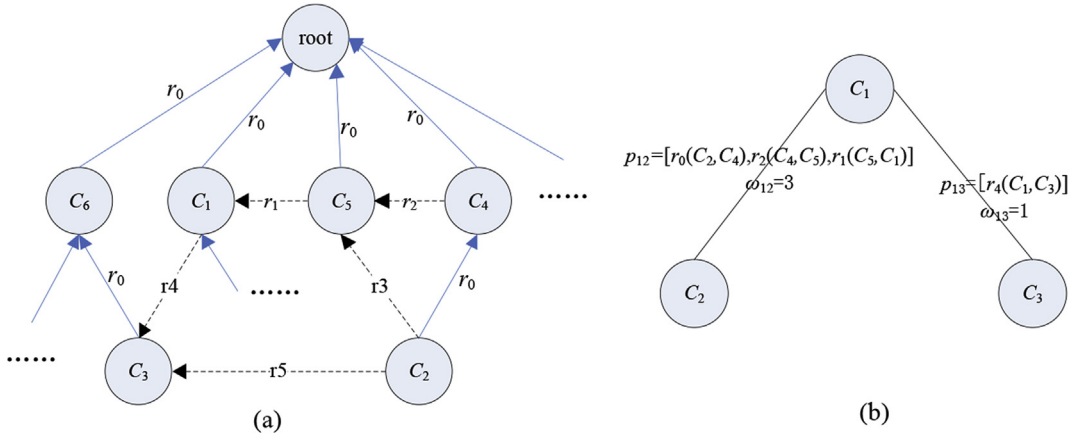


Fig. 4. An example of an intention graph.

realizing the semantic disambiguation.

(3) *Intention graph construction*: Given a target concept combination (C_1, …,C_m), this component uses the medical ontology as the background knowledge to understand the user's intention.

Definition 1. Intention graph: an *intention graph* $G_q(V_q, E_q)$ is a tree, where V_q is a sub-set of vertices that denote the concepts in (C_1, …,C_m) in the ontology, E_q is a sub-set of edges in the ontology, and an edge represents a semantic path that links C_i with C_j in the ontology. For each edge $e(C_i,C_j)\in E_q$, there are two parameters that are associated with it, i.e., p_{ij} and ω_{ij}, where p_{ij} denotes a semantic path that connects C_i and C_j in ontology O, and ω_{ij} represents the importance of the edge, and it denotes the length of the semantic path p_{ij}.

Fig. 4 shows an example of the intention graph, of which (a) is a sample medical ontology and (b) is a sample intention graph, where the concept combination is (C_1,C_2,C_3); for edge $e(C_1,C_2)$, the two parameters are path p_{12} = [r_0(C_2,C_4),r_2(C_4,C_5),r_1(C_5,C_1)] and its length ω_{12} = 3; for edge $e(C_1,C_3)$, p_{13} = [r_4(C_1,C_3)] and ω_{13} = 1.

As shown in Fig. 4, there might be multiple intention graphs for a target concept combination (C_1, …,C_m). A semantic path is an especially good candidate for representing the relations between the concepts. The longer the length is of a path that links C_i to C_j, the weaker the semantic relationship that it indicates between C_i and C_j. For an intention graph, the larger the sum of its edge weights is, the weaker the query concepts' semantic relations, and the less likely it is to represent the user's query intention.

Algorithm 1. Intention graph construction

Input: query concept set S_{query}={C_1,C_2,…,C_k}, $W_{g\text{-}min,}$ Ontology O

Output: a set of intention graphs S_{intent}={(G,η)}

Begin

1. Setting $G_{all}(V_{all},E_{all})$ as a weighted graph, V_{all} =Φ, E_{all} =Φ.

   /*each edge $e\in E'$ has two parameters, ω and p; ω denotes the edge importance and p represents the semantic path in the ontology that corresponds to e.*/

2. V_{all} =S_{query}={C_1, C_2, …, C_k};
3. For (int i=1; i<k; i++)
4. For (int j=i+1; j≤k; j++)
5. {
6. E_{all} = E_{all} ∪ {e_{ij}(p_{ij}=*path*(C_i,C_j), ω_{ij}=*length*(*path*(C_i,C_j))) | *path*(C_i,C_j) is a semantic path that links C_i and C_j in the ontology && *length*(*path*(C_i,C_j))≤ $W_{g\text{-}min}$ −k+2+d && S_{path}∩(S_{query}-{C_i, C_j})=Φ};
7. }

   /*k denotes the number of concepts in S_{query}; d is an optimizing parameter that constrains restrains which path should be chosen and mapped to an edge in E_{all}, to control the graph scale (the method for choosing d will be introduced in section 4.3); S_{path} is the set of concepts that lie on the path, path(C_i,C_j).*/

8. S'={ (G_{st}, W_{st}) | G_{st} is a spanning tree of G_{all} && (W_{st}= $\sum_{e_i\in G_{st}} \omega_i$)≤$W_{g\text{-}min}$+$d$ };

   /* refer to (Yang 1983) for the spanning tree generation algorithm. */

9. For each (G_{st}, W_{st}) ∈S'
10. {
11. S_{intent}=S_{intent} ∪ (G_{st}, η_{st} =$1/W_{st}/\sum_{i=1}^{|S'|} 1/W_i$);
12. }
13. Return S_{intent};

End

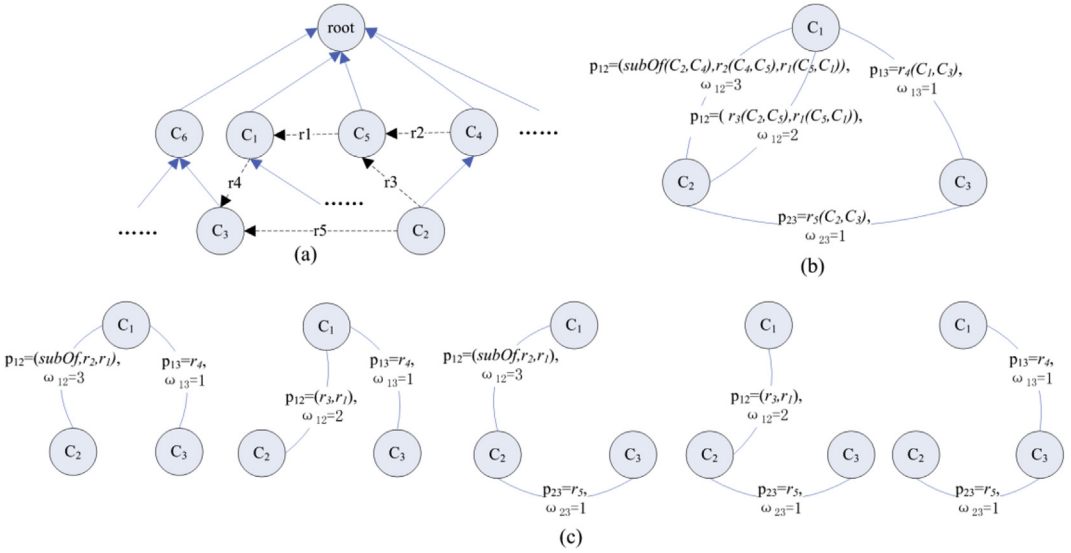


Fig. 5. (a). A sample domain ontology (b). G_{all} extracted from (a) with C_1, C_2 and C_3 as query concepts (c). S_{intent} acquired based on G_{all}.

