Semantics-based key concepts identification for documents indexing and retrieval on the web

Mohammed Maree

Department of Multimedia Technology,
Faculty of Engineering and Information Technology,
Arab American University,
P.O. Box 240 Jenin, 13 Zababdeh, Palestine
Email: mohammed.maree@aaup.edu

Abstract: Bridging the semantic gap on the web remains one of the crucial challenges for current horizontal and domain-specific information retrieval systems. This challenge becomes even more pronounced when users express their information needs using short queries that are formulated using a few number of keywords, therefore relying on keywords for indexing web documents results in degrading the quality of the returned results. In this article, we introduce an approach that employs knowledge captured by large-scale knowledge resources to identify key query terms and retrieve semantically-relevant documents. In this context, key terms are mapped to their semantic correspondences and variable term weights are assigned based on the semantic and taxonomic relations for each term. To demonstrate the effectiveness of the proposed approach, we have conducted experimental evaluation using Glasgow's NPL test collections. Findings indicate that the effectiveness has improved against four conventional similarity metrics that are based on the bag-of-words similarity model.

Keywords: key concepts; large-scale ontologies; semantic matching; information indexing; data analysis; precision measures.

Reference to this paper should be made as follows: Maree, M. (2021) 'Semantics-based key concepts identification for documents indexing and retrieval on the web', *Int. J. Innovative Computing and Applications*, Vol. 12, No. 1, pp.1–12.

Biographical notes: Mohammed Maree received his PhD in Information Technology from Monash University. He has published papers in various high impact journals and conferences such as *ICTAI*, *Knowledge-Based Systems* and *Journal of Information Science*. He is also a committee member/reviewer of several conferences and journals. He has supervised a number of Master students in the fields of knowledge engineering, data analysis, information and natural language processing, and hybrid intelligent systems. He joined the Arab American University, Palestine (AAUP) in 2014 and he is currently the Chairperson of the Multimedia Technology Department at the Faculty of Engineering and Information Technology at AAUP.

1 Introduction

Bridging the semantic gap between user queries and their corresponding documents on the web has been receiving increasing attention by researchers across multiple research fields and disciplines (Koopman et al., 2016), such as medical (Yang and He, 2016), news (Yokoo et al., 2016), cultural heritage (Hyvönen and Rantala, 2019), multimedia content retrieval and recommendation (Zhang et al., 2017; Liu et al., 2013), big data search and analysis (Al-Shawakfa and Alsghaier, 2018; Kong et al., 2017), and many other domains. However, with the ever increasing size of online documents, web-based indexing and retrieval approaches are still characterised by their low precision and recall (Tanaka et al., 2010; Mai, 2016; Lashkari et al., 2019). As reported in Engel (2017), information seekers still struggle to find relevant documents and data needed to perform their jobs effectively and efficiently. This issue becomes even more pronounced when users express their information needs using short queries that are formulated using a few number of keywords (Mitra et al., 2017). In this context, the utilisation of keyword-based indexing techniques to index and retrieve web documents becomes insufficient; failing to cope with the requirement of finding documents that are semantically-relevant to the users' intent. To tackle these limitations, newer approaches have been proposed for addressing the semantic-gap problem through the incorporation of semantic resources. A semantic resource in this context is employed to assist conventional keyword-based techniques through mapping keywords to their semantic correspondences such as synonyms, hypernyms, meronyms, holonyms, etc. Such corresponding entities are then added to the index structure assuming more relevant documents will be retrieved accordingly. Several studies have investigated and confirmed the impact of employing semantic resources on improving the quality of the indexing and retrieval process (Krishnan et al., 2018; Hyvönen and Rantala, 2019). However, we argue that

existing semantic recourses still suffer from a number of limitations that hinder their actual exploitation in practical application domains. Examples of these limitations are:

1 semantic knowledge incompleteness in existing semantic resources

2 lack of semantic interoperability (Li et al., 2018) and semantic contradictions across the definitions of concepts and their associated semantic relations

3 variant breadth and depth of domain coverage, i.e., whether the developed semantic resource is a generic resource such as YAGO3 (Mahdisoltani et al., 2013), WordNet (Miller, 1998), VerbNet (Schuler, 2005) and FrameNet (Baker et al., 1998) or targets a particular domain such as the MESH ontology which targets the medical domain (Lipscomb, 2000) and the CIDOC CRM ontology which covers entities in the cultural heritage domain (Crofts et al., 2008).

For a detailed discussion on these limitations, we refer the reader to Maree and Belkhatir (2015) and Bimson and Hull (2016).

In an attempt to address these drawbacks, we present an approach that combines knowledge represented by multiple large-scale general-purpose semantic resources and key concept identification and weighting scheme for identifying and analysing the latent semantic structure of queries and their corresponding documents on the web. Unlike conventional approaches that treat query terms independently, in our proposed method, we first identify key concepts in user queries using the normalised similarity technique; which assigns different weights to query terms based on their semantic informativeness. Key concepts among query terms are then mapped to their semantic correspondences in the exploited semantic resources to further enrich initial queries with additional semantically-relevant terms. Next, expanded queries are submitted to find documents with similar semantic content. In our approach, we convert queries and documents into semantic network structures wherein the nodes of each network encode terms and edges between nodes represent the different types of taxonomic and semantic relations that may exist between them. It is important to point out that as we are exploiting multiple semantic resources, there is a high probability of constructing multiple heterogeneous semantic networks. To overcome this issue, we employ the semantic networks merging algorithm (further detailed in Section 4) to integrate such networks into a single coherent network. The main goals of our work are summarised as follows:

- Identifying key concepts in user queries and their corresponding documents for constructing semantics-based inverted indexes.
- Exploiting a number of large-scale semantic resources for analysing the latent semantic structure of queries and their corresponding documents on the web.

The rest of this paper is organised as follows. Section 2 presents a discussion on the role of semantic resources in bridging the semantic-gap between user queries their corresponding documents. It also highlights the main challenges that hinder the utilisation of existing semantic resources in real-world scenarios and application settings. In Section 3, we provide the formal definitions to the key elements and resources used to build the proposed approach. In Section 4, we introduce a detailed characterisation of the proposed approach and discuss its main components. Section 5 presents the experimental evaluation steps that we have carried out to evaluate the quality of the proposed results using our proposed approach. Section 6 outlines the conclusions and highlights the future updates to our work.

