Expert Systems With Applications 53 (2016) 138–148

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Microblog semantic context retrieval system based on linked open data and graph-based theory

Fahd Kalloubi*, El Habib Nfaoui, Omar El beqqali

LIIAN laboratory, Sidi Mohamed Ben Abdellah University, Fez, Morocco

ARTICLE INFO

Keywords:
Information retrieval
Semantic similarity
Linked open data
DBpedia
Named entity linking
Graph centrality

ABSTRACT

Microblogging platforms have emerged as large collections of short documents. In fact, the provision of an effective way to retrieve short text presents a significant research challenge owing to several factors: creative language usage, high contextualization, the informal nature of micro blog posts and the limited length of this form of communication. Thus, micro blogging retrieval systems suffer from the problems of data sparseness and the semantic gap. This makes it inadequate to accurately meet users' information needs because users compose tweets using few terms and without query terms inside; thus, many relevant tweets will not be retrieved. To overcome the problems of data sparseness and the semantic gap, recent studies on content-based microblog searching have focused on adding semantics to micro posts by linking short text to knowledge bases resources. Moreover, previous studies use bag-of-concepts representation by linking named entities to their corresponding knowledge base concepts. However, bag-of-concepts representation considers only concepts that match named entities and supposes that all concepts are equivalent and independent. Thus, in this paper, we present a graph-of-concepts method that considers the relationships among concepts that match named entities in short text and their related concepts and contextualizes each concept in the graph by leveraging the linked nature of DBpedia as a Linked Open Data knowledge base and graph-based centrality theory. Furthermore, we propose a similarity measure that computes the similarity between two graphs (query-tweet) by considering the relationships between the contextualized concepts. Finally, we introduce some experiment results, using a real Twitter dataset, to expose the effectiveness of our system.

© 2016 Elsevier Ltd. All rights reserved.

1. Introduction

Microblogging platforms allow users to exchange short texts, such as tweets and user statuses in friendship networks. Microblogging has emerged as one of the primary social media platforms for users to post short messages and content of interest. Twitter is one of the most popular microblog service providers. In fact, it has attracted more than 500 million registered users and publishes 340 million tweets per day,[1] and many queries are issued each day (more than 1.6 billion search queries[2] in Twitter); however, determining the subject of an individual micro post can be nontrivial owing to several factors: creative language usage, the highly contextualized, informal nature of microblog posts, and the limited length of this form of communication. Therefore, these factors make micro blogging streams an invaluable sources for many types of analyses, including online reputation management, news and trend detection; targeted marketing and customer services (Boyd, Golder, & Lotan, 2010; Kwak, Changhyun, Hosung, & Sue, 2010; Manos, de Rijke, & Weerkamp, 2011; Xiangmin & Lei, 2013); these applications mainly analyze and utilize the wisdom of the crowds as a source of information rather than relying on individual tweets. Searching and mining microblog streams offer interesting technical challenges in many microblog search scenarios, and the goal is to determine what people are saying about concepts such as products, brands, and persons (Brendan, Ramnath, Bryan, & Noah, 2010). Microblogging retrieval systems suffer from the problems of data sparseness and the semantic gap owing to the length of microblog posts and their high contextualization. This makes it inadequate to accurately meet users' information needs because users compose tweets using different terms and without query terms inside; thus, many relevant tweets will not be retrieved. Most current microblogging Information Retrieval (IR) systems rely on a term-based model such as TF-IDF, BM25 and the probabilistic model (Lau, Li, & Tjondondronegoro, 2011).

* Corresponding author. Tel.: +212661860460.
E-mail addresses: fahd.kalloubi@usmba.ac.ma (F. Kalloubi), elhabib.nfaoui@usmba.ac.ma (E.H. Nfaoui), omar.elbeqqali@usmba.ac.ma (O. El beqqali).

[1] http://techcrunch.com/2012/07/30/analyst-twitter-passed-500m-users-in-june-2012-140m-of-them-in-us-jakarta-biggest-tweeting-city/ (last access: December 2015).

[2] http://techcrunch.com/2011/06/01/new-twitter-search-relevant/ (last access: December 2015).

http://dx.doi.org/10.1016/j.eswa.2016.01.020
0957-4174/© 2016 Elsevier Ltd. All rights reserved.

Term-based models are efficient in terms of computation performance, and the maturity of term weighting theories has made these models rampant. However, term-based approaches often suffer from the problems of polysemy and synonymy and are very sensitive to term use variation. Topical features such as phrases and named entities (e.g., person, location and proper nouns) are also often neglected (Lau et al., 2011). This problem is even more evident in microblogs owing to the amount of noise, which causes poor retrieval performance. To overcome this problem, many research efforts have been conducted to understand and model the semantics of individual microblog posts. Linking free text to knowledge resources, however, has received an increasing amount of attention in recent years. Starting from the domain of named entity recognition (NER), current approaches establish links not just to entity types but to the actual entities themselves (Meij, Weerkamp, & de Rijke, 2012; Xiaohua, Shaodian, Furu, & Ming, 2011). With more than 3.5 million articles, Wikipedia has become a rich source of knowledge and a common target for linking; automatic linking approaches using Wikipedia have achieved considerable success (He, de Rijke, Sevenster, van Ommering, & Qian, 2011; Meij, Bron, Hollink, Huurnink, & de Rijke, 2011). DBpedia as a central Linked Open Data dataset is a knowledge base created from Wikipedia by converting structured information (e.g., infobox) to a Resource Description Framework (RDF) data model.

The main contribution of the Semantic Web as a new form of Web content is the provision of meaning to computers with the utilization of ontology as a source of knowledge representation (Berners-Lee, Hendler, & Lassila, 2001). The Resource Description Framework (RDF) is introduced as an underlying framework to use ontology in the Web environment. The RDF data model treats each piece of information as a triple: subject–property–object (Lassila & Swick, 1999). The application of RDF for data representation has become a very popular means of representing data on the Web. Over time, more attention has been paid to it, and the term Linked Open Data (LOD) has been used to describe the network of data sources based on RDF triples for information representation (Shadbolt, Hall, & Berners-Lee, 2006). The main contribution of LOD contrary to hypertext Web is that entities from different sources/locations are linked to other related entities on the Web by the use of a Unified Resource Identifier (URI). This enables one to view the Web as a single global data space (Bizer, Heath, & Berners-Lee, 2009). In other words, hypertext Web connects documents in a naive way. However, in the Web of LOD, single information items are connected. As a result, DBpedia allows for better representation of structured data in a machine-understandable way.

We propose a graph-of-concepts method that considers the relationships between concepts and their related concepts and contextualizes each concept in the graph by leveraging the linked nature of DBpedia as a knowledge base. Furthermore, we propose a similarity measure that computes the similarity between two graphs (query-tweet). Our similarity measure considers the overlapping between named entities, which have been shown to obtain the best results in microblog searching (Tao, Abel, Hauff, & Houben, 2012a) and the relationships between the contextualized concepts in the graph.

The remainder of this paper is organized as follows. In Section 3, we present a method for enriching and adding semantic to micro posts by processing tweets and linking the entities extracted to LOD concepts using DBpedia as a knowledge base. This will help us represent micro posts as graphs-of-concepts by leveraging the linked nature of DBpedia to define our semantic context similarity measures. Equally important, we present our approach to contextualize all entities extracted by using graph-based centrality scoring. In Section 4, we introduce our algorithms for semantic context retrieval over tweets using the constructed graph-based context. Finally, in Section 5, we evaluate our system using a real dataset harvested from Twitter.

2. Related work

In this section, we review related works for adding semantic information to tweets and Information Retrieval systems over tweets.