The following is the pseudo code for constructing a set of intention graphs S_{intent} from a target concept combination $S_{query} = \{C_1, C_2, \ldots, C_k\}$ with a penalty value $W_{g\text{-}min}$, where each graph in S_{intent} is assigned a weight η, which denotes the possibility of the intention described by this graph being the user's true intention.

In Lines 1–7, a weighted graph G_{all} that covers all of the *reasonable* semantic relationships between every two query concepts is constructed. Each semantic path which links two query concepts (C_i, C_j) in the ontology with a path length no larger than the threshold ($W_{g\text{-}min}$–$k + 2 + d$) is selected as a *reasonable* semantic relation between C_i and C_j and is abstracted as an edge in G_{all}. Here, d and $W_{g\text{-}min}$ codetermine whether a semantic path or an intention graph (described in Line 8) is reasonable or not, and a well-chosen d will save running time without losing accuracy in the query interpretation. The experiment that we adopted to determine d will be introduced in Section 4.3. In Line 8, a set of intention graphs is constructed based on G_{all}. Speaking concretely, each spanning tree (it relates all of the query concepts together by a semantic relation combination) of G_{all}, with its graph weight (the sum of its edge weights) no larger than the threshold $W_{g\text{-}min} + d$, is chosen as an intention graph. The underlying consideration for why we chose $W_{g\text{-}min} + d$ to constrain the graph selection is that there is little possibility for a graph whose graph weight (W_{st}) is much larger than $W_{g\text{-}min}$ to represent the actual user's intention. In Lines 9–12, the intention graph weights are normalized to values in [0, 1]. Finally, S_{intent} will be returned in Line 13. Fig. 5 illustrates an example on the intention graph construction.

3.1.2. Document semantics understanding

Definition 2. Document semantic graph: a *document semantic graph* $G_d(V_d, E_d)$ is a weighted connected graph, where V_d is the set of nodes that represents all concepts contained in the intersection of the medical document concept set and the query concept set, and each edge in E_d denotes the semantic path that links two concepts in V_d together, i.e., $e(C_i, C_j) \in E_d$ in the ontology ($C_i, C_j \in V_d$). There are two parameters, λ_{ij} and p'_{ij} for each edge, where λ_{ij} denotes the length and then the importance of the edge, and p'_{ij} is a parameter that records all of the concepts that appear is the path.

This step constructs a set of document semantic graphs, $S_d = \{(G_d, \eta_d) | \ 0 \le \eta_d \le 1\}$, for a document, to imitate all possible query intentions covered by it. To insure the run time efficiency, we construct S_d offline before the ranking process. Let C_{doc} be the set of concepts identified from a document. Because V_d of a document semantic graph is query-relevant and we have no knowledge about what V_d contains before a query is acquired, we choose to construct a document semantic graph set for each subset, C_{sub}, of C_{doc} in advance. When a query arrives, S_d can be acquired by matching query concepts with C_{sub}. The detailed process is as follows.

① Constructing a document semantic graph set S_{sub} for each C_{sub}. ($C_{sub} \subseteq C_{doc}$, $2 \le |C_{sub}| \le d$, d is set to 5 in our implementation). The way to build S_{sub} is almost the same as that of building S_q for C_q. The only difference is that when finding semantic paths between concepts, we ensure that all concepts in the semantic paths must belong to C_{doc}.

② Acquiring S_d based on $\{(C_{sub}, S_{sub})\}$. When a query arrives, we first compare the query concept set C_q with all of the document concept subsets C_{sub}. The C_{sub} that satisfies $C_{sub} = C_{doc} \cap C_q$ is chosen. If $C_{sub} = = C_q$, then its corresponding S_{sub} is chosen as S_d directly, otherwise, S_{sub} should be modified by resetting the graph weight for each graph that it contains (denoted as G_d), which is realized by first finding all of the super graphs of G_d from S_q, denoted as $S' = \{G_q(V_q, E_q) | \ G_q(V_q, E_q) \in S_q \ \&\& \ G_d$ is a sub graph of $G_q\}$, and then changing the graph weight of G_d from η_d to $\min_{G_q \in S'} \eta_q \times ((|V_d| - 1)/(|V_q| - 1))$.

Note that identifying the ontological concepts contained in a document has been widely studied [33,34]. This paper adopts an existing approach proposed by Ref. [34], which not only recognizes but disambiguates concepts to ensure accurate concept identification. The basic idea is to partition the text into chunks and then match each chunk against the ontology, where the weighting of the matched candidate concepts is used for sense disambiguation. A more detailed explanation and examples about the concept mapping and disambiguation are described in Ref. [34]. $\min_{G_q \in S'} \eta_q \times ((|V_d| - 1)/(|V_q| - 1))$

3.1.3. Query-document relevance

After obtaining the intention graph set $S_{intent} = \{(G_q, \eta_q) | \ 0 \le \eta_q \le 1\}$ and the document semantic graph set $S_d = \{(G_d, \eta_d) | \ 0 \le \eta_d \le 1\}$, we use Eq. (1), as follows, to combine S_{intent} and S_d for computing $score_R$.

$$score_R = \frac{\sum_{G_d \in S_d} \eta_d}{\sum_{G_q \in S_q} \eta_q} \tag{1}$$

We also compute the concept-based relevance $score_C$ by applying the vector-space model [2] on the top of the ontology concepts. Let S_{query} be the set of concepts that correspond to the query keywords, and let S_O be

the set of all concepts in the ontology. A query vector is defined as $q = (q_1, \ldots, q_n)$, $n = |S_O|$, and each element in q corresponds to a concept in S_O. Eq. (2) shows how to compute q_i, where C_i is the concept that q_i corresponds to, and $Similarity(a,b)$ computes the concept semantic similarity using the model introduced in Ref. [35]. A document vector is defined as $d = (d_1, \ldots, d_n)$, where d_i is set to 0 if its representing concept C_i does not appear in the document, otherwise, d_i is computed by adaption of the tf-idf algorithm [2] based on the concept occurrence frequency.

$$q_i = \begin{cases} if & C_i \in C_q \ \max_{C_j \in C_q} similarity(C_i, C_q) \\ else & C_i \in C_q \end{cases} \quad (2)$$

Then, the combined relevance score of a document with a query is measured as $r = \delta\ score_R + (1\text{-}\delta)\ score_C$, where $0 \le \delta \le 1$ is used to balance the concept-based score and the relation-based score in the final ranking.