2 Related works

Over the past years, semantic resources have proved to play a key role in discovering the latent topics in user queries and their corresponding documents (Anandarajan et al., 2019; Ngo and Cao, 2018; Nguyen et al., 2015; Ngo et al., 2018; Corcoglioniti et al., 2016). For details on the importance of exploiting semantic resources for query expansion purposes, we refer the reader to a recent survey conducted by Raza et al. (2019). Another recent study that explores the role of semantic resources for document classification, clustering and recommendation purposes is provided in Dou et al. (2015). As reported by the semantic web scientific community, semantic resources are fundamentally used to provide explicit machine-readable and formal specifications of conceptualisations that allow for interpreting the explicit and implicit semantics of natural language queries and their relevant documents (Bonatti et al., 2019). With the growing interest in semantic resources, several recent approaches have been proposed for semantically analysing user queries and matching them at a semantics-based level to their related documents. However, these approaches either use a single semantic resource such as Lu et al. (2015), Ngo et al. (2018), Han et al. (2016), Selmi et al. (2018), Boiński et al. (2018) and Royo et al. (2005) or multiple heterogeneous semantic resources such as Maree et al. (2016), Vigneshwari and Aramudhan (2015), Shen and Lee (2018), Kmail et al. (2015), Zhu and Iglesias (2018), Goldfarb and Le Franc (2017) and Wimalasuriya and Dou (2009). For instance, the system proposed in Royo et al. (2005) maps query keywords to their corresponding synsets in WordNet ontology. Although this system was able to identify and recognise relations among keywords, the breadth and depth of extracted relations was subjected to the limitations of WordNet, namely, limited number of semantic and taxonomic relations and limited domain coverage. In the work proposed by Lei et al. (2006), query keywords were interpreted as entities (instances, concepts or properties, respectively) that can be mapped to ontological entities to formulate formal queries from user queries. The assumption was to map keywords that represent entities in the ontology and connect them through direct relations defined in it. While the authors claimed to be able to discover relations

and handle simple and complex queries, it was not clear how this was achieved, especially given the fact that ontologies suffer from semantic knowledge incompleteness and domain coverage problems. A similar approach has been proposed by Tran et al. (2007). The main difference lies in the way queries are computed. A parameter *d* can be set and configured by the users to expand the queries graphical representation by considering neighbouring entities. In a similar line of research, Ngo et al. (2018) proposed to exploit WordNet ontology for disambiguating query keywords and mapping them to their most relevant senses. The authors argued that the problem is that in many cases a word has more than one possible direct sense, and arbitrarily picking one of them may drift the quality of the retrieved results. In addition, they enriched senses of query keywords with their most specific common hypernym to represent word meanings. Although the proposed system produced promising precision results, the precision degrades significantly when most of the keywords are not recognised by the used ontology. As acknowledged by the authors, exploiting WordNet only is not sufficient and therefore they planned to combine additional ontologies to maximise the depth and breadth of domain coverage. Similarly, the systems proposed in Lu et al. (2015), Han et al. (2016), Selmi et al. (2018), Boiński et al. (2018) and Royo et al. (2005) exploited single ontologies such as FrameNet, VerbNet and MESH to tackle the issue of semantically interpreting user queries and match them to their relevant documents. However, as reported in Maree and Belkhatir (2015), the domain coverage of these ontologies is limited and is not frequently updated; leading to significantly degrading the quality of the produced results by such systems.

Recently, there has been attempts to exploit multiple semantic resources in specialised domains such as the recruitment domain (Kmail et al., 2015), biomedical information retrieval domain (Shen and Lee, 2018) and the information extraction domain (Wimalasuriya and Dou, 2009). For instance, in Wimalasuriya and Dou (2009), the authors demonstrated through experimental results that by using multiple ontologies the quality of the system's precision can be improved. Similarly, Kmail et al. (2015) have confirmed the increase in the system's effectiveness when utilising multiple semantic resources for the purposes of matching resumes to their corresponding job posts. However, the authors in this work have also acknowledged the fact that even with the exploitation of multiple semantic resources, some entities were still unrecognised by any of the used resources. This was either because such entities have been introduced to the domain after the creation of the respective semantic resource/s or because domain experts did not include them due to time and efforts required to define each and every term in the respective domains of interest. In the work proposed in Kmail et al. (2015), the authors have utilised a set of semantic resources in the e-recruitment domain to map keywords in job posts and resumes to their corresponding entities in the used ontologies. The authors have conducted a set of experiments to demonstrate that the exploitation of multiple semantic resources can lead to better assigning relevance scores between resumes and their corresponding job posts. Inspired by the works that have utilised multiple semantic resources, we aim to explore the impact of utilising multiple semantic resources for indexing and retrieving documents on the web. In our proposed approach, and due to the specific nature of web documents that cover a wide range of domains, our interest will not be restricted to a particular domain; therefore, we propose to use large-scale general-purpose semantic resources that cover knowledge about various domains. Additionally, unlike conventional query expansion approaches, we aim to assist the semantics-based indexing and retrieval approach with the key term identification and expansion method. We argue that in real scenarios, it is not correct to assume that all query terms have the same weights and accordingly they should not be expanded equally with all potential candidate expansion terms. Therefore, it is important to identify key query terms and assign different weights to them based on their contribution to the meaning of the user's query.

3 Problem formulation

Let $Q = \{x_n\}_{n=1}^{N}$ be the set of query terms $\{x_1, x_2, \ldots, x_N\}$ and $D = \{w_m\}_{m=1}^{N}$ be the set of word tokens $\{w_1, w_2, \ldots, w_N\}$ in a given document *D*. To find the similarity between *Q* and *D*, we can apply the cosine similarity metric which deals with $x_i \in Q$ and $w_i \in D$ as vectors, and find the similarity between a given pair $\{Q, D\}$ using the following equation:

$$Similarity(Q, D) = \frac{\vec{V}(Q).\vec{V}(D)}{|\vec{V}(Q)|.|\vec{V}(D)|} \quad (1)$$

The cosine similarity weighting approach is a well-recognised scheme that has been used extensively for matching queries and their corresponding documents. This scheme is a.k.a. the TF.IDF weighting technique which stands for *term frequency-inverse document frequency*. The *frequency* for a term in this context is represented by the number of times it appears in a document, while the *document frequency* for a term is calculated by the number of documents that contain that term. To calculate the TF.IDF, we use equation (2):

$$tf.idf(t, d) = tf(t, d) * ifd(t) \quad (2)$$

where

- $tf(t, d) = \sum_{i \in d}^{|d|} 1\{d_i = t\}$
- $idf(t) = \log\left(\frac{|D|}{\sum_{d \in D}^{|D|} 1\{t \in d\}}\right)$.

Although this scheme has proved to be efficient in matching documents with their corresponding queries, it still suffers from a number of limitations as follows:

- When documents contain a huge number of terms, the process of computing the *tf* as well as *idf* will be characterised by a huge time consumption. This is because the TF.IDF scheme computes document similarity directly in the term-count space, wherein its efficiency degrades for documents with large vocabularies.
- The algorithm assumes that the counts of different terms provide independent evidence of similarity. However, this may not be true for every term in the document as there will be more informative terms that have significant contribution to the meaning of the document than other supportive terms.
- It is based on the bag-of-words (BoW) model, therefore it does not capture the position of terms in the text, terms co-occurrences across different documents, and compound terms (uni, bi or tri gram terms). However, in real-world scenarios, recognising compound terms, in addition to the grammatical category/ies they belong to can lead to better understanding of user queries and accordingly find their relevant documents.
- Another important limitation in this scheme is that it ignores the latent semantic dimensions in the processed queries and documents. This is indeed one of the major obstacles towards the utilisation of this scheme in practical settings. For instance, if a query contains the term 'flu' and a documents contains the term 'influenza', they will be judged as irrelevant using this model, though they are relevant as both terms are synonymous terms and have the same meaning.