2.1. Enriching and adding semantics to tweets

Linking text to a knowledge structure has received a great deal of attention, especially in the social Web because of the lack of semantics in such a complex structure. A rampant approach of linking text to concepts is to perform lexical matching between parts of text and the concept titles (Mendes, Passant, Kapanipathi, & P. Sheth, 2010). However, lexical matching suffers from many drawbacks, including ambiguity (polysemy and synonymy) and possible lack of specificity (less “meaningful” concepts are identified). Short texts have the characteristics of sparsity and noisiness owing to their limited length. Thus, when using the “bag of words” model to represent short text, contextual information is neglected and hence often leads to synonymy and polysemy problems (Tang, Wang, Gao, Hu, & Liu, 2012). To overcome the problem of data sparseness and the semantic gap in short text, various approaches have been proposed for adding semantics to text contained in tweets. (Somnath, Krishnan, & Ajay, 2007) developed a method to enrich short text representation with additional features from Wikipedia. This method used only the titles of Wikipedia articles as additional external features; it showed improvement in the accuracy of short text clustering. (Xiaohua et al., 2011) focused on NER in tweets and used a semi-supervised learning framework to identify four types of entities.

Meij et al. (2012)) proposed an approach to link n-grams to Wikipedia concepts based on various features. Their approach is divided into two steps; in the former, they generate a ranked list of candidate concepts for each n-gram in a tweet by applying a various type of features (*n*-grams features, concept features, tweet features). In the latter, they aim to improve precision by applying supervised machine learning. However, in our unsupervised approach, we consider only concept features by using different properties and contextualizing each concept by leveraging the linked nature of DBpedia as knowledge base to construct a weighted graph-of-concepts representation that depicts the context of each tweet by performing semantic linking and named entity resolution together, unlike their approach in which they suppose that all concepts are equal.

Abel, Gao, Houben, and Tao (2011) presented an approach that aims to contextualize tweets. After adding context, the authors use the tweets to profile Twitter users. Their approach is based on semantic enrichment with the news article’s content. Finally, the semantically enriched tweets are used for user modeling. Pertaining to the semantic enrichment, the authors use OpenCalais. Our approach proposed in Section 3 differs from their approach in the sense that we assume that the tweets are not related to news article, which makes our approach more general.

Tang et al. (2012) presented a framework for enriching short text for clustering purposes in which they perform multilanguage knowledge integration and feature reduction simultaneously through matrix factorization techniques.

Mendes et al. (2010) proposed Linked Open Social Signals, a framework that includes annotating tweets with information from Linked Data. Their approach is rather straightforward and involves either looking up hashtag definitions or lexically matching strings to recognize (DBpedia) entities in tweets.

Kwak, Changhyun, Hosung, and Sue (2010) showed that hashtags are good indicators to detect events and trending topics, and

(Mika & David, 2010) explored the use of hashtags in Twitter and the relation to (Freebase) concepts. Using manual annotations, they find that approximately half of the hashtags can be mapped to freebase concepts, most of which are named entities. In a few cases, more general hashtags are mapped to concepts. Assessors showed high agreement on the task of mapping tags to concepts. The authors make the assumption that hashtags are mainly used to "ground" tweets, an assumption that we adopt in our work, enabling us to add semantics to tweets with the use of hashtag definitions.

Guo, Chang, and Kiciman (2013) proposed a structural SVM method to address the problem of end-to-end entity linking in Twitter by considering mention detection and entity disambiguation together.

Guo, Qin, Liu, and Li (2013) proposed microblog entity linking by leveraging extra posts; they proposed a context expansion-based method to enrich the context of the tweet by leveraging the redundancy nature of microblog posts and using a graph-based method in the ranking step, they use Wikipedia as knowledge base. The main limitation of their approach is that they use extra posts, meaning that only redundant topics can be enriched.

2.2. Searching over tweets

The vast amount and rapidly increasing rate of information contained in Twitter have presented a great challenge for researchers in recent years by creating a need for a system to assist users in distilling useful information. Users search Twitter particularly for answers, timely information (e.g., news, events), people information and topical information (Efron & Winget, 2010). Various retrieval models have been proposed for searching microblogging streams. Matteo, Danilo, Gabriele, and Luca, 2011 proposed a user-based tree model for retrieving conversations from microblogging. A query-likelihood retrieval model can be used to identify subtopics for further browsing (Brendan, Connor, Krieger, & Ahn, 2010). Hashtags are also used to conduct searches for topic of interest, monitor event development and discuss trending issues. Hence, they have shown useful relevance feedback and query expansion (Efron, 2010). In fact, current microblogging retrieval is based on the bag-of-words (BOW) model, and each tweet is represented as a collection of preprocessed terms and a weight score is assigned to each term. This approach is suitable for tweets because it is topic specific and its performance is reliable without any advanced computation. However, it is also shown to be sensitive to noise (Naveed, Gottron, Kunegis, & Che Alhadi, 2011).

The length limit of tweets ensures that users use different terms and abbreviations to condense their ideas, which degrades retrieval performance. To overcome this issue, different strategies have been adopted. For instance, query expansion is used to capture more relevant terms from the initial query (Lau et al., 2011). In the same context, Efron, Organisciak, and Fenlon, 2012 proposed an approach to improve information retrieval (IR) for short texts based on aggressive document expansion by submitting documents as pseudo-queries and analyzing the results to learn about the documents themselves; they start from the hypothesis that short texts tend to be about a single topic, and we adopt the same hypothesis in our work, which enables us to develop our approach for context retrieval over tweets.

Topical features are widely used in clustering tasks (Abel, Celuk, Houben, & Siehndel, 2011; Xiang, Ruhua, Yan, & Bin, 2013). One attempt for using topic feature for retrieval tasks was proposed by Lau et al., 2011, who proposed a Twitter retrieval framework focused on using topic features by adopting a Vector Space Model (VSM) and TF-IDF weighing scheme combined with query expansion using pseudo-relevance feedback (PRF). The experimental results show that their approach performs better than existing term-based solutions. Furthermore, Lau, Tao, Tjondronegoro, and Li, 2012 proposed a topical feature discovery method in Twitter from both data mining and information retrieval perspectives using frequent pattern mining and pseudo-relevance feedback. The experimental results show that their method outperforms existing term-based information retrieval solutions.

Recent studies use semantic information to leverage contextual information, which is neglected in term-based methods. Carlos and Antonio, 2015 presented a system for topic discovery of tweets; unlike other approaches that extract topics from a tweet's content, they link hashtags to knowledge base concepts to identify the topics underlying a tweet set because hashtags can translate the main ideas of the post, which have been considered in our approach by involving hashtags definitions (see Section 3.1). Mohammed Nazim et al. 2013 proposed a model to enhance the accuracy of finding semantic relationships between tags using a co-occurrence matrix and WordNet, which is a lexical base; as reported in their future work, the use of a knowledge base can improve their system in the sense that they can identify the meaning of a tag with its related resource. In the context of microblogging search, linking micro posts to knowledge bases has received a great deal of attention owing to the limited length of this form of communication and its noisy nature. Tao et al., 2012a; Tao, Abel, Hauff, and Houben, 2012b conducted an in-depth analysis comparing the importance of different types of features and found that semantic features, mainly the overlap score between named entities within messages, plays a major role in determining the relevance of a tweet with respect to a query. Shangsong, Zhaochun, and Maarten, 2014 addressed a cluster-based fusion method in the task of retrieving micro posts; they found very limited clustering information by applying recent methods on microblog posts, and they proposed to link microposts to Wikipedia articles to enrich short text representation for clustering. Moreover, previous studies use concept-based representation using knowledge bases by linking short text to knowledge bases' concepts to form a bag of concepts (Kuang, Hui, & Diego, 2014; Taiki, Kazuhiro, & Kuniaki, 2014); however, they do not consider relationships among concepts and suppose that all concepts are equivalent and independent. Furthermore, they consider only concepts that match named entities in the text and not their related concepts. For example, we find the following post:

(1) Oracle has open a good perspectives to java by the acquisition of sun Microsystems #Oracle #Java

The content of the tweet (1) can be linked to three knowledge base concepts: (I) Oracle Corporation, (II) Java Programming Language and (III) Sun Microsystems. However, by using the bag-of-concepts method, post (1) is irrelevant to a query linked to "Java Virtual machine" or "Oracle Database", which is not the case.