3.2. Document popularity computing

This module utilizes an external document set as the reference to rank the popularity of a target document and contains two steps: (1) Building the "class-instance" association between the target documents and external documents through comparing their content, the aim of which is to find to what extent a target document (i.e., a class) is supported by the external documents (i.e., an instance); (2) Evaluating the document popularity by using the intuition that the more instances a class has, the more popular it is.

The intuitions that the semantic relationship construction is advantageous for evaluating the document popularity could be derived from two aspects:

- The "class-instance" associations reflect the coverage of a document. Considering a usage scenario in which a user wants to learn something about his condition when he feels pain in the abdomen. Many documents will be returned after he inputs a query. If a document has more support information (external document links) about pain (e.g. stabbing pain, distending pain, labour pain) in the abdomen, the document could be more attractive or recommendatory, and it will be more deserving of attention.
- The "class-instance" associations could be used to disambiguate the query implied intentions. Considering the user scenario where a user has a fever that could have resulted from phthisis, influenza, or enteritis, for example, then if a cause has more actual cases (external documents) that support it, this circumstance makes it more likely to be the cause, rather than only the fact that it is a cause of the symptom. For the query 'eye infections', if a document A on 'bacterial eye infections' has more supports in the external document set than that of the document B on 'fungal eye infections', the rank of document A should be higher than that of document B.

3.2.1. Semantic association construction

This step builds the "class-instance" relationship between target documents and external documents.

Since the supervised method [30] for text classifier building requires a large amount of hand-labelled data, and the manual data labelling task is time consuming and labour intensive, we employ the FACT (Fully Automatic Categorization of Text) approach [36] to construct the "class-instance" connections between the target documents and reference documents.

We instantiate the template of the FACT approach as follows: (1) category name understanding, the concepts in the target document are recognized with the support of the medical ontology as an initial keyword set; (2) representative profile generation, using the semantic relationship defined in the medical ontology (e.g., the immediate children of the recognized concepts), to extend the initial keyword set as the representative profile of the corresponding category; (3) initial document labelling, the similarity between a category's representative profile and each external document is computed and used for labelling the document with the probability that the document belongs to the corresponding category; (4) refinement of the initial document labelling, using the clustering results of the external document set to adjust the probability score for labelling; (5) training data construction, according to the rank of the probability score, selecting a set of high confidence documents of this category as positive examples and a set of high confidence documents of other categories as negative examples; (6) classifier building, with the generated trained data, the SVM (Support Vector Machine) [37] is used for building the classifier. Finally, the classifier is obtained to categorize the external documents (i.e. the instances) into the target documents (i.e., the categories).

3.2.2. Document popularity calculation

This step uses the "class-instance" connections between the target document and the external document to calculate the popularity of the target document:

(1) For each target document d, the number of its instances in the external documents set $n(d)$ is obtained.
(2) We set the popularity of the target document d_i with the maximum number max of instance documents to 1, and the popularity score of every other target document d_j $(i \neq j)$ is measured as $score_p(d_j) = n(d_j)/max$.

3.3. Synthetic document ranking

This module combines the query-document relevance r and document popularity $score_p$ to produce the final ranking list of the given query.

With the assumption that for the documents with similar relevance scores to the given query, the more popular a document is, the higher the position that it should have in the ranking list. The following three steps are used to generate the final ranking list:

(1) The query-document relevance r is employed to generate a document list, and then, the returned documents are clustered into k groups by using the query-document relevance score as the only feature.
(2) The average value of the relevance scores of the contained documents in each group is computed, which is used to determine the sequence of these groups.
(3) The document popularity $score_p$ is used to re-rank the documents inside of each group.

Here, k is a factor to balance the query-document relevance r and the document popularity $score_p$. In this study, the improved k-means algorithm [38] was adopted to cluster the obtained documents based on the query-document relevance, where k is not defined in advance but is evaluated in the clustering process based on the data density.

4. Experiment and evaluation

4.1. Data collection and description

Two datasets, i.e., the Real-world Dataset (RD) and the Medical Literature Dataset (MLD), are built for the experimental evaluation of the proposed GRR approach.

4.1.1. Real-world dataset (RD)

Document set: The goal of this research is to achieve the second and meaningful use of Electronic Medical Records, and with the help of one of the cooperating hospitals, we collected 7483 inpatient medical records as target documents to be retrieved. Each medical record

describes the patient's symptoms and the entire course of their medical treatment and diagnosis during their stay in the hospital. Over 400,000 actual medical cases of outpatients acted as the external document set. The EMR data released to us for conducting the experiments were processed with well-controlled data anonymization operations [23], where all of the patient ID related information were removed. Additionally, the experiments were conducted under a closed computing environment, and only authorized users could access the EMR search service.

Medical ontology: MeSH is a widely used thesaurus for semantic medical information retrieval. We use MeSH together with the diagnostic knowledge extracted from Collin's medical textbook (rather than other controlled vocabularies) [43] as the Medical Diagnosis Ontology (MDO), for the semantic analysis of the clinical text. MeSH contains 1526 classes and 9 types of relations that link symptoms and diseases, diseases with diseases, checking items and symptoms, and so on. MDO is used here mainly because the purpose of MeSH is for indexing journal articles and books, which provides great potential in that the defined concepts in MDO are widely used in medical documents.

Sample queries: with the help of medical domain experts, 30 sample queries are defined for experimental evaluations, where the above mentioned ''pain in the abdomen' is a concrete example. In the implementation, we first selected 100 candidate queries about the frequent symptom descriptions that dictate the patient's situations. Then, the entropy-based measure proposed by Demidova, E., et al. [39] was used to estimate the ambiguity of the result. Finally, the top 30 queries with high ambiguity were chosen as sample queries. High ambiguity means that a query has multiple interpretations, which is the main reason why leading traditional search engines produce low accuracy search results.

4.1.2. Medical Literature Dataset (MLD)

Document set: We build the MLD document set by downloading the abstracts of approximately 100,000 medical studies published between 1966 and 2016 from the website:www.ncbi.nlm.nih.gov/pubmed/, where the document selection was chosen randomly from and downloaded as the target document set. Another 200,000 abstracts were also downloaded, which serve as the external document set.

Medical ontology: Considering that the MDO is target medical dataset independent, we built a semantic knowledge base by running a concept recognizer (similar to [45]) on the document dataset from Pubmed (www.ncbi.nlm.nih.gov/pubmed/) to mine the valuable medical concepts and reconstruct the logical hierarchical relationships between these vocabularies by referring to the structure of the ontology MeSH. The resulting ontology, containing 1927 classes and 11 types of relations, is target dataset dependent, i.e., a large number of defined concepts in the ontology come directly from the dataset to be retrieved, and this fact could provide great potential for exploiting the advantages of the proposed semantic search approach.

