Social Network Analysis and Mining (2018) 8:46
https://doi.org/10.1007/s13278-018-0523-0

ORIGINAL ARTICLE

# Entity linking of tweets based on dominant entity candidates

**Yue Feng[1] · Fattane Zarrinkalam[2] · Ebrahim Bagheri[2] · Hossein Fani[2] · Feras Al-Obeidat[3]**

Received: 3 March 2018 / Revised: 12 June 2018 / Accepted: 15 June 2018


**Abstract**
Entity linking, also known as semantic annotation, of textual content has received increasing attention. Recent works in this area have focused on entity linking on text with special characteristics such as search queries and tweets. The semantic annotation of tweets is specially proven to be challenging given the informal nature of the writing and the short length of the text. In this paper, we propose a method to perform entity linking on tweets built based on one primary hypothesis. We hypothesize that while there are formally many possible entity candidates for an ambiguous mention in a tweet, as listed on the disambiguation page of the corresponding entity on Wikipedia, there are only few entity candidates that are likely to be employed in the context of Twitter. Based on this hypothesis, we propose a method to identify such dominant entity candidates for each ambiguous mention and use them in the annotation process. Particularly, our proposed work integrates two phases (i) dominant entity candidate detection, which applies community detection methods for finding the dominant candidates of ambiguous mentions; and (ii) named entity disambiguation that links a tweet to entities in Wikipedia by only considering the identified dominant entity candidates. Our investigations show that: (1) there are only very few entity candidates for each ambiguous mention in a tweet that need to be considered when performing disambiguation. This helps us limit the candidate search space and hence noticeably reduce the entity linking time; (2) limiting the search space to only a subset of disambiguation options will not only improve entity linking execution time but will also lead to improved accuracy of the entity linking process when the main entity candidates of each mention are mined from a temporally aligned corpus. We show that our proposed method offers competitive results with the state-of-the-art methods in terms of precision and recall on widely used gold standard datasets while significantly reducing the time for processing each tweet.

**Keywords** Entity linking · Disambiguation · Linked data · Microblogging

## 1 Introduction

With the dramatically fast adoption of social media, Twitter has become one of the largest microblogging platforms. Every second, there are around 6000 tweets posted on Twitter on average, which corresponds to over 500 million tweets per day.[1] Hence, Twitter is often considered to be a source for significant information for many applications such as trend identification, user interest detection and customer service, among others (Jansen et al. 2009; Zarrinkalam et al. 2015; Pak and Paroubek 2010). The need to process tweets for such applications demands the development of new techniques that are specifically built for processing tweets and allow for the extraction of semantics and actionable insight.

The primary goal of work in the area of semantic annotation and entity linking (Jovanovic et al. 2014) is to process a textual document, identify the mentions that have the potential to be linked to some entity in knowledge bases such as DBpedia or Freebase and connect them to those entities. This allows for text analytics at a higher level, which focuses on the meaning of the documents in addition to its syntactics (Liu et al. 2013; Zou et al. 2014). While several practical semantic annotation systems have already been introduced

✉ Ebrahim Bagheri
bagheri@ryerson.ca

Yue Feng
luna.feng@thomsonreuters.com

Feras Al-Obeidat
Feras.al-obeidat@zu.ac.ae

[1] Thomson Reuters Labs, Toronto, Canada

[2] Laboratory for Systems, Software and Semantics (LS³), Ryerson University
http://ls3.rnet.ryerson.ca/

[3] Zayed University, Dubai, United Arab Emirates

[1] http://www.internetlivestats.com/twitter-statistics/.

Published online: 26 June 2018

**Table 1** Sample dominant use of Wikipedia entities on Twitter

| Mention | Wikipedia entity candidates | Dominant candidates |
|---|---|---|
| Apple | Cashew_apple, Custard_apple, Love_apple, Apple_(album), Apple_Inc....(52 entity candidates) | Apple_(fruit) Apple_Inc. |
| Java | Java_Sea, Java_Trench, Java_Alabama, Java_Road, Java_(programming_language), Java_(band), Java,_New_York, Chrysler_Java...(38 entity candidates) | Java_coffee Java_(programming_language) Java_Sea |
| Maze | Maze_(film), Maze_(band), Maze_(solitaire) Maze_(puzzle)...(39 entity candidates) | Maze_(puzzle) Maze_(band) |
| Balance | Balance_(accounting), Balance_(band), Balance_(1983 film), La_Balance...(35 entity candidates) | Balance_(ability) Balance_(accounting) |

in the community, they are not necessarily guaranteed to offer the best results in the case of Twitter content given the special characteristics of tweets, which are known to be short and noisy (Inches et al. 2010; Massoudi et al. 2011). These characteristics often affect the efficiency of existing techniques.

To address these challenges, recent works for semantic annotation consider textual context characteristics. For example, the work in Ferragina and Scaiella (2010) and Meij et al. (2012) is specially designed for annotating tweets. The central goal of such work is to link a mention within a *tweet* to the best Wikipedia entity. The challenging aspect of the tweet annotation process is to correctly link ambiguous mentions to the right entity despite the relatively short context. This is because for each ambiguous mention in a tweet, multiple entity candidates (also known as disambiguation options) are available. For instance, as shown in Table 1, the term Apple has 52 different entity candidates on Wikipedia (equivalent to 52 entries on Apple's disambiguation page on Wikipedia).[2] Existing techniques consider all of the candidates for an ambiguous mention as possibly valid disambiguation options.

## 1.1 Research objectives

Inspired by the early idea from Gale et al. (1992), which states that within a given discourse there is often one main sense for each term, and also by reviewing Twitter content, we developed a hypothesis that for a given ambiguous mention in a tweet, and within a given time interval, it is very unlikely that all of the possible entity candidates have an equal likelihood to be observed on Twitter. In other words, we hypothesize that from amongst the available entity candidates of an ambiguous mention, there are only a limited set of entity candidates that are actually being used on Twitter. Therefore, for a tweet that consists of a set of ambiguous mentions, it would be rational to consider only dominant entity candidates, i.e., the entities that are frequently observed on Twitter, for the purpose of disambiguation as opposed to considering the whole entity candidate set. Further to our example and as we will show later in the paper, there are primarily two dominant entity candidates for Apple on Twitter, referring to either the `Apple corporation`[3] or the `Apple fruit`[4] and the other 50 entities are very rarely, if at all, observed on Twitter.

Based on the dominant entity candidates hypothesis, the objectives of our work are to provide support for the annotation process in both offline and online stages:

- (offline) identify those entity candidates of ambiguous mentions that are frequently observed on Twitter as their dominant candidates. Our focus will be on identifying these candidates through an unsupervised approach. The dominant entity candidates are identified once within a certain time period in an offline process and will be used in the online annotation process;
- (online) perform semantic annotation for tweets by considering only the identified dominant entity candidates on the fly. The reason we are interested in identifying dominant entity candidates and only using them in the annotation process is that by doing so, we hope that the annotation process will be faster and more efficient.

## 1.2 Contributions

To address the research objectives of our work, we provide three technical contributions in this paper that relate to the identification of dominant entity candidates for ambiguous mentions and performing semantic annotation by only considering the dominant entity candidates in the annotation process. More concretely, the contributions of our work are as follows:

[2] https://en.wikipedia.org/wiki/Apple_(disambiguation).

[3] https://en.wikipedia.org/wiki/Apple_Inc.

[4] https://en.wikipedia.org/wiki/Apple.

- We propose a graph-based method to find the relevant term clusters for ambiguous mentions on Twitter;
- We formulate an approach for finding the most suitable Wikipedia entity for each of the identified term clusters to identify dominant entity candidates for each ambiguous mention; and
- We present an annotation technique that links a set of ambiguous mentions in a tweet to an unambiguous Wikipedia entity by only considering dominant entity candidates.