Thus, we propose a graph-of-concepts method that considers the relationships among concepts and their related concepts and contextualizes each concept in the graph by leveraging the linked nature of DBpedia as a knowledge base and graph-based centrality. Furthermore, we propose a similarity measure that computes the similarity between two graphs (query-tweet). Our similarity measure considers the overlap between named entities, which have been shown to obtain the best results in microblog searching (Tao et al., 2012a) and the relationships among the contextualized concepts in the graph. Moreover, our similarity measure prioritizes concepts that match named entities in the text because they have a high weight; for example, post (1) will be more relevant to a query linked to "Oracle Corporation" than a query linked to "Oracle Database".

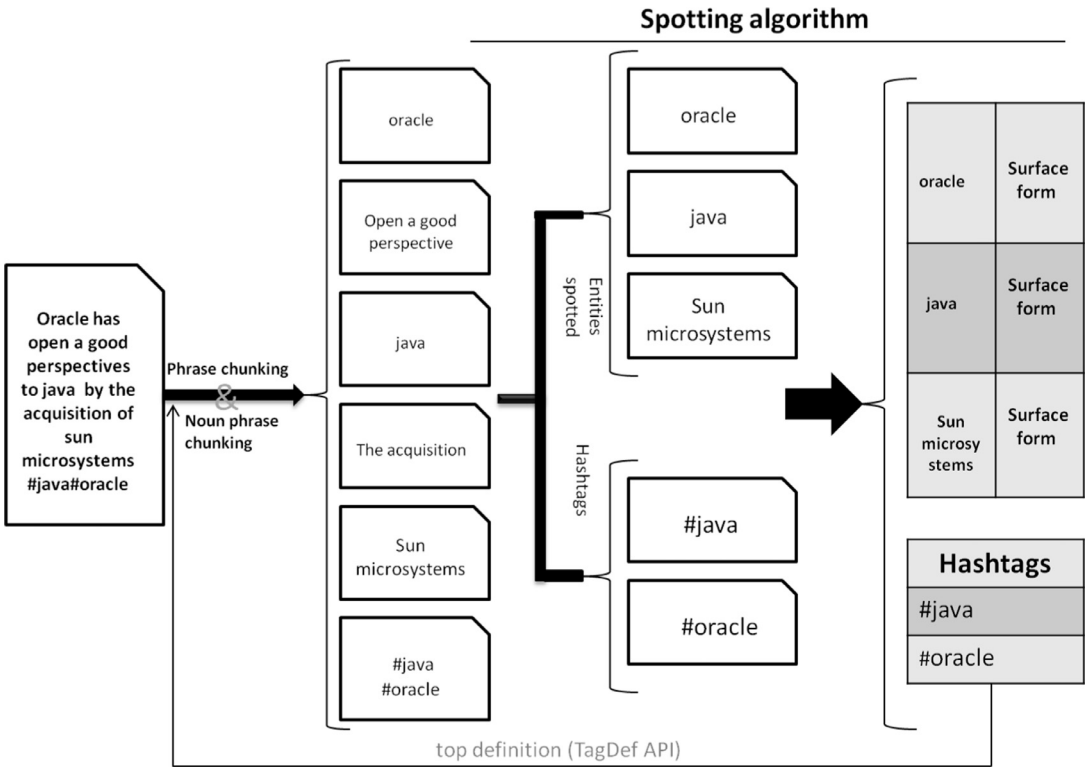

Fig. 1. Method for generating surface forms.

3. Proposed method

3.1. Proposed approach for concept extraction and weighting

Textual data in social media have the problems of data sparseness and the semantic gap. One effective way to solve these problems is to integrate semantic knowledge, which has been found to be useful in addressing the semantic gap (Tang et al., 2012). In fact, DBpedia is receiving more attention in recent years as one of the central datasets in the LOD cloud. For each entity, DBpedia defines a globally unique identifier that can be referenced over the Web in the RDF description of an entity. As an NER tool, the DBpedia spotlight project annotates documents automatically by using DBpedia resources. Surface forms are identified with a preprocessing step and stored in the DBpedia Lexicalizations Dataset[3] (DLD); surface forms in this dataset are identified by entities labels, redirects and disambiguation in DBpedia. As its context similarity measure, DBpedia spotlight uses term frequency (TF) and an inverse candidate frequency (ICF) to matches the correct resources in the disambiguation step (Pablo, Max, Andrés, & Christian, 2011). In our context, and given the nature of the query in the microblogging platforms, we define a novel method for extracting relevant entities from tweets and adding semantics to microblogging posts by contracting surface forms that depict the context in which a microblogging post occurs. Hence, we use it to construct a graph of contextualized and weighted entities for each tweet, based on the graph centrality theory. In the next sections, we explain all steps that we follow to construct the graph of contextualized and weighted DBpedia concepts.

3.1.1. Surface form construction

In the context of the text corpus, different techniques in NLP can be employed to extract features. The only requirement is that the extracted features could be informative to cover the key subtopics described in the short texts. Shallow parsing has been proposed previously in short text clustering (Xia, Nan, Chao, & Tat-Seng, 2009). Its aim is to divide the text into a series of words that together compose a grammatical unit (i.e., phrases chunking). We use Shallow Parser to detect key phrases that are often named entities in microblogging platforms. Our method is depicted as follows: In each step, the selected word set (phrase) is searched. If a surface form is found, then it is added to the list of spotted surface forms, and the next phrase is searched; otherwise, if no match is found, we use NP-chunking (Noun Phrase chunking), and each noun phrase is searched separately in the knowledge base until a match is found. Fig. 1 shows an example of processing applied to a tweet. To construct the surface forms, we use a similar approach to DBpedia Spotlight Project. Owing to the nature of microblogging search scenarios, we use many types of sources in DBpedia to extract surface forms. These are: rdfs:label, foaf:name, dbpprop:officialName, dbpprop:name, foaf:givenName, dbpprop:birthName, dbpprop:alias and disambiguation that is used to group entities that have various meanings for the same title and redirects pages used to show alternative titles of a given entity (DBpedia ontology properties dbont: wikiPageDisambiguates[4] and dbont:wikiPageRedirects[5]). The surface forms formed using these sources will be used for the construction of the graph of interlinked entities that depict the context of the document.

3.1.1.1. Stop words. In our method, we skip stop words, meaning that each phrase is searched without removing stop words, which do not have any meaning. They are meaningful with named entities.

[3] http://dbpedia.org/Lexicalizations (last access: December 2015).

[4] http://dbpedia.org/ontology/wikiPageDisambiguates (last access: December 2015).

[5] http://dbpedia.org/ontology/wikiPageRedirects (last access: December 2015).

