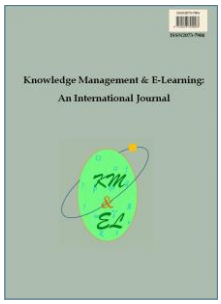
Semantic based entity retrieval and disambiguation system for Twitter streams

Narayanasamy Senthil Kumar
Muruganantham Dinakaran
Vellore Institute of Technology, Vellore, India



Knowledge Management & E-Learning: An International Journal (KM&EL)
ISSN 2073-7904

Recommended citation:
Kumar, N. S., & Dinakaran, M. (2019). Semantic based entity retrieval and disambiguation system for Twitter streams. *Knowledge Management & E-Learning, 11*(2), 262–280. https://doi.org/10.34105/j.kmel.2019.11.014

Knowledge Management & E-Learning, 11(2), 262–280

Semantic based entity retrieval and disambiguation system for Twitter streams

Narayanasamy Senthil Kumar*

School of Information Technology & Engineering
Vellore Institute of Technology, Vellore, India
E-mail: senthilkumar.n@vit.ac.in

Muruganantham Dinakaran

School of Information Technology & Engineering
Vellore Institute of Technology, Vellore, India
E-mail: dinakaran.m@vit.ac.in

*Corresponding author

Abstract: Social media networks have evolved as a large repository of short documents and gives the greater challenges to effectively retrieve the content out of it. Many factors were involved in this process such as restricted length of a content, informal use of language (i.e., slangs, abbreviations, styles, etc.) and low contextualization of the user generated content. To meet out the above stated problems, latest studies on context-based information searching have been developed and built on adding semantics to the user generated content into the existing knowledge base. And also, earlier, bag-of-concepts has been used to link the potential noun phrases into existing knowledge sources. Thus, in this paper, we have effectively utilized the relationships among the concepts and equivalence prevailing in the related concepts of the selected named entities by deriving the potential meaning of entities and find the semantic similarity between the named entities with three other potential sources of references (DBpedia, Anchor Texts and Twitter Trends).

Keywords: Named entity; DBPedia spotlight; Vector space model; Semantic similarity; Term frequency; Inverse document frequency

Biographical notes: Prof. N. Senthil Kumar received his Master Degree in M.Tech – IT from VIT University, Vellore and currently working as Assistant Professor in VIT University, Vellore, India. He has pocketed 13 years of teaching experience and his research areas includes Semantic Web, Information Retrieval and Web Services. He is currently holding a project on semantic understanding of named entities in the web and building a project on it.

Dr. M. Dinakaran received his Doctorate in Computer Science from Anna University, Chennai and Master Degree in M.Tech IT from VIT University, Vellore. He is currently working as Associate Professor in VIT University, Vellore, India. He has good teaching experience of more than 10 years. His area of research includes Information Retrieval, Networking and Web Service Management.

1. Introduction

Searching on the micro blogging system has been heavily suffered with data sparseness and data redundancy. Owing to the restricted length of the blog posts, there has been high contextualization and absence of apparent query terms in the post. And this has turned the blogging retrieval system inefficient and failed to return the desired search results to the users. Most of the recent blogging retrieval system follows the conventional term-based search like Term Frequency – Inverse Document Frequency (TF-IDF), probabilistic models, Bag of Words (BOW) model, etc. The term-based models are effective only to the document-based retrieval system and web page search system. In that cases too, it has suffered with polysemy of the word mapping and struggled with term variations in many of the context. To overcome the above problems, it is deemed to model the semantic based retrieval system which removes the ambiguity persists over the text (i.e. unstructured text) and links the entities in the text to the appropriate real-world entity sets. Thus, it has brought into the focus of entity-based retrieval system over the micro blogging search operations and disambiguates the entities with the populated knowledge base ontologies (such as DBpedia, Freebase, YAGO, etc). The major problem in the existing informational retrieval task is that it has not identified any semantics of the text instead it has followed the term weighting and term frequency of the whole document (Kalloubi, Nfaoui, & El Beqqali, 2016). Besides, it is resembled to the bag of words model wherein it was searched based on the keywords but not on the meaning of the words or on the context. Hence, the effective way to bridge the solution is to integrate the semantic knowledge base into the information retrieval systems and address the semantic gap already overlaying on the search operations.

In this paper, we have taken Twitter as a social media site and identified the potential problems which have been obstructing the micro blogging search operations as stated above. Here, we proposed a model that extracts the entities from tweets and disambiguate the entities based on three ways semantic filtering method. Each and every tweet is normalized, preprocessed and applied for shallow parsing to detect the key phrases, that are many times be a named entity in the tweets. The major task of shallow parsing is twofold. First, for each tweet, it is scanning for the noun phrases (also called as named entities) and if a surface form (i.e., a real-world entity presents in DBpedia Knowledge Base) is found for collected noun phrases, it will be stored separately. Otherwise, it will use NP Chunker to split the noun phrases and search the divided noun phrases separately on the DBpedia Spotlight. To extract the surface form (also called as mentions) from DBpedia, we have used the semantic ontology properties such as rdfs:label, foaf:name, dbpprop:officialName, dbpprop:name, foaf:givenName, and dbpprop:alias. These semantic properties will return the surface form for the extracted noun phrases and match it accordingly.

Second, when the named entities or noun phrases consists of more than one word, in such cases, we have used dependency parser to concatenate the words into single entity. For example, “Alli” and “Baba” can be concatenated into single entity as “Allibaba”. To identify related entities, already Ritter, Clark, Mausa, and Etzioni (2011) had proposed a Machine Learning (ML) algorithm for filtering named entity detection in tweets. Sometimes, the tweet has the longest continuous sequence of tokens such as ‘A Clash of Kings’, ‘A Storm of Swords’, ‘A Feast for Crows’ and the dependency parser to concatenate the sequence of tokens and labeled it as named entity (Alqahtani, 2017). Once the potential named entities have been identified from the collected tweets that were loaded for processing, then we define the method to add semantics to the tweet with appropriate surface forms (mapping it with DBpedia mentions) that depicts the contexts in which the tweet has actually been represented. For every named entity in the tweet, we

need to link it into the DBpedia knowledge source which has the global Unique Resource Identifier (URI) coded with Resource Description Framework (RDF) of the possible real-world entities.