Due to the abovementioned limitations, we propose extending the TF.IDF scheme through incorporating semantic information derived from the exploited semantic resources. Our goal in this context is to enrich the initial semantic indexes for the processed documents with semantically-relevant terms that are not explicitly mentioned in their content. We argue that query terms that have synonyms, and are abbreviated or represent acronyms are more informative and essential for indexing and retrieval purposes. Therefore, we assign higher weights for such terms compared to other supportive query terms, and also against the full representations of acronyms and abbreviations that we automatically add to the original query. The modified similarity metric is employed in this context for finding the relevance score between a given semantically-expanded query Q_s and document D_s based on their dot product. Expansion terms in this context are obtained through exploiting additional semantic resources that can be employed to discover the latent semantic relations in texts and accordingly lead to a more precise matching procedure. In this context, a semantic resource can be defined as follows.

Definition 1: Semantic resource: a semantic resource ζ is a quadruple, $\zeta := <C, P, I, V>$ where:

C Is the set of concepts that are defined in the semantic resource ζ. A concept hierarchy is defined over C in ζ where a pair $(C, \leq)$ reflects a subsumption relation on $C \times C$. We call $c \in C$ the set of concepts, and $\leq$ the hypernymy relation.

P Is the set of properties that are defined on C.

I Is the set of individuals or instances of C.

V Is the set of property values on P.

The query processing and expansion module takes a given query Q as input and produces a set of semantic networks $\Omega Q = \{\Omega_{Q1}, \Omega_{Q2}, \ldots, \Omega_{Qn}\}$ as output. These networks are used to represent the semantic index for each Q and are automatically constructed based on the set of exploited semantic resources $S_\zeta = \{\zeta_1, \zeta_2, \ldots, \zeta_n\}$. A semantic network in this context can be formally defined as follows.

Definition 2: Semantic network: a semantic network is a triplet, $\Omega := <C, \leq_C, \sigma, R, \leq_R>$ where:

C Is the set of concept identifiers in Ω.

R Is the set of relation identifiers in Ω.

$\leq_C$ Is the partial order relation on C, a.k.a. concept hierarchy or taxonomy.

$\leq_R$ Is the partial order relation on R, a.k.a. relation hierarchy or taxonomy.

σ Is a function $R \rightarrow C \times C$ that maps concepts in the semantic network Ω. We would like to point out that since we are exploiting more than one semantic resource, there is a probability that the produced semantic networks are semantically heterogeneous in terms of their conceptual and terminological representations. Therefore, we propose resolving such heterogeneity through integrating such networks into a single coherent semantic network. The merging algorithm can be defined as follows.

Definition 3: Merging heterogeneous networks: the merging algorithm takes a set of semantic networks as input and produces a single merged network as output. Formally, we define the merged network as $\Omega_{merged} := <C, \leq_C, \sigma, R, \leq_R>$ where the set of concepts $\{(c_i, c_j) \mid c_i \in \Omega_i, c_j \in \Omega_j\}$.

It is important to point out that we construct semantic indexes to represent each document in the corpus in the same manner as we process user queries.

As we highlighted in this section, the exploitation of multiple semantic resources does not necessarily resolve the semantic knowledge incompleteness problem. Therefore, we propose to further enrich the constructed merged semantic networks with additional concepts that are not recognised by the utilised semantic resources. To do this, we employ the semantic networks enrichment technique that is formally presented in Maree and Belkhatir (2015). In the next sections, we present the detailed characterisation of our

proposed system and experimentally evaluate the quality of the proposed techniques using real-world scenarios and dataset.

4 Detailed characterisation of the proposed system

In this section, we present an approach for automatic query interpretation using multiple semantic resources. The intuitions behind using these resources are:

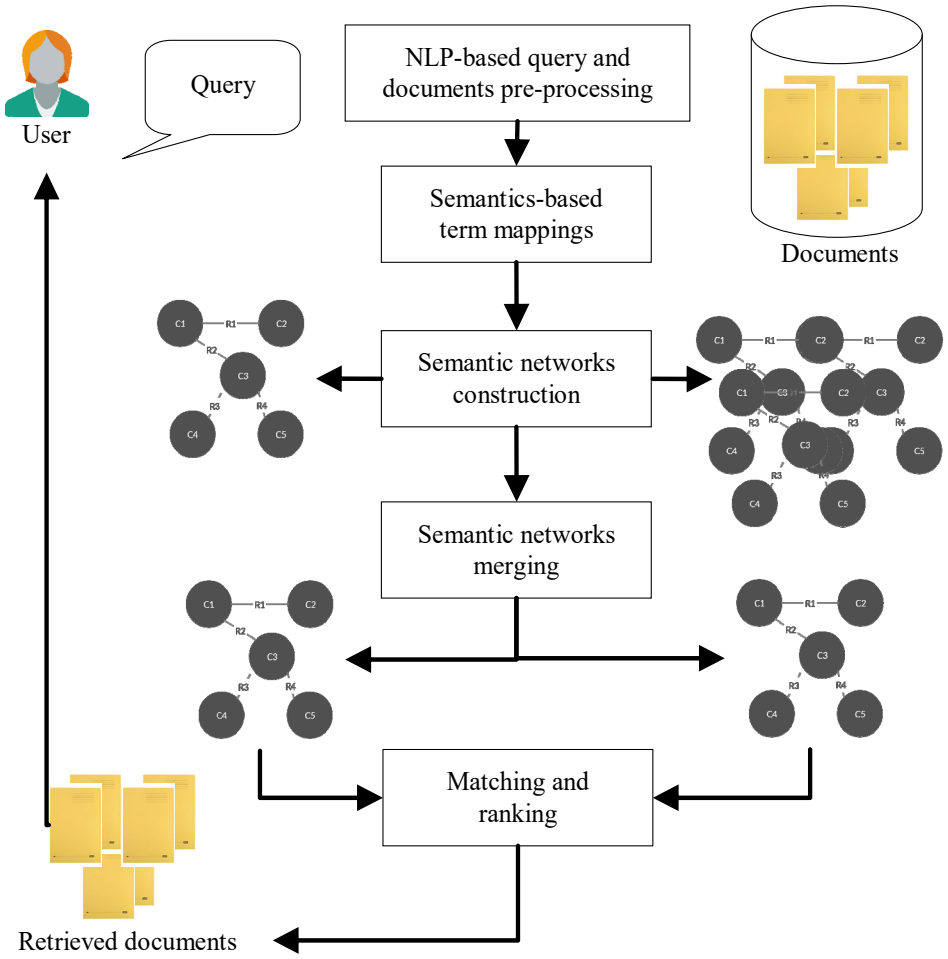
1 they make cooperative semantics-based query interpretation, weighting and expansion decisions

2 they provide wider depth and breadth of domain coverage and knowledge representation.

Figure 1 depicts the main components and modules of our proposed approach.