From a theoretical perspective, our work is among the early works that focus on the unsupervised mining of important and relevant entity candidates of ambiguous mentions from temporally aligned Twitter corpora. The dominant entity candidates are mined without considering a specific input tweet and are extracted in an offline process. From a practical perspective, this leads to statistically significant improved execution time compared to the state of the art while maintaining a competitive accuracy with the state of the art on 'un-aligned' gold standard datasets and improved accuracy on a temporally aligned gold standard dataset.

We evaluate our proposed approach on two publicly available datasets released in Meij et al. (2012) and Basave et al. (2014). Experimental results show that our method is competitive with other state-of-the-art baselines including supervised and non-supervised approaches in terms of precision and recall despite the fact that we only consider dominant entity candidates of an ambiguous mention and ignore the majority of the other candidates as disambiguation options. Furthermore, we will report that when the tweet that is being processed is temporally aligned with the corpus used for identifying the dominant entity candidates that our approach shows improved performance compared to the state-of-the-art techniques. We also show that our method has a significantly faster processing time compared to other techniques. Our source code, datasets and evaluation metrics can be accessed at https://github.com/lunafeng/ELTDS.

The remainder of this paper is organized as follows: In the next section, we cover the most relevant work to our paper. In Sect. 3, we present our proposed approach, which includes two phases, namely (i) dominant entity candidates detection, and (ii) named entity disambiguation. Section 4 is dedicated to the details of our experimental results. In Sect. 5, we discus some limitations of our method and finally Sect. 6 concludes the paper.

## 2 Related work

The task of linking textual mentions to the most relevant entities from structured knowledge bases has attracted a lot of attention over the past several years (Zou et al. 2014; Liu et al. 2013; Yamada et al. 2015; Tran et al. 2015; Huang et al. 2014; Shen et al. 2013). This task is primarily composed of two major steps: (i) The first step is concerned with the identification of the mentions that have the potential to be linked to some entity in the knowledge base. This involves performing tasks such as term expansion (Zou et al. 2014), abbreviated form expansion, and domain dictionary lookup (Yamada et al. 2015) to detect misspelled mentions and acronyms. (ii) The second step deals with assigning a candidate entity to the identified mentions from the first step based on a set of features that measure the relevance of the mention and the candidate entities. There are typically two types of features that have been used in the literature, namely local and global features. Local features include such things as the distance obtained from a cosine similarity measure, edit distance similarity, the probability of the mention serving as the anchor text for the entity candidate (Liu et al. 2013) and the temporal relevance of an entity candidate for a given mention (Tran et al. 2015).

Global features take a more comprehensive view towards candidate entity ranking where the relations between the entity candidates for the different mentions of the tweet are taken into consideration. For instance, Liu et al. (2013) introduce a collective inference model to link mentions in a tweet to entities of a knowledge base. The authors integrate two sets of global features to train their collective inference model, namely the entity-to-entity and the mention-to-mention similarity features. Through the use of these two sets of features, the authors try to link mentions to similar entities while preserving high total similarity between matched mention-to-entity pairs. They consider the inter-entity link structure amongst the pairs of entities on Wikipedia as a measure for entity-to-entity similarity. The mention-to-mention set of features consist of the textual similarity between pairs of tweets and whether they are from the same author. To combine the above-mentioned three sets of features, the authors employ a greedy hill-climbing approach in the training process to learn the best weighting coefficients for each of the features. Sarmento et al. (2009) disambiguate the entities on the Web by clustering the mentions such that each cluster refers to only one entity. To calculate the similarity of mentions, they create a feature vector, which is a TF/IDF vector based on the terms that the mention has co-occurred with, for each mention. In Zou et al. (2014), Zou et al. employ a belief propagation method based on topic distribution, instead of common links, to calculate the global features. The reason is that common links between entities can imply content similarity and subsequently, similar topic distribution. The method proposed by Habib and van Keulen (2012) assumes that the correct entities for mentions appearing in the same tweet should be related to each other in the YAGO KB graph. Their disambiguation algorithm gives the set of all candidate entities for the extracted mentions, then

the algorithm finds all possible permutations of the entities. For each permutation, they apply agglomerative clustering to obtain a set of clusters of related entities according to YAGO.

Similarly, a recent study by Li et al. (2016) intentionally removes the cross-links between the entities in the knowledge base from consideration. They propose a generative model instead, relying solely on textual content, to associate a mention to an entity in a linkless knowledge base. TagMe[5] (Ferragina and Scaiella 2010) is one of the better known semantic annotation tools, which has been specifically built for short text, and has shown to perform reasonably well on different datasets and for various benchmark metrics (Cornolti et al. 2013). TagMe uses Wikipedia anchor texts and pages to cross reference mentions with Wikipedia entities. Similar to the idea of global features, TagMe benefits from collective agreement between the entity associated with a mention and all of the other entities detected in the text. Different from TagMe, the work by Meij et al. (2012) performs entity linking by learning the importance of three types of features, i.e., n-gram features, concept features, and tweet features, in the linking task. The authors then use various machine learning techniques that are trained on a training set using ten-fold cross validation. The authors show that random forests or gradient boosted regression trees can improve the precision of the entity linking task.

For the purpose of determining the correct entity, some approaches adopt a graph-based representation for the interlinking of local and global features. For example, Shen et al. (2013) turn the tweet entity linking problem into a user-oriented graph-based interest propagation problem. They assume each user has a constant underlying topic interest distribution over various named entities and propose KAURI to collectively link mentions in all tweets posted by the user to the user's topics of interest. In a similar vein, Huang et al. (2014) propose a graph regularization model to collectively identify and at the same time disambiguate mentions within a tweet. This work is the only work in the literature that employs a semi-supervised method for tweet entity linking. In our work, we explore local and global features for the purpose of determining the correct entity. Although our annotation model is not graph-based, we utilize a graph-based method in our prepossessing step to identify dominant entity candidates. Most, if not all, of the above given works do not consider the fact that the choice for the most appropriate entity for a given mention could be influenced by time. In other words, these approaches build probability distributions based on the entity and text co-occurrence within the source corpus, e.g., Wikipedia, and use these distributions to calculate the local and global features. Therefore, these models will not be able to use dynamic information about the temporal co-evolution of mentions and entities. Tran et al. (2015) is one of the only few that considers the notion of temporality. The authors incorporate temporal information from the Wikipedia edit history and page view logs to link hashtags to entities. For instance, while '#sochi' refers to a city in Russia, the hashtag was used to report the 2014 Winter Olympics during February 2014. In our work, we also consider the notion of *temporality* as we determine the dominant entity candidates of ambiguous mentions on Twitter within certain time periods.

From a training perspective, tweet entity linking methods can be classified as supervised and unsupervised. Unsupervised models build probability distributions based on the characteristics of a source corpus. TagMe (Ferragina and Scaiella 2010) and DBpedia Spotlight (Daiber et al. 2013) are some examples of unsupervised methods. Such approaches would, therefore, perform in the same way regardless of the input tweets that need to be annotated. Supervised models, however, are trained and fine-tuned based on an initial set of labeled tweets and, therefore, would perform more suitably for the set that they are trained on. The work by Meij et al. (2012), Liu et al. (2013) and Wikify! (Mihalcea and Csomai 2007) are examples of supervised techniques. Huang et al.'s work (Huang et al. 2014) is the only work that has considered the semi-supervised approach for annotation and has reported competitive performance compared to supervised approaches with only 50% labeled data. Our annotation process is unsupervised and hence, does not require a training dataset.