In section 2, we have given a brief discussion on research works carried out by many authors and their domain restrictions in satisfying the expected outcomes. We have also indicated the major shortcomings and sheer limitations underlined in their research works and that has provided us the basis to reinstate this research work which has turned very advantageous at different levels. In section 3, we have proposed a system which takes the twitter streams as input and detect the potential named entities from the twitter streams. While doing so, it has encountered with many disambiguates which are persisting in large numbers and yields the contradictory results. To shun those entity disambiguates, we proposed the three ways strategic approaches such as DBpedia based Semantic Measure, Anchor Text based Cosine Similarity and Twitter Popularity Trend Detection to effectively filter out the disambiguated entities and mapped exactly to the given tweet(s) context.

Finally, in section 4, we have classified the named entities into its respective category or domain and find the coherent metrics using the machine learning algorithms to effectively categories the extracted named entities. We have used the Twitter Dataset on "Digital India Campaign" and compared the topic coherence metrics present over the collected dataset. To construct this dataset, we have tracked the eminent journalists, technologist, Data Analyst and potential users of Twitter to accumulate the post related to the event. We have preferred this topic for empirical analysis since it has attained huge reach and collected high volume of responses for the topic.

2. Related works

In most of the previous researches (Alahmari, Thom, & Magee, 2014; Hakimov, Oto, & Dogdu, 2012; Kataria et al., 2011), it was proposed with different perspectives of searching the entities and concerned mostly on entity description of the selected documents. Although it has provided the users with necessary information and facts about the chosen entity but failed to enhance the searching capabilities in three categories such as alternate entity query suggestion, prioritizing the entity attributes and selecting the appropriate entity type. Hakimov et al. (2012, May) was dealt only about the entity selection and categorizing the entities into their respective domain but not ranked the entities and thus failed to choose the right entity type for augmenting the search operations. Similarly, the authors (Hwang & Shadiev, 2014) have studied the cognitive model of student's ability and extract the potential entities based on the six different levels of cognitive processes.

As discussed in Li et al. (2013) about the query refinement and suggesting alternative query for improving the web search results, entity query has also to be refined and find the right combination of terms to find the exact match of the entity into its knowledge base such as DBpedia or FreeBase. Jung (2012) and Habib, Van Keulen, and Zhu (2013) have developed a wide range of query refinement techniques to generate the possible candidate queries and increase the precision of the query results. Unlike the query refinement method followed in the field of Information Retrieval, we have here dealing about the semantic data retrieval and its needs for the exact fit of ontology to disambiguate the entities. Hence, we looked for a reviewed approach to the entity suggestions and entity disambiguation towards domain specific ontologies. Besides, an ontology-based model for competence development was implemented by the researchers

(Malzahn, Ziebarth, & Hoppe, 2013) which has considered the professionals pervasively using the social networks and the mutual interconnection exists between colleagues and the professionals of different companies. The natural relationship among the professionals on the modern social networks has been implicitly analysed and categorized using the ontological framework.

The next problem discussed in the papers (Moro, Raganato, & Navigli, 2014; Vicient & Moreno, 2015) were about ranking the entity based on its associated attributes. When we dealt with integrated search, it has used entity-based queries to retrieve large number of attributes (i.e. as seen Sig.ma) pertaining to the entity and made the search operation based on its entity attributes. As the number of attributes increased for an entity, then the time taken to process and organize the entity attributes would gradually be high and thus reduce the scalability in ranking the entity attributes. Therefore, it has been suggested that the minimal structure of attributes would potentially increase the robustness of the entity search operation. Hence, BM25F model (Usbeck et al., 2014; Eger, 2018) has been used in the paper for ranking the fields and weighting the schema similar to Term Frequency (TF) – Inverse document Frequency (IDF). On contrary, the researchers (Murale & Raju, 2014) have developed a method to extract the entities and ranked them efficiently for the pharmaceutical company. The entity extraction has been carried out with the help of knowledge maps and social networks. For data pre-processing, we have emulated the model developed by the researchers (Zhang & Gao, 2014) and profusely followed it to tackle the contradiction on the informal text processing.

The authors in (Kataria et al., 2011; Carlson et al., 2016) had introduced the novel disambiguation method that required no external knowledge base except the entity name. They had proposed a Graph based model to assign the unique code to each entity and held the uniformity code among the entities. But it has failed to serve the purpose as the number of entities increased dynamically and also given the low precision score when compared the entities with context similarity, co-mentioned entities and co-referenced entities. But the Graph based approaches were proved useful for word sense disambiguation. The authors in (Derczynski et al., 2015) had compared different similarity measures and algorithms to detect and disambiguate the entities present in the text and they also found that the best measure to detect the entities are PageRank and Graph node degree. But when it is compared the same method for unstructured text, the results turned wrong and accuracy rate was very low. Eventually, we have considered the research work done by Aguiar and Correia (2017) for concept mapping and to reduce the informality occupied on the informal text.

Our major contribution of this paper is that we have proposed a system that addresses the problems which were stated above and enhance the capabilities of entity searching by incrementing the explicit connection mutually exists between extracted entity from tweets and DBpedia filtered mentions for that entity. With that base, we have identified the entity type for the right categorization of entity domains and respond by suggesting the appropriate entity types and entity sub types. In that way, we removed the impending problem persist in entity ambiguity and shuns the noisy attributes present over the entities. The following section would talk about how to disambiguate the entities and how to find the right entity selection over knowledge base such as DBpedia, YAGO or Freebase.

3. Proposed semantic retrieval context

Most of the times, tweets are about single topic and deals with single related events. But the problem evolves when the extracted named entities from the tweets trying to link into the existing knowledge base like DBpedia. For some named entities, there would have been more than one potential mention present in the DBpedia knowledge base (termed as Polysemy) and made it difficult to choose the correct real-world mention from the DBpedia Spotlight (Kumar & Muruganantham, 2016). Let's take for the instance that postal code and zip code are same and used to point the area of the region but the custom of using it in some countries is different with other countries and DBpedia has two referents in its knowledge base. And also, in some cases, two referent mentions are completely different like 'Jaguar' is a wild animal and it can also be a 'Jeep' to travel. Thus, identifying the most relevant and appropriate real-world entity in the DBpedia Spotlight is the challenging task and measuring the semantic relatedness between the extracted named entity and the mentions in the DBpedia knowledge base is the crucial part of the research (Shen, Wang, & Han, 2015). In order to bring out the semantic proximity between the set of ambiguous mentions from DBpedia and its candidate entity, we have measured the semantic similarity by considering the weight and the path exist between the connected nodes (i.e., the semantic connection between two or more mentions can be defined with the attached nodes and the semantic relatedness can be assured by the taxonomy "is-a" relation). This whole process is termed as entity linking. As discussed by the authors (Shen et al., 2015; Usbeck et al., 2014; Baldwin et al., 2015), entity linking for micro blog post is the complex activity as it suffers with lack of context to disambiguate the named entities. In order to effectively disambiguate the named entities in tweets, we have modeled three ways algorithmic measures which will remove any ambiguities persists over the selected named entities (see Fig. 1). They are:

i) Correlate the similarity between the selected named entity in tweet and corresponding DBpedia entity link.

ii) Find the similarity between the named entity in tweet and anchor text running over many web pages.

iii) Find the trends in twitter page that has been happened during the event.