Sample queries: Similar to the RD dataset, we first selected the 100 most frequent symptom descriptions from medical cases in the medical document database (including the symptom description, clinical manifestation, causes leading to the symptom, and so on). Then, 30 ambiguous descriptions were chosen as sample queries, where the ambiguity is derived mainly from the polysemy phenomenon.

4.2. Evaluation criteria

Two criteria were adopted to for the performance evaluation of the proposed medical search approach:

P@10 (P@20, P@50): This symbology denotes the proportion of relevant documents in the top 10 (20, 50) search results of the ranked list returned from the given query, which is widely used to evaluate the search engines.

NDCG@10 (NDCG@20, NDCG@50): A new evaluation is proposed called Normalized Discount Cumulative Gain (NDCG), which can test the accuracy of the rank models at a finer granularity. To evaluate a ranking list NDCG, considering a document with lower position is less valuable for the user because the user is less likely to examine it. We also use the top 10 (20, 50) relevant documents for the query in the ranked list.

We calculated a ranking list for NDCG at position n by using Eq. (3):

$$N(n) = Z_n \sum_{j=1}^{n} \frac{(2^{r(j)} - 1)}{\log(1 + j)} \tag{3}$$

where Zn is chosen as the normalization constant, and r(j) is the j-th document rating in the ranking list; thus, we can obtain a NDCG score of 1 in the perfect list.

For the performance evaluation, we hired 13 participants that are graduate students who majored in medical data analysis and have the skills of academic reading and writing. To evaluate the precision, the top-50 returned documents of each sample query are given to each participant. For each document, the participant makes a binary decision with regard to whether it is relevant to the given query. If more than half of them give a positive vote to the document, it is evaluated as a matched to the query. For the NDCG, the number of positive votes is used to define the ratings of each document. For example, a vector of the positive vote number of the top-10 documents is [0, 2, 6, 6, 8, 8, 8, 10, 13, 13], and there are 6 ratings that are defined for this list {6, 5, 4, 3, 2, 1, 0}, which correspond to the document with {13, 10, 8, 6, 2, 0} positive votes, respectively.

4.3. Setting the model parameters

Proper model parameter settings can improve the document rank accuracy. The two parameters that should be set in the GSRM are d and δ.

Here, d is a parameter that constrains which spanning trees of $G_{all}(V_{all}, E_{all})$ should be chosen as intention graphs. We performed an experiment on the two test datasets to choose the optimal value of $d = \{0, 1, 2, \ldots\}$. Given a query and a concrete d, a set of spanning trees of G_{all} is produced (a spanning tree represents a query interpretation; the production method was introduced in Section 3.1.2.) and presented to the user, who should answer (*Yes/No*) as to whether the set contains the tree that represents his real query interpretation. This process is conducted on 60 different queries over all of the candidate values of d. We can obtain an answer set for a concrete d, which contains 60 'Yes/No' answers. Then, we applied Eq. (4) on each answer set to calculate its query interpretation hitting accuracy (the proportion of queries from whose interpretations the user finds a satisfactory one), where N_{query} denotes the number of queries whose answers are '*Yes*'. In total, 7 participants attended the experiment, and we set the average of *Rate*(d) produced by different participants as the final value of *Rate*(d). Fig. 6 summarizes the experimental results, from which we find that $d = 2$ is the optimal choice as the *Rate* reaches a stable status from that point, and the query interpretation hitting accuracy shows no obvious improvement when choosing a larger d.

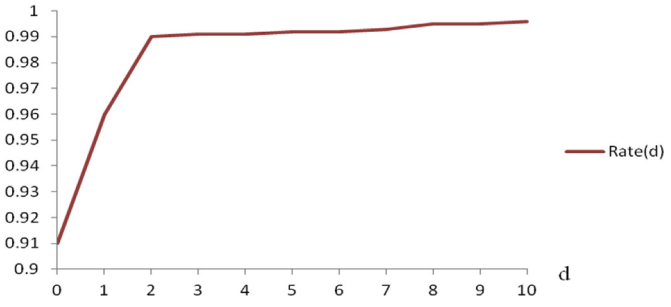


Fig. 6. Rate curve.

Table 1
The best value for parameter δ

Dataset	δ	P@20
RD	0.5	0.79
MLD	0.4	0.67
Average	0.45	–

Note that the selection of δ is a necessary step for the implementation of our proposed approach. Besides the *P@20*, the resulting δ is also used for the performance comparison under the other five criteria. These facts guarantee the fairness of the performance comparison with the existing solutions.

$$Rate(d) = \frac{N_{query}(d)}{60} \quad (4)$$

Here, δ is the regularization parameter, which balances the concept-based score and the relation-based score for calculating the query-document relevance. In our study, we performed the Precision evaluation on the two test datasets to select the optimal parameter from $\delta = \{0, 0.1, 0.2, \ldots, 1\}$. All of the candidate values of δ were tried on each dataset, and the one with the highest *P@20* was chosen (we adopted *P@20* based on the consideration that users are not patient enough to view the long list of results and they prefer to find their satisfactory results in the top-ranked documents.). The results in Table 1 show that the best values for different datasets are similar. To ensure that the GSRM is not biased, the final parameter value is 0.45, which is the average value that we set.

4.4. Experiment results

Three widely used baselines are selected for the comparison study:

(1) *BM25*: By using the famous Vector Space Model (VSM) [41], the documents are ranked according to their BM25 formula based on similarities with the given query.
(2) *Model(P)*: The model proposed by Pablo et al. [29] in which documents are ranked according to a linear combination of the concept-based and keyword-based relevancies, where the concept-based relevance is obtained by adapting the VSM on top of the detected concepts from the query and the document.
(3) *Model(B)*: The model proposed by Brauer et al. [34] that utilizes the VSM on top of the identified concepts in the query and document for the relevance ranking, where semantic relations defined in the ontology are incorporated for concept disambiguation and not explicitly for document ranking.

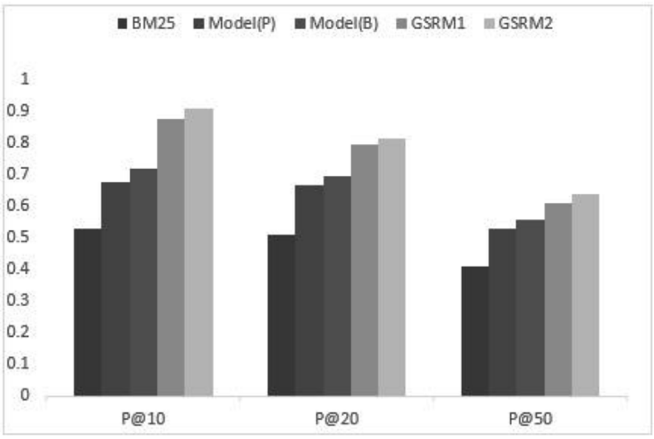
For the comparison study, two versions of the proposed medical approach are implemented:

(1) $GSRM_1$: This version uses only the query-document relevance for ranking.
(2) $GSRM_2$: This version is a full version of GSRM that employs both the query-document relevance and the document popularity for the ranking.