Overall, while all the above given methods employ the complete entity candidate set defined on Wikipedia in the disambiguation process, we only consider entities that are frequently observed on Twitter, i.e., dominant entity candidates for each ambiguous mention within a tweet and ignore the majority of the other candidates. A similar idea is followed in Shirakawa et al. (2011) to disambiguate the entities mentioned in a document with dominant concepts that are detected from the input document. Their idea of using only dominant concepts to disambiguate entities is similar to our work; however, they identify the dominant concepts of each ambiguous mention in a given document using only the existing terms in that document. So, it cannot be applied for tweets which are short and noisy and and do not contain rich information. Furthermore, their dominant concept detection step is a part of their disambiguation process. However, our dominant entity candidates detection is an offline process which is done once on a Twitter corpus and is independent of the disambiguation step. Similarly, the methods proposed in Hoffart et al. (2011); Mena and van Keulen (2016) use the idea of prominent entities to cut out the long tail of candidates. However, despite these works, which calculate the prominence

[5] tagme.di.unipi.it.

of an entity in the context of Wikipedia, simply based on Wikipedia-based frequencies of link anchors referring to that entity (i.e., prior probability), we have extracted the dominant candidate entities of an ambiguous mention in the context of Twitter. The main difference is that the above given works do not consider the fact that the choice for the most appropriate entity for a given mention could be influenced by time. Our work rests on the hypothesis that a limited set of entity candidates for an ambiguous mention emerges within a specific time period on Twitter. Therefore, we detect the dominant candidate entities of mentions in the context of Twitter. In other words, since we detect dominant entity candidates from Twitter content, our method has the ability to capture the temporal information of ambiguous mentions. Furthermore, we explore local and global features to perform disambiguation. For a comprehensive review of named entity recognition and entity linking of tweets, we encourage the interested reader to see Derczynski et al. (2015).

The development of techniques that can automatically extract the semantics of tweets through entity linking has the potential to improve the quality of applications that need to process tweets, such as online reputation (Saleiro et al. 2017), filtering Twitter streams (Kapanipathi et al. 2011), user interest detection (Abel et al. 2011; Kapanipathi et al. 2014), and Tweet classification (Varga et al. 2014; Vitale et al. 2012), among others. For example, Saleiro et al. (2017) have proposed a framework that implements text mining techniques to measure the reputation of entities from tweets, i.e., to perform online reputation monitoring, which is able to collect texts from Twitter, and identify and disambiguate entities of interest and classify sentiment polarity and intensity. Abel et al. (2011) have proposed to enrich Twitter posts by linking them to related news articles and then extracting the semantic entities mentioned in the enriched posts using OpenCalais. The identified semantic entities are then used to build user interest profiles. Similarly, for the purpose of filtering Twitter streams, Kapanipathi et al. (2011) have modeled users' interests by annotating their tweets with DBPedia concepts. A similar idea has also been applied by Vitale et al. (2012) in the context of short text classification. They have designed a classification algorithm, which works on Wikipedia-based annotators which are able to extract the main topics of short texts.

It is important to note that work in entity linking is not limited to the domain of tweets and has been extensively explored in other closely related areas. Some recent examples include collective disambiguation through query expansion (Zhao Gang et al. 2016), search query annotation (Ganea et al. 2016; Cornolti et al. 2016; Bhatia and Jain 2016), large-scale entity linking to multiple knowledge bases (Gao and Cucerzan 2017), and disambiguation in linkless knowledge bases (Li et al. 2016), just to name a few.

## 3 Tweet entity linking

In this paper, similar to Zou et al. (2014); Liu et al. (2013), we define tweet entity linking as the problem of annotating named entity mentions $\mathcal{M}_t$ recognized in a tweet $t$ with suitable Wikipedia entities.

We sub-divide the problem of annotating tweets into two sub-problems: dominant entity candidate detection and named entity disambiguation. The dominant entity candidates for ambiguous mentions are identified once within a certain time period in an offline process and will become the input of the online disambiguation process. The details of each step are described in the following sections.

### 3.1 Dominant entity candidate detection

In this step, for a given ambiguous mention within a tweet, among all possible entity candidates defined on Wikipedia, our goal is to find entities that are frequently used on Twitter within a certain time period. To do so, given a set of ambiguous mentions $A$ within a collection of tweets $\mathbb{T}$ as a training corpus, we follow two steps: term cluster detection and entity mapping. By analyzing a collection of tweets $\mathbb{T}$, the goal of the term cluster detection process is to find all possible term clusters that are relevant to a given ambiguous mention on Twitter within a certain time period. In the mapping process, we aim at mapping each identified term cluster to a Wikipedia entity to form the dominant entity candidates for the ambiguous mention.

To be able to recognize named entity mentions from tweets, we build a dictionary called mention dictionary which includes the surface forms of the Wikipedia named entities such as name variations, abbreviations, and spelling forms. To do so, as suggested in Cucerzan (2007), we use four sources of information in Wikipedia, namely entity pages, redirect pages, disambiguation pages and hyperlinks. Then, for each tweet $t \in \mathbb{T}$, we extract all possible n-grams from the tweet $t$ and then detect its mentions by querying the mention dictionary for each n-gram. Among the identified mentions, ambiguous mentions are those that have more than one possible entity candidates on their Wikipedia disambiguation page.

#### 3.1.1 Term cluster detection

To be able to identify the various entity candidates of a given ambiguous mention $a \in A$ from Twitter content, we

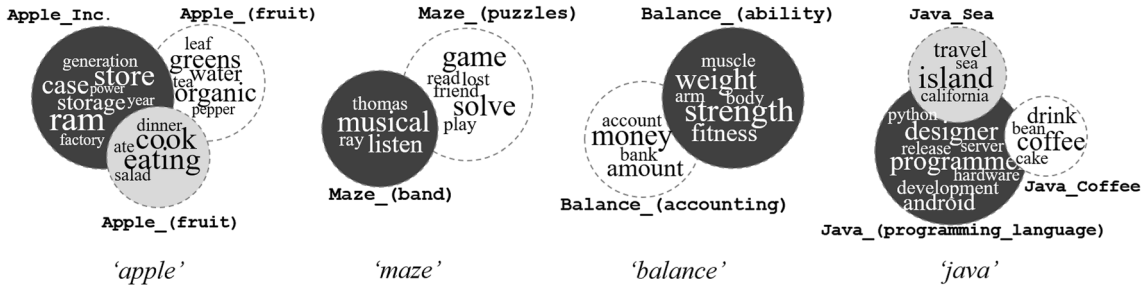


**Fig. 1** Sample term clusters for some ambiguous mentions

adopt the latent relation hypothesis that states that terms appearing in the same context tend to have related semantics (Turney 2008). In case of Twitter, the hypothesis would mean that the terms that appear together in the same tweets would carry similar or related semantics. Figure 1 shows the most related terms that were found for each of the ambiguous mentions listed in Table 1. As shown in the figure, once such semantically related terms are identified, it would be possible to see clusters of highly similar terms to each other that would form the various contexts in which the ambiguous term is used. Based on this observation, we propose an unsupervised technique that can automatically identify different term clusters which are related to an ambiguous mention on Twitter. Let us describe this process more formally.

Given an input Twitter corpus $\mathbb{T}$, to extract the related term clusters to the identified ambiguous mentions $\mathbb{A}$, we create a graph, called the term dependency graph, based on the latent relation hypothesis as follows:

**Definition 1** *(Term Dependency Graph)* A term dependency graph denoted as $\mathcal{G} = (\mathbb{V}, \mathbb{E}, g)$ is a weighted graph in which $\mathbb{V}$ includes all of the terms in a Twitter corpus $T$. $\mathbb{E}$ denotes a set of weighted edges $e_{w_i,w_j}$ from term $w_i$ to term $w_j$ whose weight $g(e_{w_i,w_j})$ is calculated using the following conditional dependency between terms:

$$g(e_{w_i,w_j}) = \mathrm{P}(w_j|w_i) = \frac{f(w_i, w_j)}{\sum_{w_k \in V} f(w_i, w_k)} \tag{1}$$

where $f(w_i, w_j)$ is the number of times terms $w_i$ and $w_j$ have co-occurred in the same tweet. To find all terms from $\mathbb{T}$, we extract all possible n-g from each tweet $t \in \mathbb{T}$.