As depicted in Figure 1, when a user submits a query, we utilise NLP-based techniques for pre-processing the query terms. This includes applying tokenisation, stopwords removal, depluralisation and compound terms recognition. After this step, each term is submitted to the utilised semantic resources and mapped to the proper entity(ies) that it belongs to. By semantics-based term mapping, we aim to derive all semantically-relevant terms to each query term. We need to find out whether a term t is a noun, verb, an adjective or adverb, and also the synonyms as well as other taxonomically-related terms to that term. It is important to point out that the same steps are also applied on the documents in the collection in an attempt to construct semantic indexes using the *semantic networks construction* module. However, prior to starting the execution of this module, we assign different weights to query terms and their semantic correspondences using the normalised cosine similarity metric. In this context, original query terms that have synonyms and hypernyms are assigned higher weights against other semantically-relevant terms such as meronyms, hyponyms or holonyms. Formally, for a query $Q = \{x_n\}_{n=1}^{N}$ that comprises terms $\{x_1, x_2, \ldots, x_N\}$, each x_i can belong to any of the following categories:

Figure 1 General overview of the proposed system (see online version for colours)



- Original terms: Are terms that can be mapped to concepts in the exploited semantic resources, in addition to their hypernyms. The reason for assigning the same weight for synonyms and hypernyms is because we believe that users often express their information needs using the exact keywords, their synonyms or a more generalised form the keywords. For instance, a user looking for information about police stations he/she may submit the query: 'police station', its synonym: 'police headquarter' or 'police office', or a higher level query such as: 'station' or 'facility'. Although there are some other taxonomically-relevant terms such as a meronym of the query: 'squad room', we believe that users do not often use such terms to express their information needs. Similarly, a user interested in information about 'elections' may type the query 'elections' or its hypernym 'voting' instead of typing the hyponyms 'general elections' or 'primary elections'.
- Acronyms and abbreviations: *Acronyms* are terms that are formed from the initial letters of some longer names and are pronounced differently than the full representation, while *abbreviations* are terms that are written differently from their full representations, but are pronounced the same. Examples of acronyms are ESL which stands for *English as a second language* and SEO which stands for *search engine optimisation*. Examples of abbreviations on the other hand are JSP which stands for *Java Server Pages* and SQL which stands for *Structured Query Language*. It is important to point out that in some case, acronyms can cause polysemy problems that need to be addressed as well. Our attempt in this context is to use other query words as the context of each polysemous query term in order to disambiguate its meaning.
- Supportive terms: Any other terms in the user query that neither could be recognised by the used semantic resources nor were classified as acronyms or abbreviations.

Next, we use the set of weighted and expanded query terms to automatically construct semantic networks that represent query terms and relations that hold between them. In this context, a semantic network $\Omega := <C, \leq_C, \sigma, R, \leq_R>$ includes directed arcs with labels from R between nodes from $x \in C$ and $y \in C$ to signify that a predicate in form $R(x, y)$ is true. In particular, we employ Algorithm 1 to construct semantic networks for each query-document pair in the corpus.

Algorithm 1 Semantic networks construction algorithm

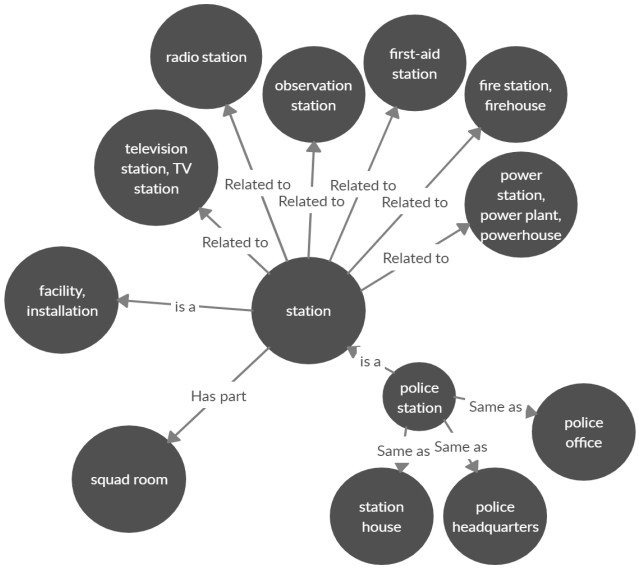
Input: weighted and expanded query terms String[] QTerms

Output: semantic networks based on the exploited semantic resources

```
String [][] computeRelations (QTerms)
String [][] SemRel;
Sresources ← <>;
res = getSemanticResources(Sresources);
for each res ∈ Sresources
  for all ik ∈ QTerms
  for all jk ∈ QTerms
      SemRel[i][j] = getRel(QTerms[i], QTerms[j], res);
  return SemanticRelations;
  String getRelation(token1, token2, res)
  String relation = "Unknown";
      relation = semRelation(token1, token2, res);
      if(relation == "UnKnown)
          relation = stringMatcher(token1, token2);
  return relation;
makeAxioms(ComputeRelations);
String semRelation(token1, token2, res)
String relation = "Unknown";
  if( res == "WORDNET")
      findRelationWN(token1,token2);
  if(res == "OpenCyc")
      findRelationOpenCyc(token1, token2);
  if (res == "YAGO")
      findRelationYAGO(token1, token2);
return relation;
String stringMatcher(token1, token2)
String relation = "Unknown";
relation = findSimilarity(token1, token2);
return relation;
```

During this step, a semantic resource may produce the set $S_\Omega = \{0, 1, \ldots, n\}$ networks. This indeed demonstrates the issue of semantic knowledge incompleteness in the used semantic networks. For instance, the compound term 'police station' may be recognised by some of the exploited semantic resources, while it may not be defined by the rest. Additionally, the same term may have multiple senses or derivationally or taxonomically related terms in one semantic resource which could be different than the way it is captured by others. Therefore, it is important to resolve such semantic heterogeneity in the produced semantic networks prior to starting the matching and ranking procedure. We argue that being able to merge heterogeneous networks results in obtaining an aggregated decision made by the majority of the employed semantic resources on the types of relations that should hold between their entities. In the next examples, we first depict – using Figure 2 – a visual semantic network for the query 'police station' including all nodes in the network. In the second example, we demonstrate the details of the utilised techniques in our proposed system using a document sample. The third example is presented to demonstrate the semantic networks that can be produced from a short query that comprises compound terms.

Figure 2 An example of a semantic network



Note: Synonyms, meronyms as well as additional semantically-relevant terms are depicted in this diagram.

Example 1: Sample query = 'police station'.

In this example, we consider a compound term that belongs to the NLP 2-gram tokens category. This token is first submitted to the exploited semantic recourse in order to first find whether it is recognised as a 2-gram token. Considering WordNet ontology, we can see that it recognises this token, in addition to a set of semantically and syntactically relevant terms, such as synonyms, hypernyms and meronyms. These indeed represent the set of nodes that are depicted in Figure 2. To reproduce the same semantic network, we refer the reader to using the viswords (https://visuwords.com/police%20station) open access online application to demonstrate the semantic network that is returned when submitting the query to viswords.

Example 2: Content extracted from a document (https://en.wikipedia.org/wiki/Java) that describes the term *java* (see online version for colours)

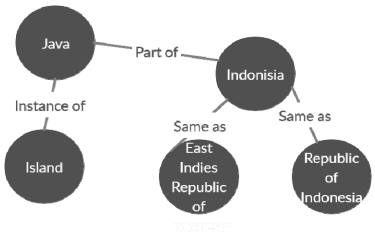
Segment	*Image*
Java (Indonesian: Jawa) is an island of Indonesia and the site of its capital city, Jakarta. Once the centre of powerful Hindu-Buddhist kingdoms, Islamic sultanates, and the core of the colonial Dutch East Indies, Java now plays.	