Using the above stated approaches, we would assign the appropriate referent entity in the DBpedia knowledge base.

3.1. DBpedia based entity disambiguation

As stated above, for some of the named entities, there would have been more than one real world entities present in the DBpedia knowledge base. The seminal task here is to find out the exact match of referent mention to be linked to the named entity selected from the tweet. The property owl:sameAs is used to check whether two URIs in DBpedia have linked to the same entity in the real world as given by the authors (Hakimov et al., 2012). Although, owl:sameAs is considered as a widely applied property to connect two distinct objects, but the practical application is somewhat different from what was described.

Let's take the following example:

Example: "Boston is the newest tourist place in Turkey"

For the above text, the potential named entities are Boston and Turkey, but both have been referring to more than one real world entities (i.e. Boston is refereeing also to

Boston City, Boston University, Boston Magazine, Boston Foundation and many more in DBpedia Spotlight and Turkey is also point to country, bird, Restaurant etc). Therefore, the task is to set high emphasis on entity terms and find the appropriate surface form in DBpedia Spotlight (Buhmann et al., 2014). Using owl:sameAs alone is not sufficient to map to the exact fit of referent real world entity in DBpedia Ontology. Hence, we have used the DBpedia properties (given in Table 1) which absolutely connect the target entities (such as places, person, organization etc) into related mentions in the DBpedia Ontology. The Table 1 has listed the DBpedia properties of any related entities.

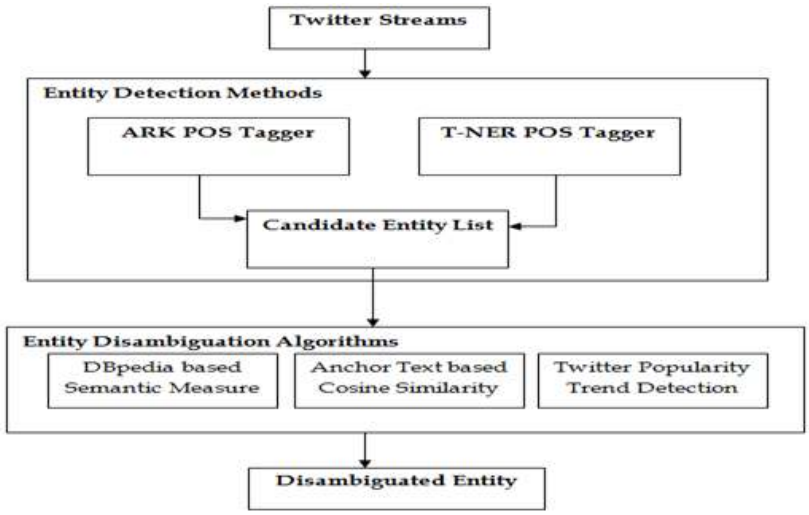


Fig. 1. Proposed architecture for entity disambiguation and linking

Table 1
DBpedia properties for the selected entities

Property	**Associated Entities**
dbpedia-owl:country	Country
dbpedia-owl:isPartOF	State and Country
dbpedia-owl:state	State
is dbpedia-owl: countrySeat of	Country
dbpprop:subdivisionName	Country and State
is dbpedia-owl:location of	Landmarks, buildings, locations, parks, companies etc.,
is dbpedia-owl:city of	Organization, University, Schools in the city
is dbpedia-owl:nearestCity of	Display the nearest city to the places.
Is dbpedia-owl:hometown of	People whose native places
is dbpedia-owl:deathPlace of	People who have died in the place
is dbpedia-owl:wikiPage Redirect of	Alias of the Place
Dbpprop:nickname	Nickname of the place

Hence, the relevant mentions in the DBpedia Spotlight can be extracted through entity labels, disambiguated pages and redirected pages.

3.1.1. Entity labeling

The real world entities in DBpedia Spotlight (Alahmari et al., 2014; Hulpuş, Prangnawarat, & Hayes, 2015) can be obtained through the data properties of the DBpedia ontologies such as rdfs:label, foaf:name, dbpprop:officialName, dbpprop:name, foaf:givenName, dbpprop:birthName, dbpprop:alias. But to extract the candid surface form of the entities, we have used rdfs:label which is giving the exact surface form of the entity. The SPARQL query for extracting the surface form of the entity is given below:

```
SELECT ?s WHERE {
?s rdfs:label “+searchText+”@en.”
?s foaf:name “+searchText+”@en.”
?s foaf:givenName
“+searchText+”@en.”
}
```

Before we disambiguate the entities, we need to fix the predefined labels to the entities and get the concepts linked to it. Using DBpedia Spotlight, we have executed the SPARQL query to get the Table 2 and obtained the concept and DBpedia Label associated with every entity fetched by the query. Given the term “ACC”, we have fetched the top 10 entity labels associated in DBpedia Spotlight and its relevant concepts. Entity Labeling will facilitate the entity annotation and made the entity disambiguation easier after this process. In the conventional information retrieval (IR), manual annotation has been carried out to increase the efficiency of the task and attained the desired accuracy rate (Liu, Zhang, Wei, & Zhou, 2011; Lu, Roa, & Fang, 2014). But here, we have made the automatic annotation of entities of the unstructured text and attained the progressive accuracy rate when compared to other existing systems.