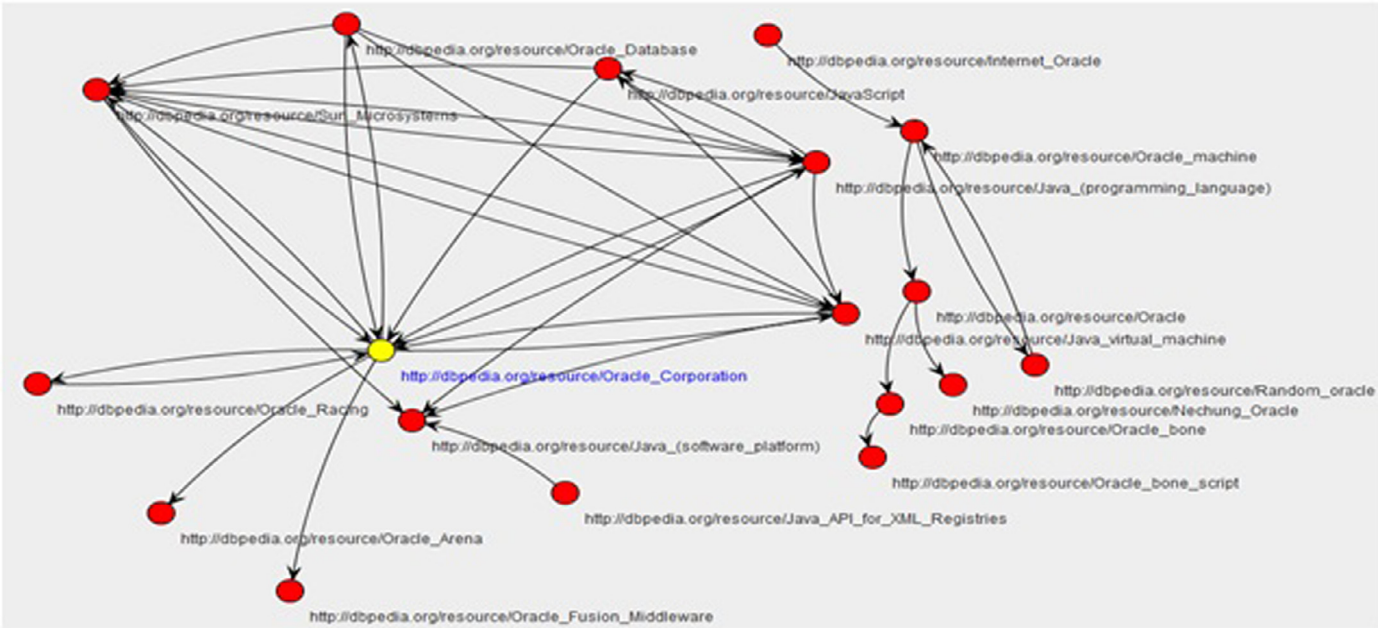

Fig. 2. A partial graph of the tweet above.

3.1.1.2. Hashtags. A hashtag is a string of characters preceded by the hash (#) character and is used to build communities around particular topics. To outside observers, the meaning of a hashtag is usually difficult to analyze because they consist of short, often abbreviated or concatenated concepts (e.g., #MSM2013). In our work, we consider hashtags to be related to the named entities spotted. Thus, they are processed separately using the TagDef API[6] to look up the highest-ranked hashtag definition and process it. Therefore, it more contextualizes the tweet. For instance, the tweet "I'm following the arab spring …#Mubarak#arab" does not provide enough information about the user's tweet without processing its related hashtags.

After the construction of surface forms belonging to a tweet, we search for "dbont:wikiPageWikiLink" links between entities in the surface forms and use entities and the links found among the entities to form the graph.

Hence, let us denote *S (T)* as the set of spotted surface forms for the tweet *T* that is to be annotated:

$$S(T) = \{s_1, s_2, \ldots, s_m\}$$

E(s) denotes the set of all entities from DBpedia for each surface form s_i in *S (T):*

$$E(s_i) = \{e_1, e_2, \ldots, e_k\}, \text{ for } s_i\, S(T)$$

The set of all entities for the tweet T is then *E (T)*:

$$E(T) = \bigcup E(s_i),\ s_i\, S(T)$$

By using the property dbont:wikiPageWikiLink, we check whether there is a link between any two entities in *E(T)*; the link is then added to the relationship set *R(T)*. We then build a graph of spotted entities and relationships found as a directed edge. The graph is a 3-tuple construct $G = (E(T), R(T), f)$, where *f is* a centrality factor that we use to attribute a weight to the nodes.

Fig. 2 shows a partial graph for the example above; we used Java Universal Network Graph (JUNG)[7] to construct and visualize the graph of entities and relationships.

[6] http://tagdef.com/ (last access: December 2015).

[7] http://jung.sourceforge.net/ (last access: December 2015).

3.1.2. Concept weighting

After the construction of the graph of entities and relationships, we must detect how central a node is in the graph, which is a crucial task in our method, based on graph centrality scoring methods that have been successfully applied because they consider the relationships between nodes. For example, the node "oracle_corporation" (the yellow one) in the partial graph above (Fig. 2) is the best candidate concept to annotate the entity "oracle" in this context. For performance reasons, we neglect isolated nodes because this type of node does not give us any information about their context.

We modified the conventional closeness centrality scoring method to consider the number of related nodes. As our graph is directed, not all nodes are reachable from other nodes. For every node, we then calculate the shortest path using Djikstra's shortest path algorithm from that node to all its successors in the graph.

Our customized centrality factor is then calculated by dividing the total number of successors plus the total number of predecessors of the node to the sum of the lengths of the shortest paths to its successors. Formally, the node N_a can be weighted using the following formula.

$$f(N_a) = \frac{\Pr(N_a) + Sc(N_a)}{\sum_{N_b \epsilon N} sh(N_a, N_b)} \quad (1)$$

$\sum_{N_b \epsilon N} s(N_a, N_b)$ Is the sum of the shortest distances between N_a and the reachable nodes from N_a.

Pr_{N_a}: Is the number of predecessors' nodes of N_a.

Sc_{N_a}: Is the number of successor's nodes of N_a.

The centrality factor is used to weight the node as depicted in the following formula.

$$W(N_a) = f(N_a)^{*} inlinks(N_a)^{*} outlinks(N_a)^{*} k \quad (2)$$

$f(N_a)$ is our customized centrality factor for node N_a; $inlinks(N_a)$ and $outlinks(N_a)$ are the number of incoming and outgoing links, respectively, for node N_a. *k is* a constant depending on the type of entity N_a. *k* is set to 1 if the entity is from disambiguation pages and 6 if the entity is from other properties. The attributed value of *k* is determined by the performance of our tests. Disambiguation pages match a high number of entities, which are