Now, given the term dependency graph $\mathcal{G}$, and an ambiguous mention $a \in \mathbb{A}$, if the mention $a$ belongs to the extracted terms from the input Twitter corpus $T$, it is possible to identify the terms that have been most frequently observed with the ambiguous mention $a$. To do so, we apply the random walk algorithm (Lawler and Limic 2010) to find the related terms to $a$ denoted as $\mathbb{R}_a$, by starting the walk of a particle at the source node $a$. The probability of finding the particle at a certain node such as $w_j \in \mathbb{V}$ after $l$ iterations is equivalent to the sum of all paths through which the particle could have reached $w_j$ starting from any other node at iteration $l-1$. Formally:

$$w_j^{(l)} = \sum_{w_k \in \mathbb{V}} w_k^{(l-1)} P(w_j|w_k) \tag{2}$$

The stationary distribution for the target ambiguous mention $a \in \mathbb{V}$ is obtained when the stationary distribution does not significantly change and can be defined as follows:

$$v(a)^{(l)} = \phi v(a)^{(0)} + (1-\phi) M_{\mathcal{G}} v(a)^{(l-1)} \tag{3}$$

where $v(a)^{(0)}$ is an initial distribution that places all of the probability mass on a single node, $\phi$ is the parameter to update the distribution at each iteration and $M_{\mathcal{G}}$ is the transition matrix associated with the term dependency graph $\mathcal{G}$.

Given the stationary distributions of terms, for an ambiguous mention $a$, we build $\mathbb{R}_a$ using the terms in the term dependency graph $\mathcal{G}$ which are related to $a$.

Once we have the related terms for an ambiguous mention $a$, i.e., $\mathbb{R}_a$, our next step is to identify relevant term clusters for $a$. To do so, we first build a graph, called the context graph, $\mathcal{CG}_a$, as follows:

**Definition 2** *(Context Graph)* a Context graph for an ambiguous mention $a$, denoted as $\mathcal{CG}_a = (\mathbb{V}, \mathbb{E}, \gamma)$, is a weighted undirected graph in which $\mathbb{V}$ is the set of all related terms to $a$, i.e., $\mathbb{V} = \mathbb{R}_a$, $\mathbb{E}$ denotes a set of edges, and the weight function $\gamma$ represents semantic relatedness between every two node in $\mathbb{V}$.

In our work, we compute the semantic relatedness between each two terms based on our earlier method proposed in Feng et al. (2015). We adopt this semantic relatedness method instead of other existing state-of-the-art semantic relatedness methods, because it is particularly designed for the Twitter sphere. It measures the semantic relatedness of terms on Twitter by constructing graph representation

of terms mentioned in tweets and applying a random walk procedure to produce a stationary distribution for each term.

Finally, to be able to identify the set of all contexts of an ambiguous mention from $\mathcal{CG}_a$, we would need to find separable term clusters within this graph. To do so, we focus on the fact that the terms about a certain context are highly related to each other and terms from distinct context do not share much relatedness. For example, as shown in Fig. 1, given an ambiguous mention apple, one set of terms consists of {fruit, greens, organic, leaf, pepper} while another includes {model, factory, store, generation, RAM, Mac}. While there is a high relatedness within each set, there is not too much similarity between the two sets. This implies that clusters within the context graph could potentially represent the possible context of an ambiguous mention. In our experiments, we exploit some of the well-known and frequently used clustering algorithms namely Louvain (Blondel et al. 2008), graph-based k-means (Ferrer et al. 2009) and agglomerative hierarchical clustering (Mannor et al. 2004), and compare their performance.

Louvain (Blondel et al. 2008) is an efficient heuristic method that finds clusters by optimizing both modularity and extraction time on a weighted graph. The k-means clustering algorithm (Arthur and Vassilvitskii 2007) is a widely used clustering technique that aims at minimizing the average squared distance between points in the same cluster and maximizing inter-cluster dissimilarity. As introduced in Ferrer et al. (2009), we use a graph-based version of the k-means clustering algorithm in which the generalized median graph is used to obtain a representative of each cluster as centroid computation. Finally, the *agglomerative* clustering method (Mannor et al. 2004) performs hierarchical clustering using a bottom-up approach and builds nested clusters by merging or splitting them successively.

Therefore, for a given ambiguous mention $a \in \mathbb{A}$, we apply clustering algorithms on $\mathcal{CG}_a$ to cluster the terms into distinctive term clusters. As a result, each ambiguous mention $a$ is associated with a set of term clusters $\mathbb{TC}_a$, each member of which includes highly semantically coherent terms.

#### 3.1.2 Entity mapping

To employ the identified term clusters for an ambiguous mention $a \in A$ in the disambiguation process as the dominant entity candidates, we need to map each term cluster to an appropriate Wikipedia entity. To do so, given a term cluster of an ambiguous mention $a$, i.e., $tc \in \mathbb{TC}_a$, we aggregate all of the terms included in the term cluster $tc$ as a single document and then calculate its similarity with the Wikipedia summary of each candidate entity in the disambiguation page of the corresponding Wikipedia entity of $a$. We map $tc$ to the Wikipedia entity with the highest similarity. To calculate the similarity, we employ various similarity methods in our experiments, we adopt three state-of-the-art document similarity methods to compare their performance in our model: (i) Words Match Similarity (Gomaa Wael and Fahmy Aly 2013); (ii) UMBC Phrase Similarity[6] and (iii) UMBC Semantic Textual Similarity.[7] The document similarity methods proposed by UMBC (Han et al. 2013) are based on distributional similarity and Latent Semantic Analysis (LSA) (Dumais 2004) combined with semantic relations extracted from WordNet[8] and assume the semantics of a phrase is dependent on its component words.

As an example, Fig. 1 shows that the identified term clusters for Apple are mapped to two Wikipedia entities Apple_(fruit) and Apple_Inc.

In the mapping process, we may find the same Wikipedia entity for multiple term clusters of an ambiguous term. For example, as shown in Fig. 1, for the ambiguous mention apple, two term clusters are mapped to the same Wikipedia entity Apple_(fruit). In such cases, we will merge the two or more term clusters that were mapped to the same Wikipedia entity into one cluster. It is also possible that an identified term cluster refers to a newly emerging entity, which is not formally represented in Wikipedia and consequently, there is no appropriate entity in Wikipedia to link that cluster to. In such cases, we simply ignore that identified term cluster.

At the end of this process, for each ambiguous mention $a \in A$, we have a set of Wikipedia entities which are mapped to its term clusters $TC_a$. We call these Wikipedia entities as the dominant entity candidates of the ambiguous mention $a$, denoted as $DE_a$.

To clearly show the significant difference between the number of entity candidates which are frequently employed on Twitter compared to the total number of entity candidates for each entity in Wikipedia, in Fig. 2, we present the list of 945 ambiguous mentions available in the dataset from Meij et al. (2012) and compare the number of candidates obtained from Wikipedia disambiguation pages and our Twitter corpus. This illustrates that the number of dominant entity candidates that we identify on Twitter is significantly less than the number of candidates formally defined on Wikipedia.

### 3.2 Named entity disambiguation

In the disambiguation step, given a set of mentions $\mathcal{M}_t$ for tweet $t$, we are interested in linking each mention $m \in \mathcal{M}_t$ to a Wikipedia entity $c \in \mathbb{C}$. To do so, we look up every $m$ in

[6] http://swoogle.umbc.edu/SimService/phrase_similarity.html.

[7] http://swoogle.umbc.edu/StsService/index.html.

[8] https://wordnet.princeton.edu/.