Table 2
Preferred DBpedia entity labels for the entity

Term	Concept	Label
"ACC Asian XI One Day International cricketers"	http://dbpedia.org/resource/Category:ACC_Asian_XI_One_Day_International_cricketers	"ACC Asian XI One Day International cricketers"
"ACC Athlete of the Year"	http://dbpedia.org/resource/Category:ACC_Athlete_of_the_Year	"ACC Athlete of the Year"
"ACC Championship Game"	http://dbpedia.org/resource/Category:ACC_Championship_Game	"ACC Championship Game"
"ACC Men's Basketball Tournament”	http://dbpedia.org/resource/Category:ACC_Men's_Basketball_Tournament	"ACC Men's Basketball Tournament"
"ACC Men's Soccer Tournament"	http://dbpedia.org/resource/Category:ACC_Men's_Soccer_Tournament	"ACC Men's Soccer Tournament"
"ACC Trophy"	http://dbpedia.org/resource/Category:ACC_Trophy	"ACC Trophy"
"ACC Twenty20 Cup"	http://dbpedia.org/resource/Category:ACC_Twenty20_Cup	"ACC Twenty20 Cup"
"ACC Women's Basketball Tournament"	http://dbpedia.org/resource/Category:ACC_Women's_Basketball_Tournament	"ACC Women's Basketball Tournament"
"ACC Women's Soccer Tournament"	http://dbpedia.org/resource/Category:ACC_Women's_Soccer_Tournament	"ACC Women's Soccer Tournament"
"ACC articles by importance"	http://dbpedia.org/resource/Category:ACC_articles_by_importance	"ACC articles by importance"

3.1.2. Disambiguation pages

In order to identify the possible disambiguated surface forms present in the DBpedia knowledge base, we have used the data property dbont:wikiPageDisambiguates that can group entities with various meanings but refereeing to the single title (Houlsby & Ciaramita, 2013; Mulay & Kumar, 2011). That is, if all the entities are grouped for disambiguation, meaning that these are the candidate list for the surface form.

For example, "Obama" and "Barack Obama" can be clustered under "US President" entity since they were represented with common referenced entity "US President".

```
SELECT distinct ?s WHERE {
?disamb dbont:wikiPageDisambiguates
?s.
?disamb rdfs:label "+searchText+".
}
```

Once the candidate list for the surface form is ready, our system is going to find the context surfaced around the information which helps to disambiguate the entities. We have used Vector Space Model (VSM) to build the multi-dimensional space of entities present in the DBpedia Ontology. As we followed the Vector Space Model (VSM) for entities disambiguation which was also described elaborately by the researchers (Buhmann et al., 2014; Moro et al., 2014; Sareminia, Shamizanjani, Mousakhani, & Manian, 2016), the TF-IDF (Term Frequency – Inverse Document Frequency) is failed to obtain the local relevance of a mention in the DBpedia candidate list. If we apply TF-IDF for disambiguation of candidate entities, then TF will find the relevance of mentions in the given candidate list and the IDF will get the related matches of mentions in the collection of DBpedia recourses. Although Term Frequency (TF) has given the global significance of the entities (Candidate List of Surface forms), but it fails to get the exact match of an entity among the ambiguous candidate list of entities. Let's take an instance to substantiate this problem in detail. Suppose the mention, 'Apple' occurs in 7 concepts in the overall collection of DBpedia resources, then its Inverse Document Frequency (IDF) will be usually high because of the simpler reason that the Term Frequency (TF) of the mention is very low when to compared to the IDF (i.e., let's assume that DBpedia has 1.5 million resources listed and for the sake of our above illustration, we have taken the mention 'Apple' which has occurred in 10 concepts of resources in the entire DBpedia resource list. Then the TF-IDF calculation would be, 7/1,500,000).

Hence, in order to map the correct entity into the DBpedia resource URI, we have taken the alternative approach is called Inverse Candidate Frequency (ICF). The basic logic behind this approach is that it will take the entity from the tweet and find the list of real-world entities relating to the given entity in DBpedia resources which we already called as Candidate list. As done in IDF (Li et al., 2013; Shen, Wang, Luo, & Wang, 2013), here we have contradicting with the approach of comparison i.e., instead of computing the inversely proportional to the mention in the entire DBpedia resources, we have compared the inverse proportional only to the number of DBpedia resources which have been selected as candidate list. Again, let's take an above illustration for the sake of clarity in ICF. The mention 'Apple' has occurred in 7 related DBpedia resources. Hence, similarity measure has been taken among the seven related DBpedia resources. According to the authors (Liang, Ren, & De Rijke, 2014; Wamba et al., 2016), the above

scenario can be well represented in mathematical terms, that is, R_s is the collection of potential resources for an available surface form. That is, let $n(W_j)$ be the total number of candidate resources in R_s that are implicitly allied with the word W_j. Then we define:

$$ICF(W_i) = \log \frac{|R_s|}{n(W_j)} \quad (1)$$

$$ICF(W_i) = \log |\ R_s\ | - \log n(W_j) \quad (2)$$

Algorithm for Entity Disambiguation

Input: Given the list of ambiguous entities to find the exact referents in KB

Output: Rank the candidate entities and return the entity with high score

foreach ambiguous entities e_i in E, do

Find the set of candidate referents r=(r1, r2, .. rn) Ɛ E

foreach referents r Ɛ E do

Extract the list of hypernym and stored in Stack S

Find the total number of resources linked to the extracted candidate sets ei Ɛ E

End loop

Perform the TF-ICF to rank the entity obtain the highest relevance score.

End loop

Return the entity with high score and label it as the exact match to the context.

3.1.3. Redirect pages

In some of the cases, there would be no surface form of the given entity and the page will be just redirected to the base content of the site. The alternative topic of an entity will be shown, and redirected page surface form will be chosen for the candidate list for the given entity. The property dbont:wikiPageRedirects yields the references page of content and labels its surface form for further process of extraction. The DBpedia doesn't hold any content on itself and gives only the redirection.

```
SELECT distinct ?s WHERE {
?redirect dbont:wikiPageRedirects
?s.
?redirect rdfs:label
"+searchText+"@en.
}
```

3.2. Anchor text-based similarity measure

When the extracted entities from tweets have no referent mention in DBpedia Spotlight and returned NIL as the result, we need to map to the correct concepts pertaining to the entity over the web pages or any other sources such as blogs, dashboards, sites, etc. and link it to the appropriate real-world entity source. Hence, we have taken the "Anchor Test Measure" to find the named entities which are present in the tweet but absent in DBpedia or Freebase ontology. The authors (Bansal et al., 2014) has stated that the Anchor text is a string that represents the concepts present in multiple web pages. A string which points to the concepts are termed as hypertext of anchors and we need to find the exact match of the anchor text for the given entity. In this connection, we have used the tool called Google Cross-Wiki Dictionary (GCD) to find the relevant named entities occupied in various web pages and list out the context in which it has been formed. In order to fetch the exact match of the anchor text, we have loaded the named entities along with the context given in the tweet to the GCD application and fetch the related entities from the various web pages. Based on the similarity score, we ranked the list of probable entities from the anchor text and applied the heuristic similarity score to rank the entities based on the given tweet context. For this task, we have used the Cosine Similarity between the anchor text and named entities extracted from tweets. As a result, we have taken the anchor text entity which is giving the higher similarity score to the given tweet context.