In this example, we use WordNet (Miller, 1998) and OpenCyc (Matuszek et al., 2006) general-purpose semantic resources. First, the stopword removal function removes stopwords based on a pre-defined list such as [(,) , : ,]. Next, the *n*-gram tokenisation algorithm tokenises the text into unigram, bigram and trigram tokens. After this step, each token is submitted to each of the ontologies to find whether it is defined in it or not. For instance, when we explore the hierarchy of WordNet, we will find that the term 'Java' has three different senses as follows:

1	Java – an island in Indonesia south of Borneo; one of the world's most densely populated regions
2	Coffee, java – a beverage consisting of an infusion of ground coffee beans; "he ordered a cup of coffee"
3	Java – a simple platform-independent object-oriented programming language used for writing applets that are downloaded from the World Wide Web by a client and run on the client's machine

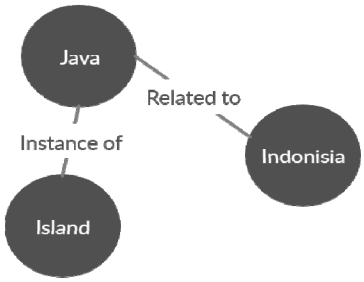
Although this term has three different senses, only the first sense is included in the produced semantic network. This is because that other senses (2 and 3) are not semantically related to the other rest of the tokenised terms such as 'Indonesia' and 'Island'. Accordingly, automatic disambiguation of the text is performed, producing a semantic network that includes only the semantically related terms. In addition, the synonyms of the disambiguated keywords are automatically included in the produced semantic network. For instance, from Example 2, we find that the set of terms {Java, Island, Indonesia} exist in both WordNet and OpenCyc semantic resources. Form this set, semantic networks are constructed based on both semantic resources as shown in Figure 3.

Figure 3 Semantic networks constructed based on the employed semantic resources, (a) WordNet-based semantic network (b) OpenCyc-based semantic network



Semantic network Ω_1

(a)

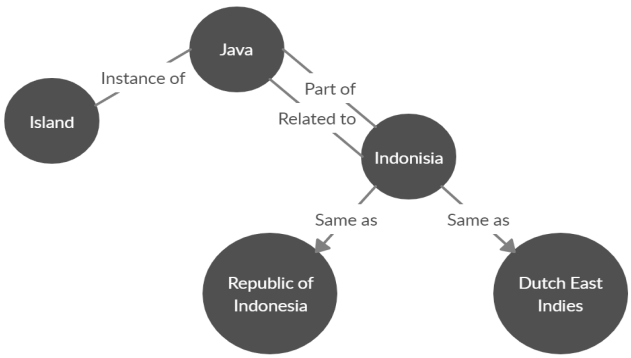


(b)

As we can see from Figure 3, the exploited semantic resources produced heterogeneous semantic networks. These semantic networks represent different perspectives as they reflect the knowledge captured and encoded by domain experts whom developed both semantic resources. For example, according to OpenCyc, the relation between the terms 'Java' and 'Indonesia' is 'related to'. While in

WordNet, the relation between the same terms is 'part of'. Therefore, to resolve this terminological heterogeneity between the produced networks, we utilise the merging algorithm described in Section 4. In this algorithm, we use the merging techniques proposed in Maree and Belkhatir (2010). The execution of these techniques is based on a prioritised procedure based on their significance and execution into semantic, string, and statistical-based merging techniques, respectively. The result of merging the semantic networks is shown in Figure 4. The rest of *n*-gram tokens that are not defined in any of the exploited semantic resources such as the token 'jawa' are considered as supportive terms and are assigned lower weights accordingly.

Figure 4 Result of integrating two heterogeneous semantic networks



Example 3: Query no. 126 [Forum for Information Retrieval Evaluation (FIRE) dataset]

```
<top lang='en'>
<num>126</num>
<title>Swine flu vaccine</title>
<desc>
Indigenous vaccine made in India for swine flu
prevention</desc>
```

As shown in this query example, we have the title of the query and also a description that further explains it. We consider the title in this context to demonstrate the semantic networks that can be produced using short-term queries such as this one. Figure 5 depicts the semantic network that contains the node 'swine flu', in addition to a number of semantically and taxonomically related terms.

As depicted by Figure 5, the term 'swine flu' has a synonym that is 'swine influenza', while the terms 'influenza, flue and grippe' are parent nodes or hypernyms of the original query term 'swine flu'. As we have pointed out earlier in this section. Original query terms and their synonyms and hypernyms are given higher weights compared to other query terms. In this context, the query term 'vaccine' has also synonyms and hypernyms (these are highlighted using red circles) as depicted in Figure 6.

Accordingly, less weights are assigned to hyponyms of the term 'vaccine' such as 'DPT vaccine, pneumococcal vaccine, Pneumovax, poliovirus vaccine, proteosome vaccine, proteasome'.

In the next section, we present the details of the experimental evaluation steps that we have carried out to evaluate the quality of the proposed approach.

Figure 5 A semantic network for the query term 'swine flu'

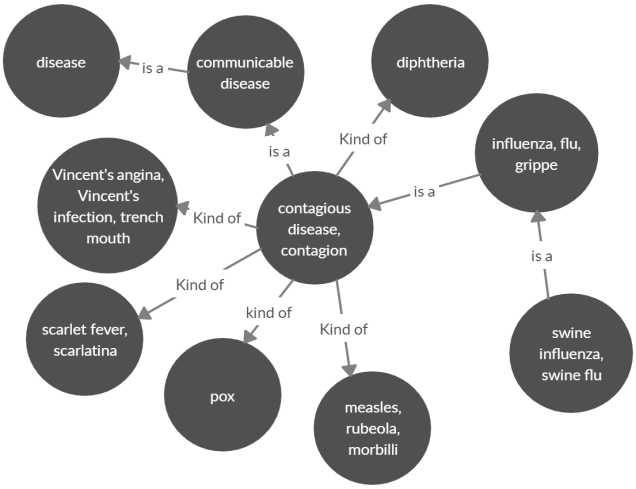
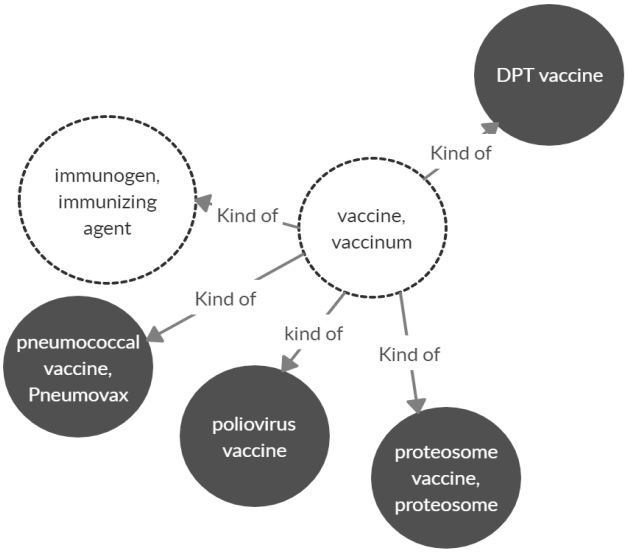


Figure 6 A semantic network for the query term 'vaccine'



5 Experimental evaluation of the proposed system

In this section, we present the steps that we have carried out to experimentally validate our proposal. To develop the system's prototype and implement its modules, we have used Java programming language on a PC with core i7 CPU (2.1 GHz) and (16 GB) RAM with Windows 10 operating system. It was important during the experimental validation step to have a dataset that comprises the following structure to ensure the reproducibility of the experiments by other researchers in the field. The structure comprises these components:

- Document collection: These are documents that contain content about different domains of interest.
- Queries: They are provided in the dataset to be used as a reference for finding relevant documents in the document collection.
- Relevance judgements: The relevance scores that are assigned between each query-document pair.