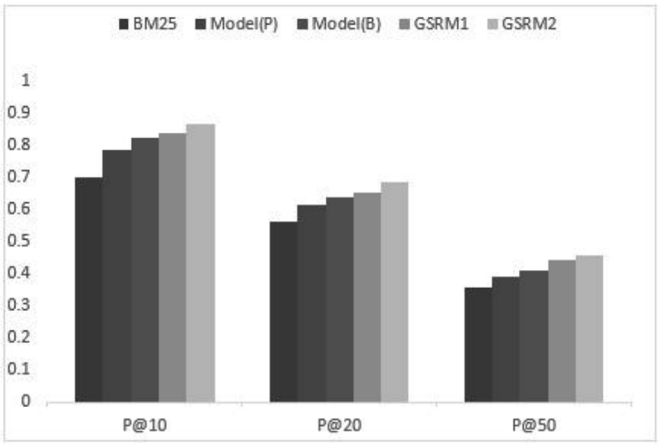
4.4.1. Precision comparison

Fig. 7 illustrates the precision comparison results. We performed testing on the MLD dataset (Fig. 7(a)) and the real-world dataset (Fig. 7(b)), which contains the all-around and complete information about a disease, which is well organized and written.

Fig. 7(a) shows the comparison for the MLD dataset, the concept-based approach (*Model(P)* and *Model(B)*) is much better than the keyword-based approach (*BM25*), and it gains improvements of 15.4% (*Model(P)*) and 18.7% (*Model(B)*) on average, respectively. In addition, it is pleasing to find that our relation incorporated method ($GSRM_1$) further outperformed the concept-based approaches in terms of the rank precision (an improvement of 11.3% on average compared with *Model(B)*), which illustrates that the semantic relations between the concepts is a pivotal factor that cannot be ignored in understanding text content and helping to match queries with documents more accurately. $GSRM_2$ only outperforms $GSRM_1$ by approximately 1.1% on average, which means that the role of document popularity is not as obvious in improving the precision.



(a) Results for MLD



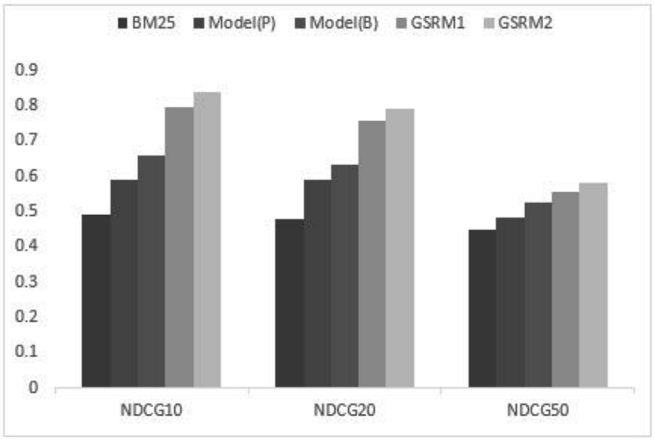
(b) Results for RD

Fig. 7. Precision comparison.

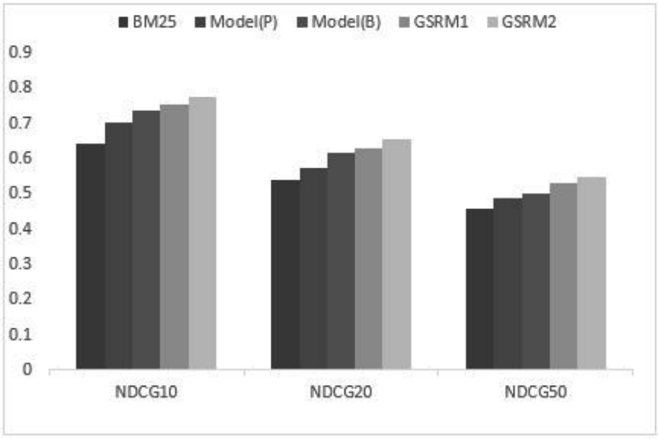
The comparison for RD is summarized in Fig. 7(b). As is observed, the performance of $GSRM_1$ on RD is much worse than on MLD, where the precision is improved only by 1.1% on average compared with *Model(B),* which is the best with the baseline approaches. The main cause of poor performance is that the semantic relations between the concepts in the knowledge base of RD are so simple that only one relation links two concepts together in most cases, which weakens the contribution of $score_R$ in distinguishing documents and finding matching ones. By comparing $GSRM_2$ with $GSRM_1$, we find that the document popularity helps to improve the precision by approximately 2.15% on average, which is a better improvement in the MLD dataset. This finding can be explained by the fact that the actual medical case set is a more reliable and valuable extra knowledge source for symptom diagnosis than the review set is for medical searching.

4.4.2. NDCG comparison

We compared in detail the rank precision of the test models in Fig. 8 with the help of one sample query chosen randomly from each dataset,



(a) Results for MLD



(b) Results for RD

Fig. 8. NDCG comparison.

such as the query 'fever and diarrhoea' for MLD, and the query 'pain in the abdomen' for RD. From the target documents, we chose the top10 (20, 50) documents from the ranking of $GSRM_2$, which were re-ranked by other models or sequenced by NDCG.

We show the ranking performance of *BM25*, *Model(P), Model(B), GSRM1* and *GSRM2* in Fig. 8. Fig. 8 shows that *GSRM1* and *GSRM2* are better than concept-based approaches with NDCG for both the MLD and RD datasets. Taking NDCG as an example in Fig. 8 (a), *GSRM2* outperform *Model(P)* and *Model(B)* by 0.137 and 0.168 for *NDCG@20* in the MLD dataset, and by more than 0.08 and 0.13 for the other NDCG values. For the RD dataset in Fig. 8 (b), the performance of *GSRM2* is as follows: 0.773 for *NDCG@10*, 0.628 for *NDCG@20*, and 0.545 for *NDCG@50*.

Overall, by comparing $GSRM_1$ with $GSRM_2$, it is observed that the document popularity, which is used in conjunction with the document relationships, further improves the accuracy of the ranking results. Taking the experiment on RD as an example, when two symptom records are both relevant to the query, the one that has more actual medical cases that support it is ranked higher, which assumes that the symptom cause occurs more frequently in the real world, and thus, it is more likely that this cause alone led to the symptom.