$$\text{Anchor\_Similarity(tweet, entity)} = \frac{|T_w \cap E_w|}{|T_w \cup E_w|} \quad (3)$$

Some of the benefits of using the anchor text for categorizing the named entities are given below:

a) Searching over the anchor text in the collection of web pages is much lesser than the searching the entire web pages by using web crawlers. By means of this, processing the anchor text is much faster than crawling the entire web pages.

b) Pages containing the referring anchor text have higher inbound links and this facilitates higher ranks for the chosen link analysis.

c) Anchor text facilitates to shun the ambiguities persist over the text by improving the relevant contexts from tweets and enhance the searching capabilities over the anchor texts. Therefore, it provides refinement over the search results and yields the better results.

3.3. Twitter popularity-based trends measure

In the above section, we have used anchor text to leverage the appropriate targeted entities in the collected tweets. But in this section, we are going to measure the popularity of the event for the collected tweets over the period of time and disambiguated the tweets in an appropriate manner. Therefore, we have calculated the ranking of every entity referred in the tweets and order them according to its ranking. The purpose of measuring the entity in the tweets is twofold: one is to find the popularity of an entity among the extracted entities and second is used to disambiguate the entities based on its popularity score. In order to find the popularity of the entity in the tweets, we first find the bursty terms in the tweets (i.e., the terms which occurs very frequently and unusually been represented a greater number of times by many users or just retweeted the mention often to indicate the event). As we are ranking the entities in the tweets for the specified event, we then cluster those bursty terms that appearing more frequently either individually or co-referenced with other entities and identify the trends from the collected bursty terms

as seen in Fig. 2. The process does not stop here to declare the trend of the event, instead it is looking for additional facts and information about the trend and augmented with interesting facts of the events to substantiate with better results.

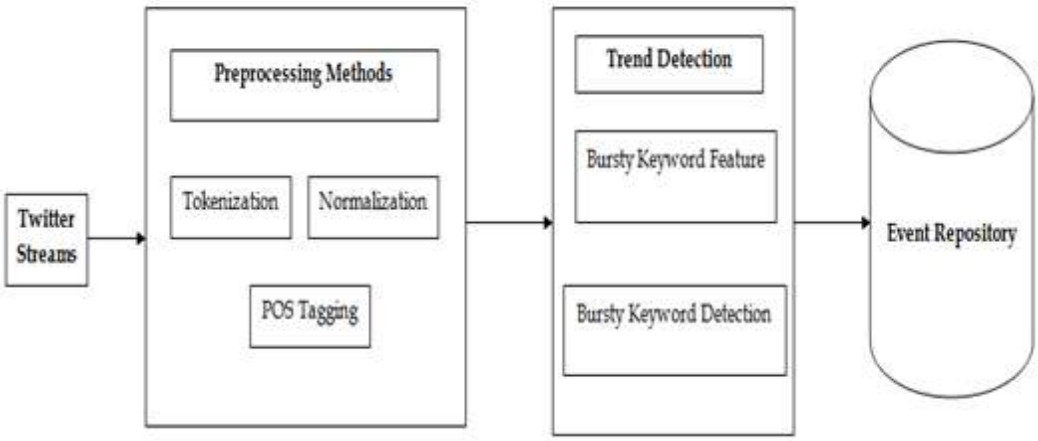


Fig. 2. Measuring the trend of the Tweets

The major problem with above approach to detect the trends over the period of time in twitter is a very complex analysis and there would be more potential entities occurred very frequently in the collected tweets. In order to determine the exact trend of the event over the time, there was an algorithm already developed called QueueBurst (Shen et al., 2015). The algorithm worked in the following manner:

- First remove the irrelevant words out of the collected tweets.
- Filters stop words and symbols from the tweets.
- Identify the bursty terms by optimizing the tweets when it arrives in the stream of tweets.
- In most cases, many terms have just appeared in many tweets over a short span of time and those things cannot be treated as bursty terms.
- To group bursty terms, assess the co-occurrences of terms in the tweets and history of the tweets is retrieved for each bursty term.
- We have computed the Term Frequency (TF) of each word and produced the list of words for each topic and ranked them all in descending order by TF value.
- It has been apparently seen that there is a strong connection between the types of trending topics.

In order to analyze the text in the tweets strenuously, we have found the following reasons are substantial to identify the trends in the tweets.

i) Several words point to the present situations like today, tonight, showing live, watching now, etc. These words are referring towards the trending topics and they are strongly connected to the happening of the events which is absolutely reflected in the user's tweet.

ii) Some words stand with Happy, Excellent, Awesome, etc. to denote the celebration or to congratulate any one and it has been promulgated very deeply

in the user's tweets. These words can also be taken as an indicator for identifying the trending topic of the times.

iii) In some cases, the terminologies or annotation has not been used but the potential named entities have been circulated among the twitter users widely and thus it has become the trending topic of the day.

iv) In some rare cases, there would be some memes, that went viral over the tweets and it has been retweeted many times by more than one user and become the trend of the day.

As a whole, the proposed entity linking system is consists of two crucial stages. One is generating the possible candidate entities from the knowledge base and second is to disambiguate the entity to its root origin. To ensure the correct generation of candidate entities and disambiguate the entities, we here proposed the model that takes the name variants and its context together to categorically define the exact matching between the entity and entity context. By means of the above model, the disambiguation is attained with low recall and high precision.

We have empirically tested and evaluated our proposed approach with the existing knowledge base entity linking datasets and the empirical results has shown that the proposed entity candidate generation has drastically increased the recall and precision of the results. We have also witnessed that co-reference-based entity matching, and context matching has resolved the long pending problem of pseudonymity and polysemy issues in entity detection and linking.

4. Classification of named entities

Once the potential named entities have been identified from the Twitter datasets using any of the above three methods described, the next crucial task would be to assign the extracted named entities into the predefined types of its classes such as person, product, geographical locations, time, company etc., Though many Information Retrieval (IR) techniques had been proposed for document processing in information retrieval (Ifrim, Shi, & Brigadir, 2014; Liang et al., 2014) , it has failed to categorize the entities into its associated domains or classes and when it is particularly extracted from unstructured text such as Twitter Streams. Therefore, we have taken this supervised machine learning approach as a classification mechanism and obtain the DBpedia knowledge base for further disambiguation and fixation of entities into its predefined types appropriately.