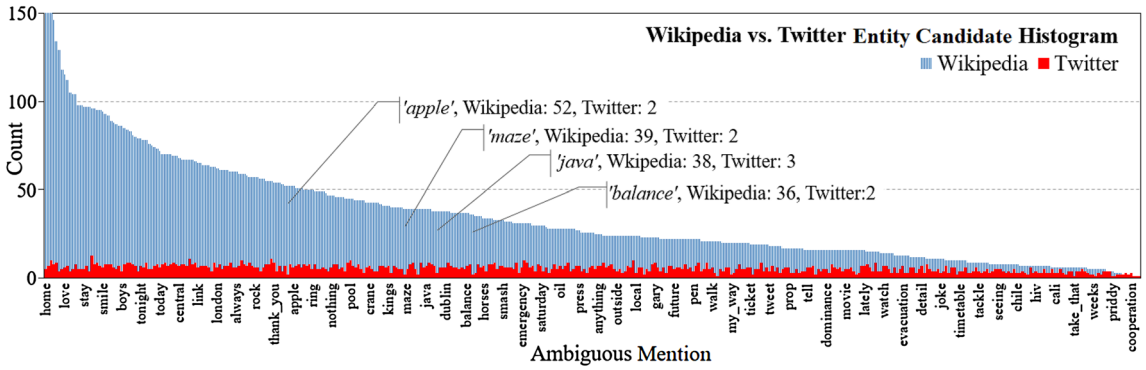


**Fig. 2** Comparison between the number of entity candidates defined on Wikipedia and dominant entity candidates on Twitter

the mention dictionary built in the dominant entity candidate detection step, which will result in three cases:

- If a hit is not found, we are not able to annotate $m$ with a Wikipedia entity and therefore we return NIL for $m$.
- if there is only one possible entity candidate on Wikipedia for the mention $m$, the mention is unambiguous and we directly link $m$ to the corresponding Wikipedia entity $c$.
- if $m$ belongs to the set of ambiguous mentions $\mathbb{A}$, we consider its corresponding dominant entity candidates $\mathbb{DE}_m$ (as derived in Sect. 3.1.1) to be the candidates for disambiguation.

As an example, for the tweet `'#NP Frankie Beverly and Maze Before I let go'`, there are two mentions, Frankie Beverly and Maze. Because there is only one possible Wikipedia entity for Frankie Beverly, i.e., Frankie_Beverly,[9] we directly link it to this entity. However, the mention Maze is ambiguous according to its Wikipedia disambiguation page and it has 39 entity candidates defined on Wikipedia. As shown in Table 1, for the mention *Maze*, two Wikipedia entities Maze_(puzzle) and Maze_(band) are identified as the dominant entity candidates and consequently we only consider these two Wikipedia entities as candidates for disambiguation.

To associate an ambiguous mention $m \in \mathbb{A}$ to the best candidate from the set of its dominant entity candidates $\mathbb{DE}_m$, we implement two similarity methods as follows:

*Context-based similarity* Based on the intuition that each annotation in a tweet should be related to the context of the tweet, we consider the similarity between each candidate $de \in \mathbb{DE}_m$ and the target tweet $t$. To do so, we apply a document similarity method to calculate the similarity between $t$ and the summary of the Wikipedia entity to which $de$ is mapped, in the mapping step of dominant entity candidate detection, as another document. The dominant entity candidate with the highest similarity score will be selected as the annotation for that mention.

*Collective similarity* We leverage the global coherence between candidate entity and apply collective similarity as defined in Eq. 4 by considering both context similarity and the similarity between the candidate entities. In other words, in this disambiguation approach, the objective is to select those senses for the mentions that are not only similar to the tweet but are also highly similar to each other. As such, the senses selected for the mentions observed in the tweet will be selected in a way that have the highest similarity with each other.

**Definition 3** *(Collective similarity)* Given a set of $k$ mentions for tweet $t$, $\mathcal{M}_t = \{m_1, m_2, \ldots, m_k\}$, and their dominant entity candidates $\mathbb{DE}_{m_1}, \ldots, \mathbb{DE}_{m_k}$ as candidates of each mention, we let CP be the Cartesian product over $k$ dominant entity candidates, $\text{CP} = \mathbb{DE}_{m_1} \times \cdots \times \mathbb{DE}_{m_k}$. Collective similarity for each combination $\text{CP}_i \in \text{CP}$, $\text{ColSim}(\text{CP}_i)$ is calculated as follows:

$$\text{ColSim}(CP_i) = \prod_{j=1}^{|CP_i|} \prod_{k=j+1}^{|CP_i|} \text{Sim}(\text{CP}_i[j], \text{CP}_i[k]) \times \text{Sim}(\text{CP}_i[j], t) \times \text{Sim}(\text{CP}_i[k], t), \tag{4}$$

where Sim() is a function that measures document-based similarity. Finally, we select the combination $\text{CP}_i \in \text{CP}$ with the highest score, $\text{ColSim}(\text{CP}_i)$, as the annotation set for the target tweet $t$.

## 4 Experiments

In this section, we describe our experiments in terms of the dataset, setup and execution time performance compared to the state of the art.

[9] https://en.wikipedia.org/wiki/Frankie_Beverly.

**Table 2** Results based on different parameter combinations

| Clustering method | Mapping method | Disambiguation method | Precision | Recall | F1 |
|---|---|---|---|---|---|
| Louvain | WordsMatch | Context-based | **0.765** | **0.581** | **0.660** |
| | | Collective | 0.739 | 0.547 | 0.629 |
| | UMBC phrase | Context-based | 0.728 | 0.530 | 0.613 |
| | | Collective | 0.683 | 0.50 | 0.577 |
| | UMBC STS0 | Context-based | 0.723 | 0.530 | 0.612 |
| | | Collective | 0.675 | 0.489 | 0.567 |
| k-means | WordsMatch | Context-based | 0.744 | 0.563 | 0.641 |
| | | Collective | 0.723 | 0.542 | 0.620 |
| | UMBC phrase | Context-based | 0.725 | 0.526 | 0.610 |
| | | Collective | 0.688 | 0.494 | 0.575 |
| | UMBC STS0 | Context-based | 0.712 | 0.523 | 0.603 |
| | | Collective | 0.677 | 0.500 | 0.575 |
| Hierarchical | WordsMatch | Context-based | 0.731 | 0.542 | 0.622 |
| | | Collective | 0.715 | 0.533 | 0.611 |
| | UMBC phrase | Context-based | 0.693 | 0.510 | 0.588 |
| | | Collective | 0.683 | 0.490 | 0.571 |
| | UMBC STS0 | Context-based | 0.712 | 0.522 | 0.602 |
| | | Collective | 0.667 | 0.499 | 0.571 |

Bold indicates the best performing model

### 4.1 Gold standard datasets

We evaluate our method on the following datasets as the gold standards to demonstrate *robustness* across different datasets:

*Micropost-2014 test* (Basave et al. 2014): This Twitter dataset was introduced in the 'Making Sense of Microposts' challenge and consists of 1165 tweets. In this dataset, the average number of annotations for each tweet is 2.5.

*Meji's dataset* (Meij et al. 2012): This Twitter dataset, which is published by Meij et al. (2012) consists of 502 tweets which are semantically annotated. In this dataset, the average number of annotations for each tweet is 2.17. There are two other datasets released by Tran et al. (2015) and Shen et al. (2013). However, we were not able to use them because they are limited only to annotations for trending hashtags and annotations customized to specific users and are not publicly available.

### 4.2 Metrics

Given the gold standard datasets, we adopt the evaluation metrics that have been used in the related literature (Liu et al. 2013; Yamada et al. 2015; Ferragina and Scaiella 2010; Huang et al. 2014) to evaluate the quality of our work. We determine the quality of the annotations using standard information retrieval metrics including Precision, Recall and F-measure and compare the performance of our proposed method with other state-of-the-art benchmarks.