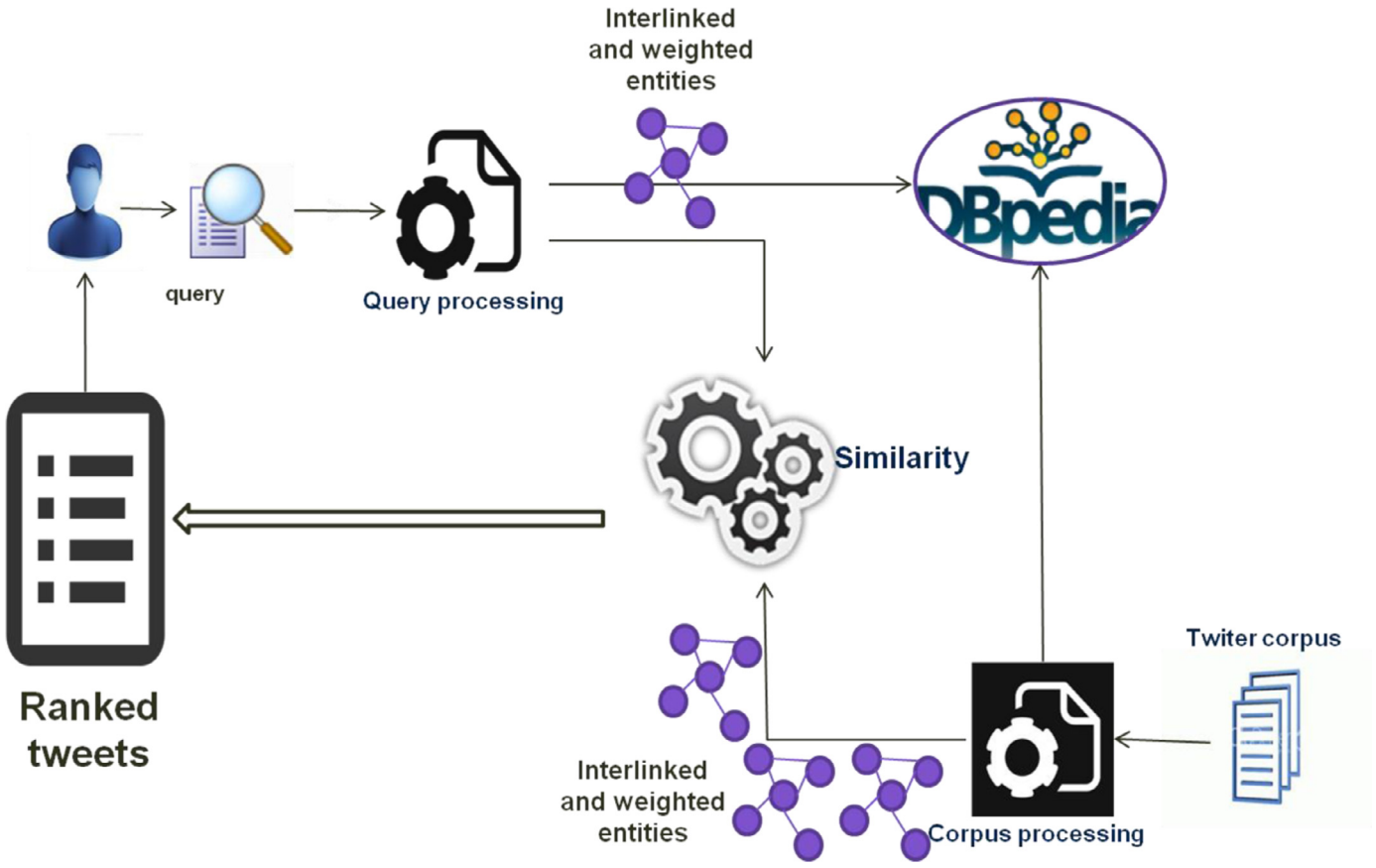

Fig. 3. System of semantic retrieval over tweets.

unrelated and might negatively affect the disambiguation. Therefore, entities from disambiguation pages are penalized with less weight.

After this step, we obtain a set of interlinked and weighted knowledge base entities (concepts) that represent each short document in the corpus. This enables us to define our semantic context similarity algorithm.

3.2. Proposed algorithms for context similarity retrieval

In this section, we detail the architecture of our system as shown in Fig. 3, and we define our algorithms of semantic context similarity between the user request and the corpus of microblogging posts as detailed in the next section.

3.2.1. System Framework

Given a microblog corpus, we first process it to extract the graph of interlinked and weighted entities as depicted in the previous section; when a user request is submitted, we process it to extract the graph of interlinked and weighted entities using DBpedia as a knowledge base. We define two types of similarity measures: the first is a local similarity measure that defines the similarity between two entities in terms of the context in which they occur, and the second is a global similarity measure that defines the similarity between the user request and the document (tweet) represented by their graphs of interlinked and weighted entities.

3.2.2. Semantic context retrieval

The problem of measuring the semantic relatedness and similarity of two concepts can be stated as follows: given two concepts A and B, determine a measure f (A, B) that expresses the semantic proximity between them. The notion of semantic similarity is associated with taxonomic (is-a) relations between concepts, whereas semantic relatedness represents more general classes of relations (André, João, Seán, Edward, & João, 2011). In this paper, because our graph is weighted, we are concerned with defining a new semantic similarity measure that considers the weight and the link between nodes; we assume that the semantics between two concepts are defined by their connections to their attached entities. As mentioned above, short texts tend to be about a single topic, and by considering this property and knowing that the user request is about a single topic, we give our model of retrieval the ability to capture semantic similarity between user request and all tweets in the corpus by considering the context of the request and the document as depicted bellow (Algorithm 1).

Algorithm 1: SemanticContextSearch

Input
Q: user query
G= (G_i,d_i): a set of Graphs of interlinked and weighted entities G_i for each document d_i
Begin
GQ=ConstructGraph(Q)// we construct the graph of interlinked and weighted entities of the query
Foreach $G_i \in G$
  Foreach $URI_i \in GQ$
    Foreach $URI_j \in G_i$
      If (URI_i == URI_j)
        CE=CE+1// the count of similar entities matched
        LS_i =ComputeLocalSimilarity(URI_i,URI_j) //see algorithm 2
      End
    End
  End
  Sim_i=ComputeGlobalSimilarity(LS_i,CE)
  Store (d_i,Sim_i)
End
End

In this algorithm, we define our approach for similarity between the request and the corpus. We obtain all content matched as similar to the request with a similarity score.

We also define two types of similarity measures: a local one that determines the similarity between two entities by considering the nature of the connections that link these entities and their attached entities, which checks the similarity at a context level, and a global similarity between the request and the content:

$$GSM = \frac{\sum LSM_i}{CE} \tag{3}$$

where

LSM_i: the local similarity measure between two URIs, which are matched as similar

CE: the count of the common similar entities between the request and the document

Moreover, and as shown in algorithm 2, we define our local similarity measure that considers the nature of the connection and the weight of similar entities matched in the process:

$$LSM = \left(\frac{CPC_{URI_d,URI_q}}{Pred_{URI_d} + pred_{URI_q}} + \frac{CSC_{URI_d,URI_q}}{succ_{URI_d} + succ_{URI_q}} \right) * \frac{W_{URI_d}}{W_{URI_q}} \tag{4}$$

where

CPC_{URI_d,URI_q}: is the common predecessor count between URI_d and URI_q.

CSC_{URI_d,URI_q}: is the common successor count between URI_d and URI_q.

$Pred_{URI_d}$: is the predecessor count of the entity URI_d belonging to a document.

$pred_{URI_q}$: is the predecessor count of the entity URI_q belonging to a user query.

$succ_{URI_d}$: is the successor count of the entity URI_d belonging to a document.

$succ_{URI_q}$: is the successor count of the entity URI_q belonging to a user query.

W_{URI_d}: is the weight of the document node URI_d defined by our customized centrality factor.

W_{URI_q}: is the weight of the query node URI_q defined by our customized centrality factor.

Algorithm 2: ComputeLocalSimilarity(URI_q,URI_d) computes the similarity between two URIs

```
Input
   URIq: a URI from the request
   URId: a URI from the document
    Gq: graph of the request (interlinked and weighted entities)
    Gd: graph of the document (interlinked and weighted entities)
Output
   Sim: local similarity between URIq and URId
Begin
Foreach URIi ∈ Successors (URIq)
   Succq=SuccessorsCount(URIi)
   Foreach URIj ∈ Successors (URId)
      Succd=SuccessorsCount(URIj)
      CSC=computeCommonSuccessorsCount(URIi, URIj)
   End
End
Foreach URIi ∈ Predecessors (URIq)
   Predq=PredecessorsCount(URIi)
   Foreach URIj ∈ Predecessors (URId)
      Predd=PredecessorsCount(URIj)
      CPC=computeCommonPredecessorsCount(URIi, URIj)
   End
End
Sim=LocalSim(URIq,URId,Succq, Succd, Predq, Predd,CSC,CPC) //see equation 4
End
```

Table 1
Queries to evaluate our system (short queries are highlighted in bold).