Table 2
Experiment result of the time complexity on both MLD and RD.

query keywords	query intent understanding (ms)	composite document rank (ms)	total delays of GSRM (ms)
1	–	8.4	9.5
2	1.3	9.5	11.6
3	3.2	10.7	14.5
4	7.6	10.4	18.9
5	15.9	11.8	28.6

4.4.3. Time complexity

The time complexity is a crucial parameter in determining the feasibility of a ranking algorithm. The realization of the GSRM model that we used in the experiments consisted of three main steps: query-document relevance calculation, which comprises three sub-steps; document popularity calculation; and synthetic document ranking. As introduced above in section 3, for the relevance calculating process, the document content interpretation sub-step can be realized offline as well as the popularity calculation process in the GSRM model. Other steps in the query interpretation process, such as $score_R$ and $score_C$ computing, expend little time, and the concepts that correspond to the query keywords (concept mapping sub-step) are indicated by users for which no computing time is consumed. The most time is spent on the query interpretation and the synthetic document ranking processes, whose performances affect the effectiveness of the GSRM directly. When performing query interpretation, most of the time is consumed by the minimum query semantic graph construction, which has a time complexity of $O(nl)$, where n denotes the number of query keywords, and l denotes the number of relations defined in the domain ontology, and the G_{all} graph construction, which has a time complexity of $O(n^2l)$).

All 60 sample queries from both datasets (30 from MLD, 30 from RD) are tested by first separating them into 5 groups according to the number of keywords that a query contains and, then, calculating the average runtime of the queries in a group to make our results unbiased. The computer that we used was a Windows-based PC with a Pentium® D CPU at 2.80 GHz and 1 GB of RAM. The experimental results are provided in Table 2; in conclusion, the promising results (29.3 ms at most) over almost 200,000 documents demonstrate the feasibility of GSRM. Analysing concretely, the increase in the number of query keywords causes the average time consumed in addressing the queries to increase, but it has little influence on the document clustering and ranking process, as column 3 shows. In fact, it is the scale of the target document set that determines the running time of the synthetic rank process. In the next stage of our experiment, we plan to move the GSRM to an even larger document set, and the parallel and distributed computing paradigms will also be sampled.

5. Discussion

Note that, in the above experiments, the FACT approach [36] adopts a binary classification setting, i.e., one medical document might be assigned to multiple categories, for the “class-instance” association building. To evaluate the accuracy of the classification results, we randomly select 600 documents in the reference document set in RD as the test data. We found that on average, each document is classified into 27 categories. We checked each of the category assignments. The percentage of the correct assignments is approximately 83.6%. This result is better than that from three standard datasets (i.e., 20 News Group, WebKB, and Reuters-21578) [44]. We conjecture that the main reason lies in the fact that there are multiple “Entry Term(s)” in the MeSH

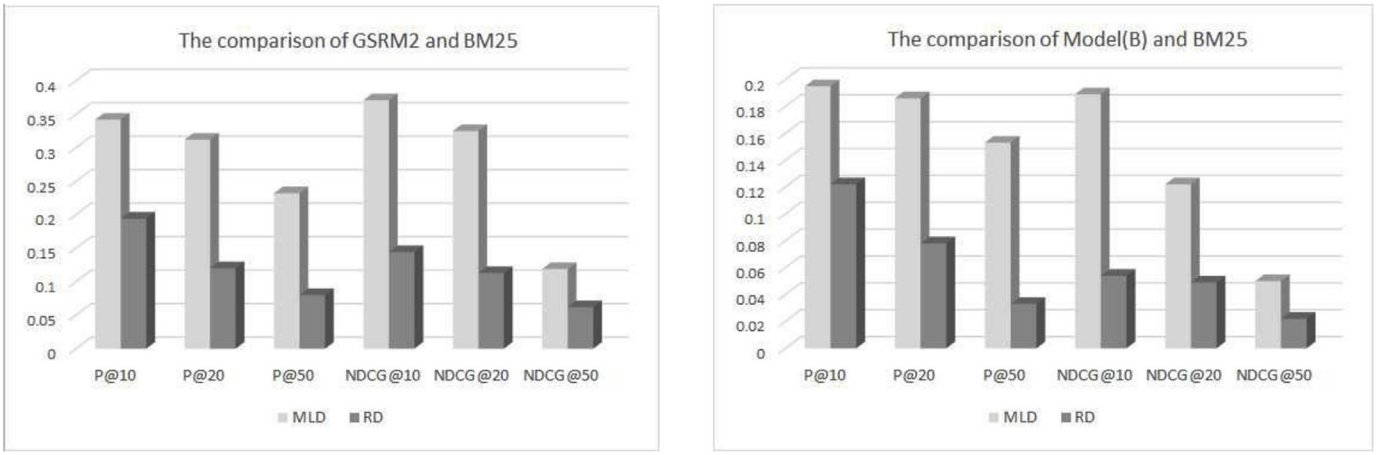


Fig. 9. The comparison of the performance improvement on two datasets.

[32], i.e., the representative profile of the concept contains more texts than the given category names, as in Ref. [36,44]. In fact, we utilized only the statistical results from the large reference dataset, i.e., the number of instances for deciding the popularity ranking of the corresponding concept. The classification error rate in a certain range, for example, 10–20% will not affect the statistics and thus the popularity ranking result.

The main limitation of the proposed approach could be summarized into two aspects:

(1) Its medical search performance relies heavily on the availability of an appropriate ontology, i.e, whether the concepts defined in the ontology are widely used in the target dataset. The adopted MDO in RD is a standard ontology, which is independent of the target medical dataset. The ontology in MLD is built by mining the medical concepts and reconstructing the logical hierarchical relationships between them by referring to the structure of the ontology MeSH, which is target dataset dependent, i.e, a large amount of defined concepts in the ontology come directly from the dataset to be retrieved. As illustrated in Fig. 9 (a), this fact makes the performance improvement of our proposed approach compared to *BM25* [41] on the MLD dataset much larger than that on the RD. In fact, as shown in Fig. 9 (b), the behavior of the best baseline approach *Model(B)* [34] is similar to the proposed approach in this paper, which also demonstrates that a dataset dependent ontology will provide great potential in exploiting the advantages of the semantic medical search approach.

(2) The computing complexity is also a potential problem for our proposed approach. When the query contains many candidate concepts defined in the ontology, the time consuming of the online query understanding might be intolerable, which is a general problem facing the semantic search technologies [14,27,31]. The ontology indexing [28,34] and concept ranking [31] could be a potential solution to speed up the query understanding process. The online k-means clustering is also a time consuming task. Since the goal of most medical search is to find the top relevant/important documents [1,16,45], for its scalability, in the implementation of our proposed approach, we could run the k-means clustering on the top 500 or 1000 returned documents, which might satisfy most of the medical searches.