Table 3
Accuracy of entity matching with DBpedia knowledge base

Classifier	Entities Detected [Total: 5500]	Correct DBpedia Match	Precision	Recall
SVM	5345	5135	0.899	0.813
Maximum Entropy	5254	5200	0.910	0.864
Hidder Markov Model	5310	5215	0.918	0.871
Proposed Model	5410	5350	0.967	0.892

In Table 3, we have used the different machine learning algorithms to estimate the precision and recall of the collected twitter streams on the topic "Digital India Campaign" and mapped to the exact fit for the real-world mention present in DBpedia Spotlight.

Besides, we have compared the different machine learning algorithms against the proposed model and witnessed that our proposed model outperforms the other supervised machine learning algorithms and yield the accurate results in terms of precision and recall.

In the earlier research on entity linking and entity classification, the authors (Kataria et al., 2011) were faced the problem of several named entities cannot have the references in the chosen knowledge base (i.e., either DBpedia or Freebase) and because of this non-existent entity in the knowledge base, the system would yield the NIL references. But in this proposed model, we have approached with three categories of entity disambiguation (i.e. DBpedia reference, Anchor text reference or Trend detection) and found the appropriate corresponding entity mention in the knowledge base and further used that a sequence to classify the named entities into its appropriate domain class. Besides, we have used other types of feature called word embedding. The word embedding is used for every potential named entity to measure an average vector of words in the n-gram entity mentions. The number of characters and words in the entities are used to enhance the quality of the model and further used to increment the expressiveness of the entities so as to classify the entity into its appropriate domain of classes.

The entity classification can also be done by the following features:

- Entity Types: Select and attribute the entity types based on the correspondence of DBpedia.
- Entity Detection: Entity detection can be done using the NER Model.
- N-gram Vector: Vector representation of n-gram.
- Entity linking relevance score: Semantic similarity score measured between entity and entity mention in the knowledge base.
- Character Length: Number of characters to process in the n-gram entity.
- Token Length: Total number of tokens identified in the n-gram.

Besides, the entity classification can semantically be linked with the following properties of the entities:

Table 4
Semantic models of entities from unstructured texts

Entity Properties	**Semantic Properties**
Topic Identification	<Document> dc:subject <Topic>
Entity Recognition	owl:NamedIndividual
Named Entity resolution (NE)	owl:sameAs
Named Entity coreference	owl:sameAs
Entity Types	owl:Class ‖ owl:ObjectProperty ‖ owl:DatatypeProperty
Entity Sense tag	owl:NamedIndividual rdf:type owl:Class
Sense disambiguation (classes)	owl:equivalentClass
Taxonomy (subclasses)	owl:subClassOf
Entity Binary relation	owl:ObjectProperty ‖ owl:DatatypeProperty
Event Identification	<Event> rdf:type <Event.type>
Frame Sets	<Event.type> owl:subClassOf <Frame>

Entity Restriction	owl:Restriction
Open Linked entities	owl:sameAs ‖ owl:differentFrom
Conjunct of individual entities	owl:NamedIndividual
Disjunction of individual entities	owl:NamedIndividual

In the Table 4, we have constructed the translation practices exist between the formal texts (NLP) and semantic modeling and also given the correspondence to every entity identified in the Tweets. By the means of the above translation process, we can very easily identify the appropriate counterparts of the entities in the exiting knowledge base such as class equivalence of the entities, class disjoint between the entity, entity restrictions, etc.

5. Empirical results

In our experiment, we have used the Twitter Dataset on "Digital India Campaign" and compared the topic coherence metrics present over the collected dataset. To construct this dataset, we have tracked the eminent journalists, technologist, data analyst and potential users of Twitter. We have preferred this topic for the empirical analysis since it has attained huge reach and collected high volume of responses for the topic. After the accumulation of the datasets, we have done the preprocessing steps and normalized it with the prescribed format for evaluation. When we compare the named entities in the datasets, we have identified that there is a high ambiguity ratio on the extracted entity sets and carried out the statistical analysis called prior probability which shows that the entire datasets hold the disambiguated ratio of 53.14%. We have then taken our proposed approach for evaluation and compared that the TF * ICF based performance is obtained the disambiguated percentage of results with 82.76% which is much better than using the TF * IDF disambiguated results 58.39%. This is the better indication that our proposed approach has handled the disambiguation of entities with proper balance and maintains the good store of results. This confirms that the use of TF * ICF is a positive indication to apply the single disambiguation methods to overcome the difficulties and assured that if the appropriate contextual evidence were given, it is providing the results with good precision as given in the Table 5. The entity extraction has been carried through the three ways algorithmic approach (DBpedia reference, Anchor text reference or Trend detection) to find the occurrence of every entity extracted from tweets to corresponding referent real world entity sources (i.e., DBpedia, Webpages, Blogs, Dashboards, and Trends). Extracted entities (as given in Table 3) were attributed to corresponding entity classes as given in the Table 5 and compared the precision, recall and F1 score with different machine learning algorithms such as Support Vector Machine, Maximum Entropy Model and Hidden Markov Model. When compared with existing machine learning algorithms, our proposed methods have shown the accuracy rate much better than other models described. In this comparison, we have taken the entity class and its associated entities for identifying the exact match of entity mention in the DBpedia Spotlight and filtered the candidate entities for disambiguation and ranking. Among all the three machine learning algorithms (Support Vector Machine, Maximum Entropy Model and Hidden Markov Model), our proposed model (i.e, following DBpedia reference, Anchor text reference and Trend detection) has shown the progressive results in finding the exact fit of entity referent in the DBpedia Spotlight. Using this proposed model, we have also the first to witness that the NIL referent entity problem has been solved and appropriate entity

source for the candidate entity has been obtained by any of the three approaches described above.