QueryID	Queries
Topic 1	The social media and the customer relationship in a company
Topic 2	Egyptian plane makes unscheduled Landing in Athens due to possible Bomb Threat
Topic 3	Barack Obama announces that Osama bin Laden was killed in a military operation
Topic 4	The wikileaks twitter account
Topic 5	The online events academy
Topic 6	Perspective of social media in education
Topic 7	Captain's share by Nathan Lowell
Topic 8	LinkedIn starts social news services
Topic 9	Recommendation for a new bank and wamu services
Topic 10	**Google buzz**
Topic 11	**Osama Bin laden**
Topic 12	**United States Army**
Topic 13	**Captain America**
Topic 14	What's new in the world of Apple? Samsung?
Topic 15	**Football team**
Topic 16	The president Barak Obama and his speech about Osama Bin Laden
Topic 17	**The U.S department of labor**
Topic 18	**The U.S justice department**
Topic 19	**Pirates of the Caribbean**
Topic 20	Which films wins the Palme d'Or in Cannes Film Festival?
Topic 21	**The Tree of Life**
Topic 22	**The Billboard Music Awards**
Topic 23	**The Country Music**
Topic 24	**UEFA Champions League**
Topic 25	Barcelona beats Manchester United and wins Champions League final
Topic 26	NHL Western Conference Finals
Topic 27	Oprah winfrey ends her twenty five year run of The Oprah Winfrey Show
Topic 28	**Indianapolis 500: Dan Wheldon**
Topic 29	**Indianapolis**
Topic 30	Obama rejects Bush Iraq withdrawal plan
Topic 31	Iraq Oil Industry
Topic 32	UK Firms Invited to Explore Opportunities in Iraq
Topic 33	Security Council extends UN mission in Iraq
Topic 34	Google Stays in China And Baidu Keeps on Winning
Topic 35	The Turkey Kurdish Conflict
Topic 36	The political structure of Turkey
Topic 37	Groupon Launches iPhone Application in the UK
Topic 38	**Sony**
Topic 39	**tour de France**
Topic 40	**Lance Armstrong**
Topic 41	Thor Hushovd just won another stage in Tour de france
Topic 42	**BBC News**
Topic 43	Explosion hits government offices in Oslo
Topic 44	A government building has been damaged in Norway's capital
Topic 45	BlackBerry PlayBook Available in the US and Canada
Topic 46	**The FIFA world cup**
Topic 47	**The BMW cars**
Topic 48	Bin Hammam accused of buying 2022 World Cup
Topic 49	Egyptians fired up their revolt against Hosni Mubarak
Topic 50	Egypt tells Iran to stay out of Arab's business

4. Results and discussion

4.1. Dataset

We evaluate the performance of our microblog retrieval using a set of 50 user queries manually selected and judged by an expert using pooling techniques (Christopher, Prabhakar, & Hinrich, 2008). The dataset used in the evaluation is a subset (Li, Wang, Deng, Wang, & Chen-Chuan, 2012) of the Twitter corpus; it contains 50 million tweets. The dataset[8] was collected in May 2011. Tweets are filtered by language, and only English tweets are used in the experiment using an existing language detection library.[9]

[8] https://wiki.cites.illinois.edu/wiki/display/forward/Dataset-UDI-TwitterCrawl-Aug2012 (last access: December 2015).

[9] https://github.com/shuyo/language-detection (last access: December 2015).

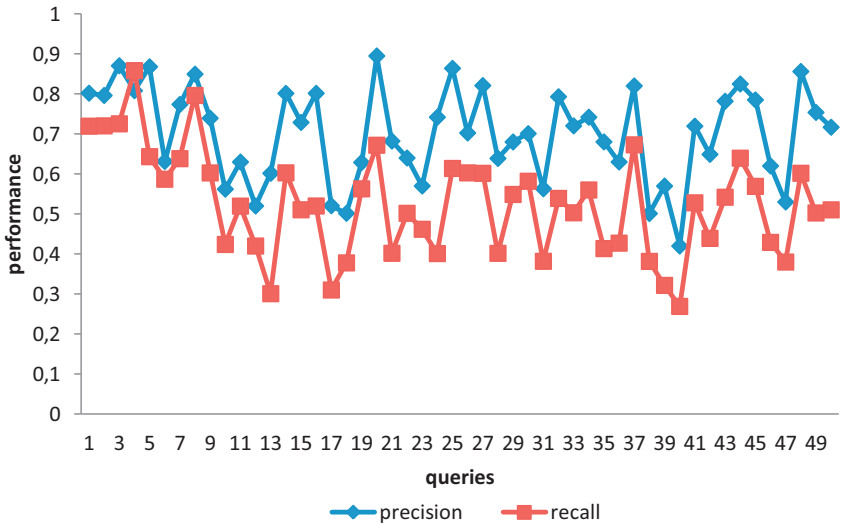

Fig. 4. Performance of our system in terms of precision and recall.

Users refer to Twitter for information about breaking news and real-time events (Lau et al., 2011). Thus, the selected queries are heterogeneous, containing news and events occurring in the period in which the dataset was collected. Table 1 shows the 50 topics that we used to test our system.

4.2. Evaluation measures

To measure the effectiveness of our system, we use in addition of precision and recall, as a conventional measure, three other standard measures for ranked systems: *Mean Average Precision (MAP), Precision at rank 30 (P@30)* and *Reciprocal Precision (R-Prec),* which are defined as follows:

- *MAP*: Mean average precision relies on the precision at each point when a relevant tweet is retrieved and is calculated as follows:

$$MAP = \frac{1}{N} \cdot \sum_{j=1}^{N} \frac{1}{Q_j} \cdot \sum_{i=1}^{Q_j} P(T_i)$$

where Q_j denotes the number of relevant tweets for query j, N is the number of queries, and $P(T_i)$ is the precision of the relevant tweets in the top i tweets. *MAP* provides a single-figure measure of quality across recall levels.

- *P@30*: The precision of the top 30 tweets, which has the advantage of not requiring any estimate of the size of the set of relevant tweets, and the total number of relevant tweets to the query has a strong influence on it.
- *R-Prec:* In this measure, we examine the top |*Rel*| tweets returned by the system (|*Rel*| are the relevant tweets for a query) and find r tweets that are relevant; then, *R-Prec* is calculated as $^{r}/_{|Rel|}$. Furthermore, the recall of this tweet set is also $^{r}/_{|Rel|}$.

These three measures are used as the main measures to evaluate our ranked system.

4.3. Performance of our system on short and long queries

We have tested our system on all of the queries above (Table 1) to show its effectiveness. In fact, the experimental results show that our system performs better in long queries than a single keyword search in terms of precision and recall.

As shown in Fig. 4, the overall performance of our system is promising in terms of precision and recall and proves that it is able to match a large number of query concepts because both queries and tweets are semantically represented and enriched. Furthermore, the main contribution of our system is optimal precision of long queries in the sense that it makes them more contextualized and enriched. These long queries better translate the user's needs. Unlike long queries, for short queries, our system shows a reasonable result in terms of precision and recall in the sense that it penalizes the absence of context in the query because short queries can't translate the users' needs properly; thus, they are not well enriched as shown in Fig. 4.

In terms of recall, our designed system shows optimal results, especially for the long queries. With regard to the lower performance for short queries, in addition to its lack of contextualization, we argue that short queries can be found in a tweet as an optional topic. For example, for query topic 10, a tweet such as "I'm using Google buzz it is a good social network compared to MySpace" results in a good similarity score because concepts that match the "Google buzz" mention are more central, whereas this is not the case with the post "Justin Bieber announces the title of its next album in his Google buzz account", which translate into a bad similarity score.

To determine whether the performance differences of our system between short and long queries is real or they simply due to chance, we conducted a deep analysis, using standard evaluation measures for ranked systems, by testing our system on short queries and long queries as shown in Table 2. Furthermore, based on the results produced by each set, we performed two-sided paired randomization tests (Smucker, Allan, & Carterette, 2007) to prove the statistical significance of the performance of our system on long and short queries.

Table 2
Performance of our system on short and long queries; * denotes a statistically significant difference between long and short query performance (two-sided paired randomization tests: p-value < 0.05).