To exploit the advantages of the proposed approach to the fullest in its real-world application, the medical ontology that is most closely correlated with the dataset to be retrieved would be the best choice. Actually, as indicated by the experiment result in that the performance improvement of our proposed approach compared to *BM25* on the MLD dataset is much larger than that on the RD, we could build an extended version of the available standard ontology by linking the concepts discovered from the dataset to be retrieved, which could ensure that the concepts defined in the ontology are widely used in the target dataset. The reference document set on diseases, symptoms and treatments should truly reflect the popularity of the corresponding concepts in real world clinical operations. For such cases, the quality of the medical search results will benefit substantially from the reference document set.

The described work in this paper is a part of a real-world project that has the goal of achieving the second and meaningful use of Electronic Medical Records [47,48], where there is no consideration about the characteristics of any competition tasks and datasets [18,46,50]. Considering that running the GSRM on the open competition dataset is a better way to demonstrate its effectiveness, we will revise the current algorithms according to the target competition task and dataset and conduct more comprehensive comparison study. In addition, current work on enhancing the accuracy of the medical retrieval by addressing the inherent ambiguity in the queries and clinical text will be extended to cover the issues of reliability and quality of the found information, which is an important part of our future work, to make our medical retrieval solution more practical.

6. Conclusion

To solve the problems that exist in medical searches, a GSRM model was proposed in this study. With the support of an ontology that describes the background knowledge about the medical domain, the semantic information (both concepts and relations between them) implied in queries and documents, which is crucial for query intention interpretation and document content interpretation, is extracted and consumed for a document relevance calculation for the medical environment. In a practical sense, document popularity is measured with the help of external document sets through a semantic association building and consuming approach, and document popularity is then employed in conjunction with the query-document relevance to acquire the final ranked list of documents that is returned to the user. The results demonstrate that the GSRM approach provided a higher accurate medical search solution compared with the three baseline rank models,

which can server as a significant reference for the development of similar medical search systems.

Conflicts of interest

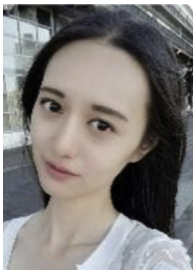
None Declared.

Acknowledgement

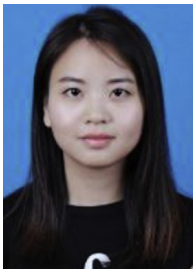
This study is supported by the National Key R&D Program of China with project no. 2017YFB1400803.

References

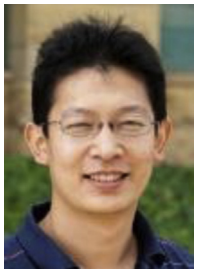
[1] W.R. Hersh, Information Retrieval: a Health and Biomedical Perspective, Springer, 2009, pp. 121–135.
[2] G. Dalton, M. McGill, Introduction to Modern Information Retrieval, McGraw-Hill, 1983.
[3] S.K. Wong, W. Ziarko, P.C.N. Wong, Generalized vector spaces model in information retrieval, Proc. 8th Annual Int'l ACM SIGIR Conf. On Research and Development in Information Retrieval, 1985, pp. 18–25.
[4] S. Deerwester, S.T. Dumais, R. Harshman, Indexing by latent semantic analysis, J. Am. Soc. Inf. Sci. 41 (6) (1990) 391–407.
[5] S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, Comput. Netw. ISDN Syst. 30 (1998) 107–117.
[6] J. Kleinberg, Authoritative sources in a hyperlinked environment, Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA'98), 1998, pp. 668–677.
[7] F. Crestani, L.M.D. Campos, J.M. Fernández-Luna, et al., A Multi-layered Bayesian Network Model for Structured Document Retrieval, Symbolic and Quantitative Approaches to Reasoning with Uncertainty, Springer, Berlin Heidelberg, 2003, pp. 74–86.
[8] S. Robertson, H. Zaragoza, The probabilistic relevance framework: BM25 and beyond, Found. Trends Inf. Retr. 3 (2) (2009) 333–389.
[9] C. Macdonald, I. Ounis, A belief network model for expert search, Proceedings of the 1st Conference on Theory of Information Retrieval, 2007, pp. 216–232.
[10] S. Niu, Y. Lan, J. Guo, et al., Which noise affects algorithm robustness for learning to rank, Inf. Retr. J. 18 (3) (2015) 215–245.
[11] C. Jung, Y. Shen, L. Jiao, Learning to rank with ensemble ranking SVM, Neural Process. Lett. 42 (3) (2015) 703–714.
[12] D. Alemida, H.M. Concalves, A combined component approach for finding collection-adapted ranking functions based on genetic programming, Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'07), 2007, pp. 399–406.
[13] P. Wang, C. Domeniconi, Building semantic kernels for text classification using wikipedia, Proceedings of KDD, ACM Press, New York, 2008.
[14] E. Makela, Survey of semantic search research, Proceedings of the Seminar on Knowledge Management on the Semantic Web, Department of Computer Science, Universityof Helsinki, 2005.
[15] Y. Yue, et al., A support vector method for optimizing average precision, Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'07), 2007, pp. 271–278.
[16] J. Li, C. Liu, B. Liu, et al., Diversity-aware retrieval of medical records, Comput. Ind. 69 (1) (2015) 30–39.
[17] M. Henzinger, Hyperlink analysis on the world wide web, Proceedings of the 6th ACM Conference Hypertext and Hypermedia (Hypertext'05), 2005, pp. 1–3.
[18] Ellen M. Voorhees, The TREC medical records track, Proc, of the International Conference on Bioinformatics, Computational Biology and Biomedical Information, BCB, 2013, pp. 189–202.
[19] Q.T. Zeng, T. Tse, Exploring and developing consumer health vocabularies, JAMIA 13 (2006) 24–29.
[20] B. Li, R.H. Li, I. King, et al., A topic-biased user reputation model in rating systems, Knowl. Inf. Syst. 44 (3) (2015) 581–607.
[21] R. Lempel, Moran, The stochastic approach for link-structure analysis (SALSA) and the TKC effect, Comput. Network.: Int. J. Comput. Telecommun. Netw. 33 (5) (2000) 387–401.
[22] B. Krishna, A.M. George, When experts agree: using non-affiliated experts to rank popular topics, ACM Trans. Inf. Syst. 20 (8) (2001) 47–58.
[23] J. Li, J. Yang, Y. Zhao, B. Liu, A Top-down approach for approximate data anonymization, Enterprise Inf. Syst. 7 (3) (2013) 272–302.
[24] Isabelle Stanton, Samuel Ieong, Nina Mishra, Circumlocution in diagnostic medical queries, SIGIR 14 (2014) 132–133.
[25] T. Hao, Z. Lu, Categorizing and ranking search engine's result by semantic similarity, Proceedings of the 2nd International Conference on Ubiquious Information Management and Conmmunication (ICUIMC), 2008, pp. 284–288.
[26] Y. An, X. Hu, I.Y. Song, Learning to discover complex mappings from web forms to ontologies, ACM International Conference on Information and Knowledge Management ACM, 2012, pp. 1253–1262.
[27] M. Daoud, et al., A personalized graph-based document ranking model using a semantic profile, Proceedings of the 18th International Conference on User Modeling, Adaption, and Personalization (UMAP'10), 2010, pp. 171–182.
[28] F. Brauer, W. Barczynsi, RankIE: document retrieval on ranked entity graphs, 3, Proceedings of the International Conference on Very Large Databases (VLDB), vol. 10, 2009, pp. 10–19.
[29] J. Li, J. Yang, C. Liu, et al., Exploiting semantic linkages among multiple sources for semantic information retrieval, Enterprise Inf. Syst. 8 (4) (2014) 464–489.
[30] F. Lamberti, et al., A relation-based page rank algorithm for semantic web search engines, 4, IEEE Transaction on Knowledge and Data Engineering, vol. 21, 2009, pp. 123–136.
[31] O. Egozi, S. Markovitch, E. Gabrilovich, Concept-based information retrieval using explicit semantic analysis, ACM Trans. Inf. Syst. 29 (2) (2011) 1–34.
[32] MeSH Homepage, http://www.nlm.nih.gov/mesh/meshhome.html, (2006).
[33] A. Kayed, E. El-Wawasmeh, Z. Qawaqneh, Ranking web sites using domain ontology concepts, Inf. Manag. 47 (9) (2010) 350–355.
[34] F. Brauer, et al., Graph-based concept identification and disambiguation for enterprise search, Proceedings of the 19th International World Wide Web Conference (WWW'10), 2010, pp. 171–180.
[35] C.C. Liu, et al., Improved semantic similarity calculating model and its application, J. Jilin Univ. (Eng. Technol. Ed.) 39 (7) (2009) 119–123.
[36] J. Li, et al., Fully automatic text categorization by exploiting WordNet, in: G.G. Lee, et al. (Ed.), Lecture Notes in Computer Science, Information Retrieval Technology, the 5th Asia Information Retrieval Symposium (AIRS), 2009, pp. 1–12.
[37] Y. Chen, M.M. Crawford, J. Ghosh, Integrating support vector machines in a hierarchical output space decomposition framework, IGARSS'04, 2004.
[38] Z. Wang, et al., A k-means algorithm based on optimized initial center points, J. Pattern Recogn. Artif. Intell. 22 (3) (2009) 299–304.
[39] E. Demidova, et al., DivQ: diversification for keyword search over structured databases, Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), 2010, pp. 331–338.
[40] J. Li, F. Wang, Semi-supervised learning via mean field methods, Neurocomputing 177 (2016) 385–393.
[41] M. Cambridge, Okapi at TREC-7: automatic ad hoc, filtering, VCL and interactive track, Inproceedings, 1999.
[42] Yuhua Li, Zuhair Bandar, David McLean, An approach for measuring semantic similarity between words using multiple information sources, IEEE Trans. Knowl. Data Eng. 15 (4) (2003) 871–882.
[43] R.D. Collins, Algorithmic Diagnosis of Symptoms and Signs: Cost-effective Approach, Lippincott Williams & Wilkins, Philadelphia, PA, 2002.
[44] J.-Q. Li, Y. Zhao, B. Liu, Exploiting external semantic resources for large scale text categorization, J. Intell. Inf. Syst. 39 (3) (2012) 763–788.
[45] J. Li, S. Zhao, Z. Huang, et al., A novel RNN-based approach for bio-NER in Chinese EMRs, J. Supercomput. (2018), https://doi.org/10.1007/s11227-017-2229-x.
[46] N.H. Shah, N. Bhatia, et al., Comparison of concept recognizers for building the open biomedical annotator, BMC Bioinf. 10 (S9) (2009) S14.
[47] Ann E.K. Sobel, The move toward electronic health records, Computer 45 (11) (2012).
[48] Doug Fridsma, Electronic health records: the HHS perspective, Computer 45 (11) (Nov. 2012) 24–26, https://doi.org/10.1109/MC.2012.371.
[49] P. Cstells, M. Fernandez, D. Vallet, An adaptation of the vector-space model for ontology-based information retrieval, 2, IEEE Transaction on Knowledge and Data Engineering, vol. 19, 2007, pp. 261–272.
[50] The CLEF Initiative, http://www.clef-initiative.eu/.