We use the baseline results as our ground-truth and compare the precision of three additional similarity metrics to those in the ground truth. Furthermore, we employ our semantics-based query term identification and expansion method to assign new relevance scores between documents and their corresponding documents.

The used dataset is a publically-available dataset that has been extensively used by several semantic matching approaches over the past years. The dataset is composed of 20 documents that we use to represent the queries and a collection of 10,000 textual segments that we use to represent our document collection. To validate the effectiveness of our proposal, we start with the baseline results that are depicted in Table 1. To obtain these results, we have used the conventional similarity measure in order to calculate the similarity between the text segments that are used to represent the queries and their associated documents. For demonstration purposes, we have selected the first ten queries and their first five corresponding documents to compute the P@5 precision results.

Table 1 Baseline similarity results using the four similarity techniques

Q_Id	*D_id*	*Cosine sim.*	*Jaccard sim.*	*Normalised Levenshtein sim.*	*Sorensen-Dice sim.*
1.txt	1.txt	37.12%	22.64%	25.86%	36.92%
1.txt	10.txt	34.94%	21.25%	27.42%	35.05%
1.txt	100.txt	45.31%	22.58%	14.51%	36.84%
1.txt	1000.txt	27.90%	16.00%	25.81%	27.59%
1.txt	10000.txt	35.49%	20.00%	25.62%	33.33%
2.txt	1.txt	41.33%	25.47%	29.31%	40.60%
2.txt	10.txt	32.99%	19.05%	23.29%	32.00%
2.txt	100.txt	56.33%	27.81%	16.98%	43.52%
2.txt	1000.txt	*17.17%*	*8.43%*	28.77%	*15.56%*
2.txt	10000.txt	41.88%	23.85%	*32.23%*	38.52%
3.txt	1.txt	46.95%	32.73%	26.72%	*49*.32%
3.txt	10.txt	30.38%	22.83%	28.57%	37.17%
3.txt	100.txt	54.89%	27.95%	18.83%	43.69%
3.txt	1000.txt	26.22%	10.75%	25.27%	19.42%
3.txt	10000.txt	49.05%	*32.14%*	28.10%	48.65%
4.txt	1.txt	31.51%	18.37%	21.55%	31.03%
4.txt	10.txt	17.97%	10.67%	25.49%	19.28%
4.txt	100.txt	36.61%	15.03%	12.04%	26.14%
4.txt	1000.txt	21.23%	8.96%	17.78%	16.44%
4.txt	10000.txt	38.54%	21.65%	21.49%	35.59%
5.txt	1.txt	27.44%	13.68%	19.83%	24.07%
5.txt	10.txt	24.35%	11.94%	29.41%	21.33%
5.txt	100.txt	33.43%	10.53%	*10.49%*	19.05%
5.txt	1000.txt	21.86%	10.17%	20.00%	18.46%
5.txt	10000.txt	38.86%	15.79%	20.66%	27.27%
6.txt	1.txt	32.90%	15.46%	20.69%	26.79%
6.txt	10.txt	20.20%	11.27%	23.53%	20.25%
6.txt	100.txt	40.61%	17.01%	10.49%	29.07%
6.txt	1000.txt	31.24%	16.95%	26.67%	28.99%
6.txt	10000.txt	31.80%	12.87%	20.66%	22.81%
7.txt	1.txt	28.19%	24.04%	25.86%	38.76%
7.txt	10.txt	29.82%	18.52%	22.22%	31.25%
7.txt	100.txt	50.19%	20.38%	16.36%	33.86%
7.txt	1000.txt	23.34%	11.69%	25.40%	20.93%
7.txt	10000.txt	37.50%	21.30%	23.97%	35.11%
8.txt	1.txt	33.92%	19.82%	25.86%	33.08%
8.txt	10.txt	24.99%	13.64%	23.19%	24.00%
8.txt	100.txt	49.38%	23.72%	14.81%	38.34%
8.txt	1000.txt	32.21%	16.88%	27.54%	28.89%
8.txt	10000.txt	32.44%	16.38%	26.45%	28.15%
9.txt	1.txt	42.12%	20.75%	27.59%	34.38%
9.txt	10.txt	29.64%	15.85%	23.19%	27.37%
9.txt	100.txt	*58.66%*	25.33%	16.98%	40.43%
9.txt	1000.txt	28.65%	10.39%	28.99%	18.82%
9.txt	10000.txt	48.30%	26.21%	24.79%	41.54%
10.txt	1.txt	43.34%	21.24%	31.03%	35.04%
10.txt	10.txt	20.60%	13.04%	23.94%	23.08%
10.txt	100.txt	48.24%	24.68%	16.98%	39.59%
10.txt	1000.txt	38.50%	20.51%	29.58%	34.04%
10.txt	10000.txt	47.43%	26.36%	28.93%	41.73%

As shown in Table 1, the produced similarity scores using the four similarity measures demonstrate a low precision quality. For instance, the returned similarity scores by the Cosine similarity measure ranged between a minimum of 17.17% and a maximum of 58.66%, while they ranged between 8.43% and 32.14%, and 10.49% and 32.23% when using the Jaccard and normalised Levenshtein similarity, respectively. Considering the Sorensen-Dice similarity score, we can see a similar range of precision results is produced when compared to the cosine similarity technique. The reason behind the low precision results is that the used measures ignore all semantically and taxonomically related relations that exist between the terms that are used to describe the textual content of the used documents and their corresponding queries in the dataset. This experimentally

demonstrates our argument on the limitations that hinder the exploitation of conventional BoW similarity models. In the next experiment, we investigate the impact of employing our proposed method for assigning similarity scores between the same query-document pairs using the same settings.