### 4.3 Experimental setup

As described in Sect. 3.1, we utilize Twitter to find the dominant entity candidates for ambiguous mentions. For this purpose, we use the publicly available Twitter dataset[10] released by Cheng et al. (2010) as our training corpus. It consists of approximately 8 million tweets posted by 106,349 unique users between 10 Nov 2006 and 17 March 2010. In this corpus, the average number of terms in each tweet is 8.4 and there are 4 million unique terms available in the corpus.

Furthermore, there are three main variation points in our proposed approach, which can affect the performance of our results:

*Clustering methods* As mentioned in Sect. 3.1.1, we require a clustering method to detect term clusters related to an ambiguous mention. We select three different clustering algorithms, namely Louvain (Blondel et al. 2008), graph-based k-means (Ferrer et al. 2009) and agglomerative hierarchical clustering (Mannor et al. 2004).

It should be noted that Louvain does not require a priori knowledge of the number of clusters ($\beta$) when running the algorithm and $\beta$ is determined by the algorithm itself. However, the other two algorithms require the number of clusters to be predefined. Therefore, we apply $\beta$ obtained from Louvain as the number of clusters in the other two methods k-means and agglomerative. The results are reported in Table 2. In addition to using $\beta$ as the

[10] https://archive.org/details/twitter_cikm_2010.

number of clusters in these two methods, we also evaluated larger and smaller cluster sizes around $\beta$. According to our experimental results, we observed that varying the number of clusters around $\beta$ does not lead to any meaningful improvements in the final results in either k-means or agglomerative clustering. Therefore, we do not report these results in Table 2.

*Similarity methods in entity mapping* As mentioned in Sect. 3.1.2, we apply three state-of-the-art document similarity methods to map a term cluster to a Wikipedia entity: (i) Words Match Similarity (Gomaa Wael and Fahmy Aly 2013); (ii) UMBC Phrase Similarity and (iii) UMBC Semantic Textual Similarity.

*Disambiguation methods* As introduced in Sect. 3.2, we implement two disambiguation methods, namely context-based similarity and collective similarity. The choice of the disambiguation method can impact the performance of the annotation process. We report our experimental results for both of the disambiguation methods.

By selecting and combining the different alternatives for these three variation points, we obtain 18 variants (3 clustering techniques $\times$ 3 document clustering methods $\times$ 2 disambiguation techniques) that are evaluated and compared using the gold standard dataset in terms of precision, recall and F-measure. The results are shown in Table 2. By fixing two of the variation points, i.e., the clustering and mapping methods, we can compare different annotation methods. Based on Table 2, context-based similarity performs better than Collective Similarity in our work in terms of all the three evaluation metrics. For instance, if we select the Louvain clustering method and the words match mapping method, the precision, recall, F-measure for context-based similarity is 0.765, 0.581, 0.660, respectively, while the same variant but with a collective similarity results in a lower performance of 0.739, 0.547, 0.629. Given several researchers (Kulkarni et al. 2009; Han et al. 2011) have mentioned that collective similarity performs better than other methods, we looked further for the reason why our observation was to the contrary. Based on our observations, we found that some Twitter users cram multiple pieces of information into one short-length tweet or there are tweets that cover multiple aspects that can mislead a collective similarity approach. Let us consider the following tweet: '`Dad doing his best charlie sheen impression. WINNINGGGGGG`'. In this tweet, when a collective disambiguation approach is used the term Dad is linked to the `Dad_(Angel)` entity to collectively disambiguate it with Charlie Sheen. In this case `Dad_(Angel)` is more similar to `Charlie_Sheen` compared to the correct entity which is `Dad`. There are many similar cases that are observed in tweets that can mislead a collective similarity approach when annotating tweets and hence result in its poorer performance. Based on this, we select context-based similarity as the choice for the disambiguation technique.

Similarly, we can compare the three mapping methods with each other. By fixing the clustering method and the annotation method, we observe that the words match similarity mapping method produces better results. By comparing the three clustering methods, it can be observed that using the Louvain clustering method results in higher quality annotations in terms of the three evaluation metrics. Therefore, we select the variant with the best performance to be compared with the state-of-the-art baselines, i.e., the variant composed of Louvain clustering, words match mapping and context-based similarity.

## 4.4 Comparison with baseline methods

In this section, we first introduce the baseline methods and then compare the quality and efficiency of our proposed method with the baselines to answer the following research questions:

**RQ1.** Would a semantic annotation technique that only considers the dominant entity candidates extracted from Twitter be able to perform competitively with the state-of-the-art annotation systems that consider the whole candidate space in terms of precision and recall (see Sects. 4.4.2 and 4.4.3)?

**RQ2.** Would the consideration of only a limited set of entity candidates significantly reduce the execution time of the named entity disambiguation process (see Sect. 4.4.4)?

### 4.4.1 Baseline methods

The baselines selected for comparison can be divided into three categories: (i) supervised, (ii) semi-supervised and (iii) unsupervised methods. Baselines belonging to the first category include Rysann (Cuzzola and Bagheri 2014), (Liu et al. 2013), Wikify! (Mihalcea and Csomai 2007) and (Meij et al. 2012). Rysann utilizes a probabilistic model that relies on a hybrid gaussian–hypergeometric combination to resolve ambiguities by producing the statistics on the distribution of words within each DBpedia concept, then a supervised training process is required to determine the hyper parameters. Liu et al. (2013) combine three types of local, entity similarity and mention similarity features. To combine these three types of features, they require a training process to determine the weight for each feature type. Wikify! (Mihalcea and Csomai 2007 )uses a combination of knowledge-based and data-driven methods and measures agreement using a voting schema to perform disambiguation. The knowledge-based method is based on the overlap between the context of the potential concept and the keywords mentioned in the input

**Table 3** Comparative analysis of the performance of different baselines on the Micropost 2014 dataset

| Method | Precision | Recall | F1 |
|---|---|---|---|
| AIDA | 0.534 | 0.365 | 0.433 |
| TagMe | 0.421 | 0.401 | 0.411 |
| Spotlight | 0.625 | 0.453 | 0.525 |
| Rysann | 0.374 | 0.321 | 0.345 |
| Our method | 0.661 | 0.455 | 0.539 |

text and the data-driven method uses a Naive Bayes classifier to integrate both local and topical features. Meij et al. (2012) employ machine learning algorithms to focus mainly on the effectiveness of semantic linking as opposed to efficiency. As mentioned earlier, (Huang et al. 2014) is the only work that benefits from a semi-supervised approach where a smaller set of labeled data is required for their method. The main differences between our method and the above supervised methods are that our method is unsupervised which does not require labeled data for training and only considers the dominant entity candidates in its disambiguation phase.

Unsupervised methods selected as our baselines include AIDA (Yosef et al. 2011), which is an online tool for entity detection and disambiguation that maps the mentions in a text onto entities in the YAGO knowledge base. To do so, the authors build a weighted graph of mentions and candidate entities, and compute a dense subgraph that approximates the best joint mention-entity mapping by taking into account three features for the disambiguation process: popularity prior for entities, similarity between the context of the mention and its candidates, and the coherence among candidate entities for all mentions in a text. TagMe is the other unsupervised annotation system (Ferragina and Scaiella 2010), which processes Wikipedia anchor texts and pages to cross reference mentions with Wikipedia articles. Also, DBpedia Spotlight (Daiber et al. 2013) builds a generative probabilistic model by processing the Wikipedia links with their anchor texts and textual context.

The above-mentioned unsupervised methods use external knowledge resources such as Wikipedia to obtain entity candidates for the purpose of disambiguation; however, in our work, we utilize Twitter to generate dominant entity candidates and only employ the identified dominant candidates to perform disambiguation.