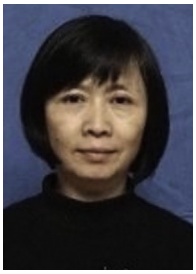
Qing Zhao, received her B.S. degree in Acting from Communication University of China, Nanguang College, Nanjing, China in 2013. M.S. degree in Software Engineering from Beijing University of technology, Beijing, China in 2016. She is currently pursuing the Ph.D. degree in Computer Science and Technology at Beijing University of Technology, Beijing, China. Her research interest includes sentiment analysis, information retrieval and big data mining.



Yangyang Kang received the M.S. degree in Software Engineering from Beijing University of Technology, Beijing, China, in 2017. Her research interest includes enterprise information system, predicative data mining, Artificial Intelligence, Internet of things, privacy protection, and big data.



Jianqiang Li, received his B.S. degree in Mechatronics from Beijing Institute of Technology, Beijing, China in 1996, M.S. and Ph.D degrees in Control Science and Engineering from Tsinghua University, Beijing, China in 2001 and 2004, respectively. He worked as a researcher in Digital Enterprise Research Institute, National University of Ireland, Galway in 2004–2005. From 2005 to 2013, he worked in NEC Labs China as a researcher, and Department of Computer Science, Stanford University, as a visiting scholar in 2009–2010. He joined Beijing University of Technology, Beijing, China, in 2013 as Beijing Distinguished Professor. His research interests are in Petri nets, enterprise information system, social computing, predicative data mining, Internet of things, privacy protection, and big data. He has over 70 publications and 42 international patent applications (22 of them have been granted in China, US, or Japan). He served as PC members in multiple international conferences and organized the IEEE Workshop on Medical Computing 2014. He served as Guest Editors to organize special issues on *Emerging Information Technology for Enhanced Healthcare* and *Big Data Technologies and Application* in Computer in Industry, Elsevier, and *Emerging Social Internet of Things: Recent Advances and Applications* in IEEE Internet of Things Journal.



Dan Wang received her B.S. degree in Computer Application from Northeastern University, Shenyang, Liaoning Province, China in 1991, M.S. and Ph.D degrees in Computer Software and Theory from Northeastern University, Shenyang, Liaoning Province, China in 1996 and 2002, respectively. She joined Beijing University of Technology, Beijing, China, in 2003. Her research interests are in Distributed Computing, Web Application Vulnerability Detection, privacy protection, and big data.