Table 5
Final result classifier for the proposed system

Entity Class	**SVM**			**Maximum Entropy**			**Hidden Markov Model**			**Proposed Model**		
	Prec	Recall	F1	Prec	Recall	F1	Prec	Recall	F1	Prec	Recall	F1
Country	0.831	0.662	0.811	0.723	0.565	0.783	0.811	0.617	0.788	0.934	0.882	0.904
Politics	0.867	0.652	0.788	0.813	0.673	0.815	0.819	0.710	0.791	0.899	0.723	0.901
Members	0.769	0.620	0.687	0.678	0.722	0.681	0.815	0.811	0.716	0.823	0.789	0.863
Campaign	0.892	0.721	0.765	0.768	0.734	0.788	0.788	0.688	0.729	0.942	0.825	0.897
Prime Minster	0.922	0.789	0.734	0.872	0.769	0.789	0.814	0.710	0.786	0.939	0.822	0.851
Clean India	0.726	0.546	0.629	0.672	0.661	0.781	0.873	0.711	0.799	0.876	0.789	0.832
Environment	0.930	0.756	0.754	0.786	0.723	0.801	0.784	0.784	0.727	0.933	0.876	0.811
Mission	0.801	0.647	0.766	0.637	0.698	0.818	0.788	0.638	0.767	0.897	0.883	0.819
Organization	0.834	0.687	0.699	0.788	0.711	0.716	0.791	0.741	0.712	0.878	0.856	0.799
Volunteers	0.876	0.598	0.756	0.822	0.619	0.798	0.801	0.716	0.784	0.901	0.834	0.817
Youth Group	0.725	0.648	0.657	0.671	0.714	0.698	0.810	0.817	0.762	0.822	0.845	0.781
Program	0.789	0.589	0.690	0.711	0.617	0.692	0.789	0.734	0.790	0.827	0.769	0.810

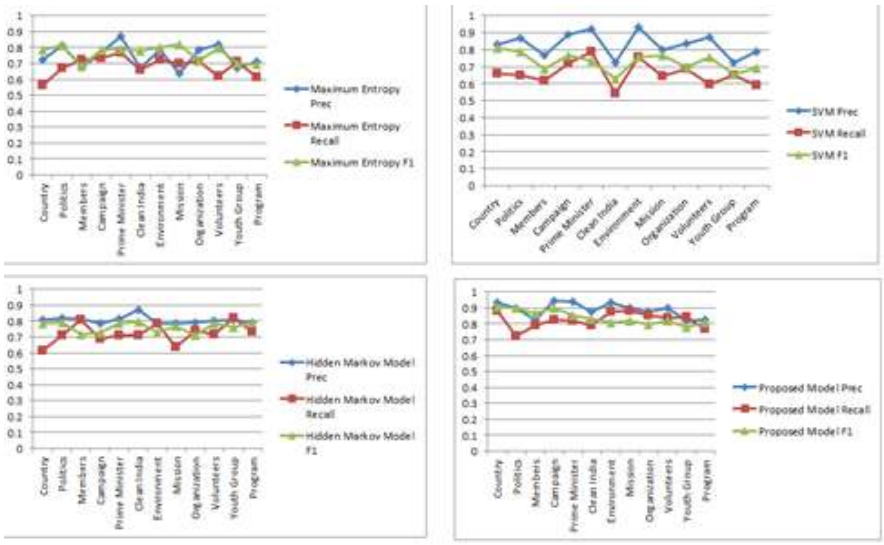


Fig. 3. Statistical representation of final result

The Table 5 is more evident that the precision and recall of the proposed system grows consistently (i.e. a rate of 1% accuracy is increased on average for the selected entity classes) and the ranking of the overall approach of the enablement as given in the Fig. 3. Besides, it has also been observed that the proposed system has tackled the disambiguation problem strenuously with three strategic approaches (i.e. DBpedia reference, Anchor text reference or Trend detection) and provide the disambiguated search environment for the potential users.

We have also compared the arrived results with the existing annotation services like Alchemy API, OpenCalais, Zemanta and Ontos Semantic API. The performance of the proposed system is working far better than the existing services and yields the results with good preciseness and better recall. All the existing annotation services have successfully linked to any of the Knowledge bases (DBpedia, YAGO or Freebase) only if the named sources in present in the knowledge base. But in our proposed work, we have given the three-way approach to tackle the problem of non-availability of named entities in the knowledge base.

6. Conclusion

The research work proposed here, deals with the problems of handling semantic aspects of collected tweets and has the ability to solve the ambiguous tendency of the extracted named entities from Twitter Streams. Unlike bag-of-words filtration of search, we have used concepts-based representation of entities with the support of DBpedia Knowledge Base. Besides, we have considered the entities which has not found in the DBpedia Ontology by proposing the new strategic approaches (i.e., using the anchor text and finding the trending of the entity). Thus, the proposed work has ensemble the mutual relationships among the concepts and related concepts and linked it with the DBpedia Ontology. Furthermore, our proposed model aggregated similarity metric is used to measure the semantic similarity score between the concepts and filter the entities which are overlapping with other unrelated concepts. The potential application to support the twitter analysis is aimed towards decision-oriented applications such as Customer Relationship Management (CRM), building new knowledge for pattern mining, tackling the natural crisis such as earthquake, flood, and many such disasters, assisting leading companies to know their product popularity, promotions and many more.

7. Further research directions

The sentimental analysis has faced huge problem to segregate the tweets based on emotions and many times, it failed to eliminate the ambiguities persist over the people (i.e., FOAF problem has not been solved effectively). Still, many anomalies striking the performance of the sentimental analysis at the large scale and by utilizing the proposed approach, we can further enhance the capabilities to curbing the anomalies and effectively perform the sentimental analysis over product ratings, election campaign survey results, and etc.

ORCID

Narayanasamy Senthil Kumar https://orcid.org/0000-0002-2951-5740

Muruganantham Dinakaran http://orcid.org/0000-0003-3326-868X

References

Aguiar, J. G., & Correia, P. R. M. (2017). From representing to modelling knowledge: Proposing a two-step training for excellence in concept mapping. *Knowledge Management & E-Learning, 9*(3), 366–379.

Alahmari, F., Thom, J. A., & Magee, L. (2014). A model for ranking entity attributes using DBpedia. *Aslib Journal of Information Management, 66*(5), 473–493.

Alqahtani, F. H. (2017). Users' perspectives on using Wikis for managing knowledge: Benefits and challenges. *Journal of Organizational and End User Computing (JOEUC), 29*(3), 1–23.

Baldwin, T., de Marneffe, M. C., Han, B., Kim, Y. B., Ritter, A., & Xu, W. (2015). Shared tasks of the 2015 workshop on noisy user-generated text: Twitter lexical normalization and named entity recognition. In *Proceedings of the ACL 2015 Workshop on Noisy User-generated Text* (pp. 126–135).