	System performance		
	MAP	R-Prec	P@30
Short queries	0.55	0.531	0.695
Long queries	0.688*	0.674*	0.764*

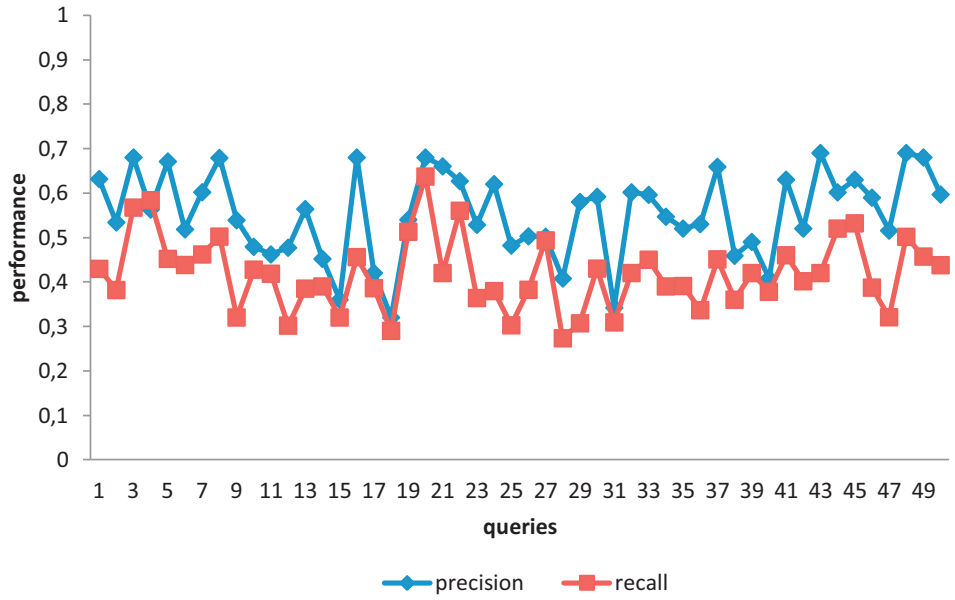

Fig. 5. Performance of our system in terms of precision and recall using the degree centrality factor.

Table 3
Performance of our system over 50 topics; * denotes a statistically significant difference from the baseline using two-sided paired randomization tests with a p-value < 0.05.

	System performance		
	MAP	R-Prec	P@30
Baseline	0.593	0.582	0.64
Our system	0.638	0.601	0.719*

As shown in Table 2, there is a significant difference between short and long queries, which indicates the effectiveness of our graph-of-concepts representation because concepts in long queries are more enriched and contextualized to match a large number of concepts in micro posts.

4.4. Performance of our system compared to baseline

To prove the effectiveness of our proposed centrality factor, we use the degree centrality factor as a baseline, which has been shown to obtain optimal results in word sense disambiguation (Navigli & Lapata, 2010; Sinha & Mihalcea, 2007). Fig. 5 shows the lower performance of our system when using the degree centrality factor instead of our customized centrality factor; we argue that this lower performance is because the degree centrality factor considers only the incoming and the outgoing edges for a node and all nodes in the graph, whereas our graph is directed, so not all nodes are reachable from other nodes. Furthermore, in some cases, the constructed graph can contain sub-graphs (see Fig. 2), which makes degree centrality factor unable to calculate the real degree centrality of a node.

Moreover, our customized centrality factor computes the centrality of a node based on its predecessor and successor nodes, giving our customized centrality factor the ability to calculate the real centrality factor if the constructed graph is disconnected (contains sub-graphs). In other words, our customized centrality factor can compute the centrality of a node based on their connected nodes. Also, we can add an adaptive pre-searching step to choose the adequate centrality factor based on the graph nature. Indeed, in some cases where the constructed query's graph is connected the degree centrality factor performs our customized centrality factor.

Table 3 shows the performance of our system and the baseline system using Mean Average Precision (MAP), Recall Precision (R-PREC) and the official Precision when 30 tweets are retrieved (P@30). The results show that our system using the personalized centrality factor improves over the baseline using the degree centrality factor, with a significant difference in terms of the P@30 measure. P@30 was used as the official measurement in the TREC Microblog ad-hoc task. In the case of other evaluation measures, there were some improvements, but they were not significant.

Moreover, this performance shows that our graph-of-concepts representation using the personalized centrality factor yields straightforward results because it better contextualizes each concept in the graph toward other related concepts, giving our context retrieval system the ability to best match users' needs. Furthermore, a combination between semantic similarity and lexical similarity can improve our system because it is unable to accurately meet users' needs if their queries do not contain named entities; in other words, the fewer named entities a query contains, the less our system can meet the users' needs. Equally important, the incorporation of other knowledge bases can significantly improve our system.

4.5. Implementation

The evaluation was conducted on the dataset described in Section 4.1, using an evaluation framework that we implemented in java. The evaluation was performed on a 4-core machine with 3 GB of RAM. We used the Apache OpenNLP[10] library for the task of surface form construction (i.e., phrase chunking). We have used version 3.9 of DBpedia,[11] which is open to everyone, and we used it to perform our tests. DBpedia English dump for page links were loaded into Virtuoso Database Server 1.6 to perform SPARQL queries to extract the links among entities.

[10] https://opennlp.apache.org/ (last access: December 2015).
[11] http://oldwiki.dbpedia.org/Downloads39 (last access: December 2015).

5. Conclusion and future work

In this paper, we proposed an approach for micro post retrieval in microblogging platforms to improve retrieval effectiveness in such corpora. We presented a graph-of-concepts representation method. Previous studies use concept-based representation using knowledge bases by linking short text to knowledge bases' concepts to form a bag of concepts. Moreover, they consider only concepts that match named entities in the text and not their related concepts, and they suppose that all concepts are equivalent and independent. Thus, the proposed method considers the relationships among concepts and their related concepts and contextualizes each concept in the graph by leveraging the linked nature of DBpedia as a Linked Open Data knowledge base and graph-based centrality. Furthermore, we proposed a similarity measure that computes the similarity between two graphs (query-tweet). Our similarity measure considers the overlap between named entities, which have been shown to obtain optimal results in microblog searching, and the relationships among the contextualized concepts in the graph.

The experimental results have shown that our system achieves good results in terms of precision, recall and standard information retrieval evaluation measures, and it has achieved significant results on long queries compared to short queries. Furthermore, to prove the effectiveness of the proposed centrality factor, we compare its performance with the degree centrality factor, and the experimental results show that our factor outperforms it with significant results. However, our system is unable to accurately meet users' needs if the query does not contain named entities; in other words, the fewer named entities a query contains, the less our system can meet the user's need. A combination between lexical similarity and semantic similarity can solve this issue. Equally important, the incorporation of other knowledge bases can significantly improve our system.

For future work, we plan to improve our content-based system by addressing the problem of social information retrieval as a novel research area that bridges information retrieval and social network analysis to improve information access. We also plan to investigate a real-time feature to address the problem of dynamicity in microblogging platforms because users care about the quality of information as much as its fresh nature and the quality of the source. In the same context, we intend to improve our system by leveraging personalization techniques based on users' personal interests.

Equally important, user profile modeling plays an important role in the tasks of personalization and socialization in information retrieval in such platform; thus, we first plan to address the problem of dynamic user profile modeling in microblogging platforms.

References

Abel, F., Celuk, I., Houben, G.-J., & Siehndel, P. (2011). Leveraging the semantics of Tweets for adaptive faceted search on Twitter. In *ISWC'11 Proceedings of the 10th international conference on The semantic web, Part I* (pp. 1–17).

Abel, F., Gao, Q., Houben, G.-J., & Tao, K. (2011). Semantic enrichment of Twitter posts for user profile construction on the social web. In *ESWC'11 Proceedings of the 8th extended semantic web conference on The semanic web*.

André, F., João, G. O., Seán, O., Edward, C., & João, C. P. (2011). Querying linked data using semantic relatedness: a vocabulary independent approach. In *NLDB'11 Proceedings of the 16th international conference on Natural language processing and information systems,Springer-Verlag Berlin* (pp. 40–51).

Berners-Lee, T., Hendler, J., & Lassila, O. (2001). The semantic web. *Scientific American Magazine*, 29–37. https://www.w3.org/People/Berners-Lee/Publications.html

Bizer, C., Heath, T., & Berners-Lee, T. (2009). Linked data-the story so far. *International Journal on Semantic Web and Information Systems, 5*(3), 1–22.

Boyd, D., Golder, S., & Lotan, G. (2010). Tweet, tweet, retweet: conversational aspects of retweeting on twitter. In *HICSS '10 Proceedings of the 2010 43rd Hawaii International Conference on System Sciences* (pp. 1–10).

Brendan, O., Ramnath, B., Bryan, R. R., & Noah, A. S. (2010). From tweets to polls: linking text sentiment to public opinion time series. In ICWSM '10 Fourth International Conference on Weblogs and Social Media.