**Table 4** Comparative analysis of the performance of different baselines on the Meij dataset

| | Method | Precision | Recall | F1 |
|---|---|---|---|---|
| Supervised | Rysann | 0.752 | 0.595 | 0.664 |
| | Liu's method | 0.752 | 0.675 | 0.711 |
| | Wikify! | 0.375 | 0.421 | 0.396 |
| | Meij's method | 0.734 | 0.632 | 0.679 |
| Semi-supervised | Huang's method | 0.658 | 0.419 | 0.512 |
| Unsupervised | AIDA | 0.294 | 0.164 | 0.211 |
| | TagMe | 0.776 | 0.60 | 0.677 |
| | Spotlight | 0.621 | 0.453 | 0.524 |
| | Our method | 0.765 | 0.581 | 0.660 |

#### 4.4.2 Comparison based on gold standards

The results of our comparison with the state-of-the-art baselines on the Micropost-2014 gold standard are reported in Table 3. To produce the results of the baselines, we adopt AIDA,[11] Rysann,[12] TagMe[13] and Spotlight[14] whose implementations are publicly available. We used their RESTful API to annotate the tweets of the Micropost-2014 dataset. Unfortunately the implementations of the other baselines are not available and we could not obtain their annotation results for the Micropost-2014 dataset.

As shown in Table 3, our proposed method outperforms the other baselines in terms of all metrics, which means considering only dominant entity candidates instead of all the candidates introduced in Wikipedia in the disambiguation step leads to improved accuracy in the semantic annotation process.

To demonstrates the robustness of our method across different datasets, we additionally compared our method with other state-of-the-art baselines on the Meij's gold standard. The results are reported in Table 4. It should be noted that, as for baselines such as Liu's Method (Liu et al. 2013) and Huang's Method (Huang et al. 2014), we report their results obtained on the same gold standard dataset as reported in their papers. With regards to the results for the method proposed by Meij et al. (2012) and Wikify! (Mihalcea and Csomai 2007), we employ the results reported in Liu et al. (2013), which uses the same gold standard dataset. Unfortunately the code for these methods is not available.

[11] https://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/aida/.

[12] https://denote.rnet.ryerson.ca/rysann.

[13] https://tagme.di.unipi.it/tagme_help.html.

[14] https://github.com/dbpedia-spotlight/dbpedia-spotlight/wiki/Web-service.

As shown in Table 4, within the supervised method category, the best results in terms of all metrics are obtained by Liu's Method (Liu et al. 2013). In their work, they consider not only local features, but also global features related to entity similarity and mention similarity and the results indicate the effectiveness of collective inference and global features. It is important to note that supervised methods require high-quality labeled data in practice and need to be retrained when being applied to a new collection of tweets.

Based on the results, within the unsupervised category, TagMe performs the best. TagMe processes all Wikipedia pages which results in 3M anchors, 2.7M pages with a link-graph of about 147M edges, and computes a score for each possible entity candidate from Wikipedia for a mention to perform disambiguation. As reported in the results, while TagMe shows the best performance overall, the performance of our approach is highly competitive with TagMe in all three metrics, i.e., precision (0.776 vs 0.765), recall (0.60 vs 0.581) and F-measure (0.677 vs 0.66). This can be viewed as a notable achievement when considering the fact that we only process the dominant entity candidates of ambiguous mentions obtained from an 8M tweet corpus. According to the ambiguous mentions that were present in the gold standard, the average number of entity candidates defined on Wikipedia is 28 entities, while we reduce this number to 5 based on the dominant candidates that were identified.

It is important to note that, due to focusing only on the dominant entity candidates, one limitation of our work is that it is not able to annotate tweets with rare entities in Twitter. For example, for a given tweet `'@danieltosh new stand up special making me cry laughing. jesus christ he's funn'`, the mention detected is 'stand up' which is ambiguous according to its Wikipedia disambiguation page and it has 65 entity candidates defined on Wikipedia. We found 7 dominant candidate entities for this ambiguous mention, which includes Stand_Up!_(song), Stand_Up_and_Take_Action, Bobby_Womack, Blue_King_Brown, 3_Words, Marathon, and The_Ben_Cohen_StandU_Foundation. However, in this tweet the ambiguous mention *'stand up'* refers to Stand-up_comedy,, which is not included in the dominant entity candidates. Hence, our method cannot produce the correct annotation in this case.

On such basis, one of the concerns that needs to be further investigated is whether the errors or omissions by our proposed method are due to the entity candidates being incorrectly omitted when dominant candidates were not detected or not. To understand the source for the annotation errors or omissions that are made by our proposed technique, we manually reviewed all of the annotations that were generated against the gold standard and classified the errors and omissions into two categories: (1) errors or omissions that happened due to a missing entity eliminated in the dominant entity candidate detection process, and (2) errors or omissions due to incorrect disambiguation. We found that in total and out of the 327 erroneous or missing annotations, only 75 ($\sim 22\%$) were due to the exclusion of the correct entity in the dominant entity candidate detection process. This is a significant observation, which shows that dominant entity candidates provide a reasonably high coverage of the right entities that are needed in the named entity disambiguation process.

**Table 5** Comparative analysis of the performance of the baselines based on random sampled tweets

| Method | Precision | Recall | F1 |
|---|---|---|---|
| AIDA | 0.101 | 0.063 | 0.077 |
| TagMe | 0.707 | 0.578 | 0.636 |
| Spotlight | 0.525 | 0.342 | 0.414 |
| Rysann | 0.717 | 0.519 | 0.602 |
| Our method | 0.788 | 0.626 | 0.698 |

#### 4.4.3 Comparison based on random sampled tweets

It is worth noting that we identified the dominant entity candidates that were used in our experiments from a corpus of only 8M tweets. Given the limited size of our Twitter corpus, it is possible that some of the ambiguous mentions in the gold standard were not observed in the training Twitter corpus at all, which could have impacted our performance. As an example, in `'@yosoyjuanson are you REALLY in tasmania?? go to the MONA MUSEUM!! email me & i'll tell you who to talk to there!!!'`, the mentioned tasmania did not exist in our training Twitter Corpus, therefore, we were not able to detect any of its dominant entity candidates. Such cases impact the performance of our model in terms of recall reported in the previous section.

Furthermore, the basic hypothesis of our work is that a tweet should be annotated based on the dominant entity candidates of ambiguous mentions on Twitter. This hypothesis implicitly carries the fact that dominant entity candidates of ambiguous mentions can change over time. Therefore, ambiguous terms within a tweet would need to be annotated with dominant candidates detected within the time period when the tweet was posted. However, given the fact that we were interested in comparing with the state-of-the-art gold standard, there may have been temporal mis-alignment between the tweets in the gold standard and our Twitter corpus could have affected the precision of our work. The best performance of our work is achieved when the training Twitter corpus and the tweets that are being annotated belong to the same time period and hence there is alignment between the dominant entity candidates and the tweets. For instance, for the term Apple, in our Twitter corpus, we only

found `Apple_(fruit)` and `Apple_corporation` as the dominant candidates; however, it is possible that within a different time frame when the musician Fiona Apple is releasing a new album that the entity `Fiona_Apple` may turn out to be a part of the dominant candidates as well. To show that temporal alignment matters in our approach, we created a third benchmark dataset that shares the same temporal alignment with our Twitter training corpus.

We adopted the approach proposed in Tran et al. (2015) in order to create a benchmark dataset, and sampled tweets from our Twitter corpus. The sampled tweets were selected such that they each had at least five English words and included at least one ambiguous mention. We asked two volunteers to manually annotate these tweets to create the gold standard and ensured that the Kappa inter-rater agreement was above 0.8.

We compared our method with AIDA, TagMe, Spotlight and Rysann whose implementations are publicly available. The results are reported in Table 5. It is important to note that comparison with TagMe is considered to be a good indication of performance, as TagMe had one of the best performances on the previous gold standard datasets. The results show that if the tweets are annotated based on the dominant entity candidates detected from a temporally aligned Twitter corpus that our method outperforms other state-of-the-art techniques (both supervised and unsupervised methods) in terms of precision, recall and F-measure.