Bansal, R., Panem, S., Radhakrishnan, P., Gupta, M., & Varma, V. (2014, April). Linking entities in #Microposts. In *Proceedings of the 4th Workshop on Making Sense of Microposts* (pp. 71–72).

Buhmann, L., Fleischacher, D., Lehmann, J., Melo, A., & Volker, J. (2014). Inductive lexical learning of class expressions. *Lecture Notes in Computer Science, 8876*, 42–53.

Carlson, J. R., Zivnuska, S., Harris, R. B., Harris, K. J., & Carlson, D. S. (2016). Social media use in the workplace: A study of dual effects. *Journal of Organizational and End User Computing (JOEUC), 28*(1), 15–31.

Derczynski, L., Maynard, D., Rizzo, G., van Erp, M., Gorrell, G., Troncy, R., ... Bontcheva, K. (2015). Analysis of named entity recognition and linking for tweets. *Information Processing & Management, 51*(2), 32–49.

Eger, L. (2018). How people acquire knowledge from a web page: An eye tracking study. *Knowledge Management & E-Learning, 10*(3), 350–366.

Habib, M. B., Van Keulen, M., & Zhu, Z. (2013, May). Concept extraction challenge: University of Twente at #MSM2013. In *Proceedings of the 3rd Workshop on Making Sense of Microposts* (pp. 17–20).

Hakimov, S., Oto, S. A., & Dogdu, E. (2012, May). Named entity recognition and disambiguation using linked data and graph-based centrality scoring. In *Proceedings of the 4th International Workshop on Semantic Web Information Management* (Article No. 4). ACM.

Houlsby, N., & Ciaramita, M. (2013). Scalable probabilistic entity-topic modeling. *arXiv preprint arXiv:*1309.0337.

Hulpuş, I., Prangnawarat, N., & Hayes, C. (2015, October). Path-based semantic relatedness on linked data and its use to word and entity disambiguation. In *Proceedings of the International Semantic Web Conference* (pp. 442–457). Springer, Cham.

Hwang, W. Y., & Shadiev, R. (2014). Cognitive diffusion model with user-oriented context-to-text recognition for learning to promote high level cognitive processes. *Knowledge Management & E-Learning, 6*(1), 30–48.

Ifrim, G., Shi, B., & Brigadir, I. (2014, April). Event detection in twitter using aggressive filtering and hierarchical tweet clustering. In *Proceedings of the Second Workshop on Social News on the Web (SNOW)*. ACM.

Jung, J. J. (2012). Online named entity recognition method for microtexts in social networking services: A case study of twitter. *Expert Systems with Applications, 39*(9), 8066–8070.

Kalloubi, F., Nfaoui, E. H., & El Beqqali, O. (2016). Microblog semantic context retrieval system based on linked open data and graph-based theory. *Expert Systems with Applications, 53*, 138–148.

Kataria, S. S., Kumar, K. S., Rastogi, R. R., Sen, P., & Sengamedu, S. H. (2011, August). Entity disambiguation with hierarchical topic models. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1037–1045). ACM.

Kumar, N. S., & Muruganantham, D. (2016). Disambiguating the Twitter stream entities and enhancing the search operation using DBpedia ontology: Named entity disambiguation for Twitter streams. *International Journal of Information Technology and Web Engineering (IJITWE), 11*(2), 51–62.

Li, Y., Wang, C., Han, F., Han, J., Roth, D., & Yan, X. (2013, August). Mining evidences for named entity disambiguation. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1070–1078). ACM.

Liang, S., Ren, Z., & De Rijke, M. (2014, April). The impact of semantic document expansion on cluster-based fusion for microblog search. In *Proceedings of the European Conference on Information Retrieval* (pp. 493–499).

Liu, X., Zhang, S., Wei, F., & Zhou, M. (2011, June). Recognizing named entities in tweets. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1* (pp. 359–367). Association for Computational Linguistics.

Lu, K., Roa, D., & Fang, H. (2014). *Concept based tie-breaking and maximal marginal relevance retrieval in microblog retrieval*. Retrieved from https://pdfs.semanticscholar.org/11a4/b50a5b436189a3b3637f436b1b6b80bee0cc.pdf

Malzahn, N., Ziebarth, S., & Hoppe, H. U. (2013). Semi-automatic creation and exploitation of competence ontologies for trend aware profiling, matching and planning. *Knowledge Management & E-Learning, 5*(1), 84–103.

Moro, A., Raganato, A., & Navigli, R. (2014). Entity linking meets word sense disambiguation: A unified approach. *Transactions of the Association for Computational Linguistics, 2*, 231–244.

Mulay, K., & Kumar, P. S. (2011, December). SPRING: Ranking the results of SPARQL queries on linked data. In *Proceedings of the 17th International Conference on Management of Data* (Article No. 12). Computer Society of India.

Murale, V., & Raju, G. P. (2014). Analyzing the role of social networks in mapping knowledge flows: A case of a pharmaceutical company in India. *Knowledge Management & E-Learning, 6*(1), 49–65.

Ritter, A., Clark, S., Mausa, & Etzioni, O. (2011, July). Named entity recognition in tweets: An experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 1524–1534). Association for Computational Linguistics.

Sareminia, S., Shamizanjani, M., Mousakhani, M., & Manian, A. (2016). Project knowledge management: An ontological view. *Knowledge Management & E-Learning, 8*(2), 292–316.

Shen, W., Wang, J., & Han, J. (2015). Entity linking with a knowledge base: Issues, techniques, and solutions. *IEEE Transactions on Knowledge and Data Engineering, 27*(2), 443–460.

Shen, W., Wang, J., Luo, P., & Wang, M. (2013, August). Linking named entities in tweets with knowledge base via user interest modelling. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 68–76). ACM.

Usbeck, R., Ngomo, A. C. N., Röder, M., Gerber, D., Coelho, S. A., Auer, S., & Both, A. (2014, October). AGDISTIS-graph-based disambiguation of named entities using linked data. *Lecture Notes in Computer Science, 8796*, 457–471.

Vicient, C., & Moreno, A. (2015). Unsupervised topic discovery in micro-blogging networks. *Expert Systems with Applications, 42*(17/18), 6472–6485.

Wamba, S. F., Akter, S., Kang, H., Bhattacharya, M., & Upal, M. (2016). The primer of social media analytics. *Journal of Organizational and End User Computing (JOEUC), 28*(2), 1–12.

Zhang, K., & Gao, F. (2014). Social media for informal science learning in China: A case study. *Knowledge Management & E-Learning, 6*(3), 262–280.