Brendan, O'Connor, Krieger, M., & Ahn, D. (2010). Tweetmotif: Exploratory search and topic summarization for twitter. *ICWSM*.

Carlos, V., & Antonio, M. (2015). Unsupervised topic discovery in micro-blogging networks. *Expert Systems with Applications, 42*(17–18), 6472–6485.

Christopher, D. M., Prabhakar, R., & Hinrich, S. (2008). *Introduction to Information Retrieval*. New York, USA: Cambridge University Press.

Efron, M. (2010). Hashtag retrieval in a microblogging environment. In Proceeding of the 33rd international ACM SIGIR Conf. on Research and Development in Information Retrieval. *Geneva, Switzerland: Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*.

Efron, M., & Winget, M. (2010). Query polyrepresentation for ranking retrieval systems without relevance judgments. *Journal of the American Society for Information Science and Technology, 61*(6), 1081–1091.

Efron, M., Organisciak, P., & Fenlon, K. (2012). Improving retrieval of short texts through document expansion. In *SIGIR '12 Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*.

Guo, S., Chang, M.-W., & Kiciman, E. (2013). To link or not to link? a study on end-to-end tweet entity linking. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Atlanta, Georgia,June. Association for Computational Linguistics* (pp. 1020–1030).

Guo, Y., Qin, B., Liu, T., & Li, S. (2013). Microblog entity linking by leveraging extra posts. In Conference on Empirical Methods in Natural Language Processing. *ACL*.

He, J., de Rijke, M., Sevenster, M., van Ommering, R., & Qian, Y. (2011). Generating links to background knowledge: a case study using narrative radiology reports. In *CIKM '11 Proceedings of the 20th ACM international conference on Information and knowledge management* (pp. 1867–1876).

Kuang, L., Hui, F., & Diego, R. (2014). Concept based tie-breaking and maximal marginal relevance retrieval in microblog retrieval. In *Proceedings of Test REtrieval Conference*.

Kwak, H., Changhyun, L., Hosung, P., & Sue, M. (2010). What is Twitter, a social network or a news media? In *WWW '10 Proceedings of the 19th international conference on World wide web* (pp. 591–600).

Lassila, O., & Swick, R. (1999). *Resource description framework (RDF) model and syntax specification*. Retrieved from http://www.w3.org/TR/1999/REC-rdf-syntax-19990222.

Lau, C. H., Li, Y., & Tjondondronegoro, D. (2011). Microblog retrieval using topical features and query expansion. In Proceedings of TREC.

Lau, C. H., Tao, X., Tjondronegoro, D., & Li, Y. (2012). Retrieving information from microblog using pattern mining and relevance feedback. In Proceedings of the Third International Conference on Data and Knowledge Engineering, ICDKE 2012 (pp. 152–160). Springer.

Li, R., Wang, S., Deng, H., Wang, R., & Chen-Chuan, C. (2012). Towards social user profiling: unified and discriminative influence model for inferring home locations. In *KDD '12 Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 1023–1031).

Manos, T., de Rijke, M., & Weerkamp, W. (2011). Linking online news and social media. In *WSDM '11 Proceedings of the fourth ACM international conference on Web search and data mining*.

Matteo, M., Danilo, M., Gabriele, N., & Luca, R. (2011). conversation retrieval from twitter. In *Proceedings of the 33rd European Conference on IR Research, ECIR 2011* (pp. 780–783).

Meij, E., Bron, M., Hollink, L., Huurnink, B., & de Rijke, M. (2011). Mapping queries to the linking open data cloud: a case study using DBpedia. *Web Semantics: Science, Services and Agents on the World Wide Web, 9*(4), 418–433.

Meij, E., Weerkamp, W., & de Rijke, M. (2012). Adding semantics to microblog posts. In *Proceedings of the fifth ACM international conference on Web search and data mining* (pp. 563–572).

Mika, P., & David, L. (2010). Making sense of twitter. In *ISWC'10 Proceedings of the 9th international semantic web conference on The semantic web, Part I* (pp. 470–485).

Mohammed Nazim, U., Trong Hai, D., Ngoc Thanh, N., Xin-Min, Q., & Geun Sik, J. (2013). Semantic similarity measures for enhancing information retrieval in folksonomies. *Expert Systems with Applications, 40*(5), 1645–1653.

Mendes, P. N., Passant, A., Kapanipathi, P., & Sheth, A. P. (2010). Linked open social signals. In *WI-IAT '10 Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology. 01* (pp. 224–231). ACM.

Naveed, N., Gottron, T., Kunegis, J., & Che Alhadi, A. (2011). searching microblogs: coping with sparsity and document quality. In *CIKM '11 Proceedings of the 20th ACM international conference on Information and knowledge management* (pp. 183–188).

Navigli, R., & Lapata, M. (2010). An experimental study of graph connectivity for unsupervised word sense disambiguation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 32*(4), 678–692.

Pablo, N. M., Max, J., Andrés, G.-S., & Christian, B. (2011). DBpedia spotlight: shedding light on the web of documents. In *I-Semantics '11 Proceedings of the 7th International Conference on Semantic Systems* (pp. 1–8). ACM.

Shadbolt, N., Hall, W., & Berners-Lee, T. (2006, May). The semantic web revisited. *IEEE Intelligent Systems, 21*(Issue 3), 96–101.

Shangsong, L., Zhaochun, R., & Maarten, dR. (2014). The impact of semantic document expansion on cluster-based fusion for microblog search. In *Proceedings of the 36th European Conference on IR Research, ECIR 2014* (pp. 493–499). springer.

Sinha, R., & Mihalcea, R. (2007). Unsupervised graph-basedword sense disambiguation using measures of word semantic similarity. In *ICSC '07 Proceedings of the International Conference on Semantic Computing* (pp. 363–369).

Smucker, M. D., Allan, J., & Carterette, B. (2007). A comparison of statistical significance tests for information retrieval evaluation. In *CIKM '07 Proceedings of the sixteenth ACM conference on Conference on information and knowledge management* (pp. 623–632). ACM.

Somnath, B., Krishnan, R., & Ajay, G. (2007). Clustering short texts using Wikipedia. In *SIGIR '07 Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 787–788).

Taiki, M., Kazuhiro, S., & Kuniaki, U. (2014). Time-aware latent concept expansion for microblog search. In *International AAAI Conference on Weblogs and Social Media(ICWSM 2014)*.

Tang, J., Wang, X., Gao, H., Hu, X., & Liu, H. (2012). Enriching short text representation in microblog for clustering. *Frontiers of Computer Science in China, 6*(1), 88–101.

Tao, K., Abel, F., Hauff, C., & Houben, G.-J. (2012a). What makes a tweet relevant for a topic? In *MSM 2012, Proceedings of the workshop on Making Sense of Microposts (MSM2012), workshop at the 21st World Wide Web Conference 2012* (pp. 49–56).

Tao, K., Abel, F., Hauff, C., & Houben, G.-J. (2012b). Twinder: a search engine for Twitter streams. In *Proceedings of the 12th International Conference on Web Engineering, ICWE 2012* (pp. 153–168). Spinger.

Xia, H., Nan, S., Chao, Z., & Tat-Seng, C. (2009). Exploiting internal and external semantics for the clustering of short texts using world knowledge. In *CIKM '09 Proceedings of the 18th ACM conference on Information and knowledge management* (pp. 919–928).

Xiang, W., Ruhua, C., Yan, J., & Bin, Z. (2013). Short text classification using wikipedia concept based document representation. In *International Conference on Information Technology and Applications (ITA)* (pp. 471–474). IEEE.

Xiangmin, Z., & Lei, C. (2013). Event detection over twitter social media streams. *The VLDB Journal, Springer, 23*(3), 381–400.

Xiaohua, L., Shaodian, Z., Furu, W., & Ming, Z. (2011). Recognizing named entities in tweets. In *HLT '11 Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, ACM, 1* (pp. 359–367).