Table 2 Similarity results after employing our proposed semantics-based method

Q_Id	*D_id*	*Cosine sim.*	*Jaccard sim.*	*Normalised Levenshtein sim.*	*Sorensen-Dice sim.*
1.txt	1.txt	55.57%	41.30%	27.07%	58.46%
1.txt	10.txt	67.76%	49.65%	27.73%	66.36%
1.txt	100.txt	75.58%	39.31%	17.82%	56.44%
1.txt	1000.txt	60.44%	29.50%	21.66%	45.56%
1.txt	10000.txt	62.48%	45.54%	25.25%	62.58%
2.txt	1.txt	56.56%	40.64%	28.31%	57.79%
2.txt	10.txt	63.54%	46.62%	27.65%	63.59%
2.txt	100.txt	64.77%	40.46%	19.19%	57.61%
2.txt	1000.txt	45.35%	28.43%	24.29%	44.27%
2.txt	10000.txt	61.94%	42.27%	28.69%	59.42%
3.txt	1.txt	66.95%	45.09%	26.19%	62.15%
3.txt	10.txt	65.24%	35.44%	24.06%	52.33%
3.txt	100.txt	*75.74%*	*54.68%*	23.38%	*70.70%*
3.txt	1000.txt	58.39%	31.17%	25.20%	47.53%
3.txt	10000.txt	72.70%	50.60%	20.95%	67.20%
4.txt	1.txt	58.61%	38.38%	26.03%	55.47%
4.txt	10.txt	56.51%	32.91%	24.32%	49.52%
4.txt	100.txt	60.16%	32.23%	15.80%	48.75%
4.txt	1000.txt	49.77%	*22.60%*	21.07%	*36.86%*
4.txt	10000.txt	64.91%	36.00%	24.92%	52.94%
5.txt	1.txt	52.62%	36.61%	25.62%	53.60%
5.txt	10.txt	56.66%	32.47%	28.57%	49.02%
5.txt	100.txt	57.67%	33.96%	15.66%	50.70%
5.txt	1000.txt	*44.67%*	27.04%	22.55%	42.57%
5.txt	10000.txt	63.20%	36.99%	23.93%	54.00%
6.txt	1.txt	63.97%	36.97%	*29.55%*	53.98%
6.txt	10.txt	45.66%	25.87%	25.97%	41.11%
6.txt	100.txt	59.21%	29.30%	*12.63%*	45.32%
6.txt	1000.txt	58.88%	25.00%	25.52%	40.00%
6.txt	10000.txt	52.26%	28.37%	21.80%	44.20%
7.txt	1.txt	49.76%	40.00%	26.65%	57.14%
7.txt	10.txt	66.74%	34.27%	22.87%	51.04%
7.txt	100.txt	66.09%	33.46%	15.08%	50.15%
7.txt	1000.txt	55.80%	28.11%	26.11%	43.88%
7.txt	10000.txt	55.67%	41.18%	24.26%	58.33%
8.txt	1.txt	55.29%	43.84%	22.73%	60.96%
8.txt	10.txt	52.56%	36.67%	27.11%	53.66%
8.txt	100.txt	74.97%	43.32%	21.14%	60.45%
8.txt	1000.txt	59.32%	29.91%	24.83%	46.05%
8.txt	10000.txt	62.56%	48.05%	24.92%	64.91%
9.txt	1.txt	57.22%	41.50%	26.65%	58.66%
9.txt	10.txt	49.22%	34.66%	25.27%	51.48%
9.txt	100.txt	75.40%	44.24%	19.05%	61.34%
9.txt	1000.txt	52.90%	34.93%	20.97%	51.77%
9.txt	10000.txt	63.70%	50.00%	25.74%	66.67%
10.txt	1.txt	60.57%	39.47%	24.79%	56.60%
10.txt	10.txt	54.37%	32.73%	20.64%	49.32%
10.txt	100.txt	66.46%	38.06%	15.15%	55.14%
10.txt	1000.txt	50.04%	26.32%	22.55%	41.67%
10.txt	10000.txt	62.95%	40.63%	24.43%	57.78%

As we see in Table 2, a significant level of improvement upon the overall's quality of the semantic matching process has been achieved. This indeed supports our argument that the exploitation of ontologies and other semantic resources can play a significant role in improving the quality of conventional similarity methods. Additionally, the utilisation of the proposed key concepts identification approach has resulted in retrieving more relevant documents due to the fact that processing query terms using the developed weighted term dependence model can emphasise the relations that exist among important terms and accordingly enables constructing semantic network-based indexed that includes all semantically relevant terms that were not initially submitted in the user query. We would like to point out that we plan to further extend out proposed method by integrating newer semantic resources in an attempt to investigate their impact on the quality of the proposed system.

6 Conclusions and future work

In this article, we discussed one of the crucial challenges for current search engines; that is their low precision results. We demonstrated the fact that the semantic gap between query-document pairs has a substantial impact on the quality of the retrieved results. In addition, we discussed the importantce of identifying and reweighting query terms to produce more precise results that better match the users' information needs. We introduced our method of employing knowledge captured by multiple large-scale semantic resources for identifying candidate key concepts for indexing and retrieving documents that are semantically-relevant to user queries. We have conducted experiments using a real-world dataset to demonstrate the effectiveness of the proposed approach. We have demonstrated that conventional similarity measures can actually be further extended to incorporate additional

semantically-expanded terms that lead to retrieving more precise results. In the future work, we plan to explore the impact of exploiting newer semantic resources on the quality of the retrieved results. We also plan to conduct additional experiments using different datasets, namely FIRE and Reuters text collections. We plan to participate in future forums for information indexing and retrieval and publicise our results and system implementation details to be used by other researchers in the field.

References

Al-Shawakfa, E. and Alsghaier, H. (2018) 'An empirical study of cloud computing and big data analytics', *International Journal of Innovative Computing and Applications*, Vol. 9, No. 3, pp.180–188.

Anandarajan, M., Hill, C. and Nolan, T. (2019) 'Semantic space representation and latent semantic analysis', in *Practical Text Analytics. Advances in Analytics and Data Science*, Vol. 2, Springer, Cham.

Baker, C.F., Fillmore, C.J. and Lowe, J B. (1998) 'The Berkeley FrameNet project', *Proceedings of the 17th international conference on Computational linguistics*, Association for Computational Linguistics, Vol. 1, pp.86–90.

Bimson, K. and Hull, R. (2016) 'unnatural language processing: characterizing the challenges in translating natural language semantics into ontology semantics', in Workman. M. (Ed.): *Semantic Web*, Springer, Cham

Boiński, T., Szymański, J., Dudek, B., Zalewski, P., Dompke, S. and Czarnecka, M. (2018) 'DBpedia and YAGO based system for answering questions in natural language', *International Conference on Computational Collective Intelligence*, Springer, pp.383–392.

Bonatti, P.A., Decker, S., Polleres, A. and Presutti, V. (2019) 'Knowledge graphs: new directions for knowledge representation on the semantic web', *Dagstuhl Seminar 18371*, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Corcoglioniti, F., Dragoni, M., Rospocher, M. and Aprosio, A.P. (2016) 'Knowledge extraction for information retrieval', *European Semantic Web Conference*, Springer, pp.317–333.

Crofts, N., Doerr, M., Gill, T., Stead, S. and Stiff, M. (2008) *Definition of the CIDOC Conceptual Reference Model*, CIDOC CRM Special Interest Group, No. 5, ICOM/CIDOC Documentation Standards Group.

Dou, D., Wang, H. and Liu, H. (2015) 'Semantic data mining: a survey of ontology-based approaches', *Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015)*, IEEE, pp.244–251.

Engel, J. (2017) 'Improving retrieval of structured and unstructured information: practical steps for better classification, navigation and search', *Business Information Review*, Vol. 34, No. 2, pp.86–95.

Goldfarb, D. and Le Franc, Y. (2017) 'Enhancing the discoverability and interoperability of multi-disciplinary semantic repositories', *S4BioDiv@ ISWC*.

Han, X., Lv, T., Hu, Z., Wang, X. and Wang, C. (2016) 'Text summarization using FrameNet-based semantic graph model', *Scientific Programming*, November, Vol. 2016, pp.1–10.

Hyvönen, E. and Rantala, H. (2019) 'Knowledge-based relation discovery in cultural heritage knowledge graphs', *DHN*, pp.230–239.

Kmail, A.B., Maree, M. and Belkhatir, M. (2015) 'MatchingSem: online recruitment system based on multiple semantic resources', *2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, IEEE, pp.2654–2659.

Kong, F., Wang, J., Yu, J. and Wang, X. (2017) 'Improved phrase search construction over encrypted data in cloud storage', *International Journal of Innovative Computing and Applications*, Vol. 8, No. 1, pp.3–11.

Koopman, B., Zuccon, G., Bruza, P., Sitbon, L. and Lawley, M.J. (2016) 'Information retrieval as semantic inference: a graph inference model applied to medical search', *Information Retrieval Journal*, Vol. 19, No. 1, pp.6–37.

Krishnan, A., Deepak, P., Ranu, S. and Mehta, S. (2018) 'Leveraging semantic resources in diversified query expansion', *World Wide Web*, Vol. 21, No. 4, pp.1041–1067.

Lashkari, F., Bagheri, E. and Ghorbani, A.A. (2019) 'Neural embedding-based indices for semantic search', *Information Processing & Management*, Vol. 56, No. 3, pp.733–755.

Lei, Y., Uren, V. and Motta, E. (2006) 'Semsearch: a search engine for the semantic web', *International Conference on Knowledge Engineering and Knowledge Management*, Springer, pp.238–245.

Li, T., Gupta, B.B. and Metere, R. (2018) 'Socially-conforming cooperative computation in cloud networks', *Journal of Parallel and Distributed Computing*, July, Vol. 117, pp.274–280.

Lipscomb, C.E. (2000) 'Medical subject headings (MeSH)', *Bulletin of the Medical Library Association*, Vol. 88, No. 3, pp.265–266.

Liu, M., Li, J. and Liu, H. (2013) 'The colour and texture – a novel image retrieval technology based on human vision', *International Journal of Innovative Computing and Applications*, Vol. 5, No. 1, pp.47–56.

Lu, M., Sun, X., Wang, S., Lo, D. and Duan, Y. (2015) 'Query expansion via wordnet for effective code search', *2015 IEEE 22nd International Conference on Software Analysis, Evolution, and Reengineering (SANER)*, IEEE, pp.545–549.

Mahdisoltani, F., Biega, J. and Suchanek, F.M. (2013) 'YAGO3: a knowledge base from multilingual Wikipedias', in *7th Biennial Conference on Innovative Data Systems Research. CIDR Conference*, January, Asilomar, USA, hal-01699874.

Mai, J-E. (2016) *Looking for Information: A Survey of Research on Information Seeking, Needs, and Behavior*, 4th ed., 29 April, p.528, Emerald Group Publishing.

Maree, M. and Belkhatir, M. (2010) 'A coupled statistical/semantic framework for merging heterogeneous domain-specific ontologies', *2010 22nd IEEE International Conference on Tools with Artificial Intelligence*, IEEE, pp.159–166.

Maree, M. and Belkhatir, M. (2015) 'Addressing semantic heterogeneity through multiple knowledge base assisted merging of domain-specific ontologies', *Knowledge-Based Systems*, January, Vol. 73, pp.199–211.

Maree, M., Belkhatir, M., Fauzi, F., Kmail, A.B., Ewais, A. and Sabha, M. (2016) 'Multiple ontology-based indexing of multimedia documents on the World Wide Web', *Intelligent Decision Technologies 2016*, Springer.

Matuszek, C., Witbrock, M., Cabral, J. and Deoliveira, J. (2006) 'An introduction to the syntax and content of Cyc', *UMBC Computer Science and Electrical Engineering Department Collection*.

Miller, G.A. (1998) 'WordNet: an electronic lexical database', *Computational Linguistics*, pp.292–296, MIT Press.

Mitra, B., Diaz, F. and Craswell, N. (2017) 'Learning to match using local and distributed representations of text for web search', *Proceedings of the 26th International Conference on World Wide Web*, International World Wide Web Conferences Steering Committee, pp.1291–1299.

Ngo, V.M. and Cao, T.H. (2018) *Discovering Latent Concepts and Exploiting Ontological Features for Semantic Text Search*, arXiv preprint arXiv: 1807.05578.

Ngo, V.M., Cao, T.H. and Le, T. (2018) *WordNet-based Information Retrieval Using Common Hypernyms and Combined Features*, arXiv preprint arXiv: 1807.05574.

Nguyen, H-M., Nguyen, H-Q., Tran, K-N. and Vo, X-V. (2015) 'GeTFIRST: ontology-based keyword search towards semantic disambiguation', *International Journal of Web Information Systems*, Vol. 11, No. 4, pp.442–467.

Raza, M.A., Mokhtar, R., Ahmad, N., Pasha, M. and Pasha, U. (2019) 'A taxonomy and survey of semantic approaches for query expansion', *IEEE Access*, January, Vol. 7, pp.17823–17833.

Royo, J.A., Mena, E., Bernad, J. and Illarramendi, A. (2005) 'Searching the web: from keywords to semantic queries', *Third International Conference on Information Technology and Applications (ICITA'05)*, IEEE, pp.244–249.

Schuler, K.K. (2005) *VerbNet: A Broad-coverage, Comprehensive Verb Lexicon*, Thesis, University of Pennsylvania, USA, pp.1–146.

Selmi, W., Kammoun, H. and Amous, I. (2018) 'MeSH-based semantic query expansion', *2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA)*, IEEE, pp.1–8.

Shen, F. and Lee, Y. (2018) 'Biobroker: knowledge discovery framework for heterogeneous biomedical ontologies and data', *Journal of Intelligent Learning Systems and Applications*, Vol. 10, No. 1, pp.1–20.

Tanaka, K., Nakamura, S., Ohshima, H., Yamamoto, Y., Yanbe, Y. and Kato, M.P. (2010) 'Improving search and information credibility analysis from interaction between Web 1.0 and Web 2.0 content', *JSW*, Vol. 5, No. 2, pp.154–159.

Tran, T., Cimiano, P., Rudolph, S. and Studer, R. (2007) 'Ontology-based interpretation of keywords for semantic search', in Aberer, K. et al. (Eds.): *The Semantic Web. ISWC 2007, ASWC 2007. Lecture Notes in Computer Science*, Vol. 4825, Springer, Berlin, Heidelberg.

Vigneshwari, S. and Aramudhan, M. (2015) 'Social information retrieval based on semantic annotation and hashing upon the multiple ontologies', *Indian Journal of Science and Technology*, Vol. 8, No. 2, pp.103–107.

Wimalasuriya, D.C. and Dou, D. (2009) 'Using multiple ontologies in information extraction', *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, ACM, pp.235–244.

Yang, C. and He, B. (2016) 'A novel semantics-based approach to medical literature search', *2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 15–18 December, pp.1616–1623.

Yokoo, R., Kawamura, T. and Ohsuga, A. (2016) 'Semantics-based news delivering service', *International Journal of Semantic Computing*, Vol. 10, No. 4, pp.445–459.

Zhang, Z., Sun, R., Zhao, C., Wang, J., Chang, C.K. and Gupta, B.B. (2017) 'CyVOD: a novel trinity multimedia social network scheme', *Multimedia Tools and Applications*, Vol. 76, No. 18, pp.18513–18529.

Zhu, G. and Iglesias, C.A. (2018) 'Exploiting semantic similarity for named entity disambiguation in knowledge graphs', *Expert Systems with Applications*, July, Vol. 101, pp.8–24.