In summary, our proposed work is an unsupervised method that generates dominant entity candidates from the context of Twitter without relying on all candidates from other knowledge bases and yet produces results that are competitive with state-of-the-art techniques. Based on the observed performance and comparison with the state-of-the-art technique, we conclude that we can positively respond to our research question (RQ1) and conclude that the consideration of the dominant entity candidates for ambiguous terms on Twitter can positively enhance the semantic annotation of tweets.

#### 4.4.4 Execution time performance

In this section, we are interested in answering our second research question (RQ2) as to whether 'the consideration of only a limited set of entity candidates significantly reduces the annotation process time of tweets?' To this end, we compare the execution time of the different baselines for annotating the tweets in the primary gold standard. The experiments were conducted on an Intel(R) Xeon(R) 3.50 GHz with 40 GB RAM. We first deploy the baseline methods, whose implementations were publicly available, on our server and then calculate their execution time for annotating each tweet in seconds. Given a downloadable version of AIDA was not available and we had to resort to using the publicly available Web installation of AIDA, we report the time returned by the service under 'runtime', which does not include communication and transfer overheads. It should be noted that we have only compared the annotation time of different baselines together and have not considered the execution time of their pre-processing steps. For example, in case of our method, we have not considered the execution time needed for detecting dominant entity candidates of the training Twitter corpus, because the dominant entity candidates are identified once in an offline process and are used in the online annotation process. The same applies for the other baselines, i.e., Rysann, TagMe, Spotlight and AIDA, where their preprocessing step is not included in the reporting of the execution time.

**Table 6** The mean and standard deviation of the execution times (in seconds)

| Method | Mean | STDev |
|---|---|---|
| AIDA | 0.004 | 0.003 |
| TagMe | 0.005 | 0.019 |
| Spotlight | 0.013 | 0.003 |
| Rysann | 0.003 | 0.001 |
| Our method | 0.001 | 0.001 |

The mean and standard deviation (STDev) of the results for each method are shown in Table 6. Based on the results, our proposed method is the most efficient in terms of execution time, which is primarily due to two reasons: (i) it only considers a small set of entity candidates for the purpose of disambiguation, and (ii) it only uses context-based similarity, which is much less time-consuming compared to collective similarity. To determine statistical significance of the results, we ran a paired $t$ test between the execution times reported by our method for each tweet compared to Rysann, which is the next fastest approach. We obtained a $p$ value of $< 0.01$, which shows statistically significant difference between the execution time of our approach compared to Rysann.

Based on these results, it is possible to answer both research questions (RQ 1 and 2) simultaneously that by relying only on dominant entity candidates for the purpose of disambiguation, the entity linking process can be performed significantly faster while maintaining a competitive performance in terms of precision and recall.

## 5 Limitations

Recent studies have shown that trending topics and user's interests on social networks can rapidly change in reaction to real-world events (Abel et al. 2011; Huang et al. 2017). Therefore, our work rests on the hypothesis that, as time passes, the set of dominant entity candidates for an ambiguous mention might also evolve depending on real-world events and users' interests. In other words, a limited set of

entity candidates for an ambiguous mention emerges within a specific time period on Twitter and there is no guarantee that this set of entity candidates will remain dominant over time. As a result, our work needs to annotate a tweet based on the dominant entity candidates that are extracted from a Twitter corpus that is related to the same time period as the tweet. Although it implicitly carries the fact that dominant entity candidates of ambiguous mentions can change based on time; however, as a limitation, the performance of our annotation process is affected by the richness of the Twitter corpus and the temporal alignment between the Twitter corpus and tweets that are being annotated.

Therefore, to achieve the best performance of our work, we need to constantly update our training Twitter corpus and repeat our dominant entity candidate detection process to get new dominant entities of ambiguous mentions, which is time-consuming as an offline task. As our future work, we are interested in exploring the evolution of dominant entity candidates for ambiguous mentions. Furthermore, we would also like to study whether it would be possible to find appropriate length of the time intervals (windows) for which dominant entity candidates will be valid to find the optimal update intervals.

## 6 Concluding remarks

In this paper, we have proposed a semantic entity linking method for tweets. Unlike other state-of-the-art techniques that consider all the entity candidates of an ambiguous term from knowledge bases such as Wikipedia, we focus solely on the dominant entity candidates of ambiguous mentions mined from Twitter. To identify dominant candidates, we exploit the latent relation hypothesis whereby context terms for an ambiguous mention are clustered to represent the entity candidates for that term. Once the term clusters for an ambiguous mention is determined, to identify its dominant entity candidates, we map each term cluster onto its corresponding Wikipedia entity. Based on the identified dominant entity candidates for ambiguous mentions, we can annotate tweets to Wikipedia entities. Using a publicly available gold standard dataset, we have been able to show that our method has a competitive performance to other baselines including some recently proposed methods in terms of precision and recall. We also present an evaluation of our dominant entity candidate detection method to show that only a small portion of annotation errors were due to the reduced entity candidate set. In other words, eliminating less frequently observed entities on Twitter does not significantly impact the quality of the annotation results. We further showed that if the tweet under consideration is temporally aligned with the Twitter corpus then our approach shows improved performance compared to other state of the art techniques. In addition, our method has a statistically significant speed up in terms of execution time compared to other baselines.

There are several directions, which we would like to explore in the future. Given the fact that existing techniques such as TagMe and Liu's method consider all Wikipedia entity candidates for an ambiguous mention, and our observation that dominant entity candidates can play a positive role in reducing the candidate space, we are interested in applying the notion of dominant entity candidates to limit the exploration space of these methods and observe the outcome. TagMe's source code is openly available; therefore, our next step would be to modify TagMe to only consider the dominant candidates when performing entity linking on a Tweet as opposed to considering all possible entity candidates from Wikipedia. As another future work, it has been shown in Santamaría et al. (2010) that Wikipedia entities that occur more often in Web search results are also more central to the Wikipedia graph, and are more visited in the Wikipedia web pages. The same might happen on Twitter. We are interested in eliminating the least visited Wikipedia entities and, as a result, the less central Wikipedia entries, and see if focusing on Twitter provides any additional advantages.

## References

Abel F, Gao Q, Houben G-J, Tao K (2011) Analyzing temporal dynamics in twitter profiles for personalized recommendations in the social web. In: Web Science 2011, WebSci '11, Koblenz, Germany—June 15–17, 2011, pp. 2:1–2:8

Abel F, Gao Q, Houben G-J, Tao K (2011) Semantic enrichment of twitter posts for user profile construction on the social web. In: The semanic web: research and applications—8th extended semantic web conference, ESWC 2011, Heraklion, Crete, Greece, May 29–June 2, 2011, proceedings, Part II, pp. 375–389

Arthur D, Vassilvitskii S (2007) k-means++: the advantages of careful seeding. Symp Discret Algorithms SODA 2007 2007:1027–1035

Bhatia S, Jain A (2016) Context sensitive entity linking of search queries in enterprise knowledge graphs. In: International semantic web conference, Springer, New York, pp. 50–54

Blondel VD, Guillaume J-L, Lambiotte R (2008) Lefebvre E (2008) Fast unfolding of communities in large networks. J Stat Mech Theory Exp 10:P10008

Cano BAE, Rizzo G, Varga A, Rowe A, Stankovic M, Dadzie A-S (2014) Making sense of microposts (#microposts2014) named entity extraction & linking challenge. In: Proceedings of the the 4th workshop on making sense of microposts co-located with the 23rd international world wide web conference (WWW 2014), Seoul, Korea, April 7th, 2014, pp. 54–60

Cheng Z, Caverlee J, Lee K (2010) You are where you tweet: a content-based approach to geo-locating twitter users. In: Proceedings of the 19th ACM conference on information and knowledge management, CIKM 2010, pp. 759–768

Cornolti M, Ferragina P, Ciaramita M (2013) A framework for benchmarking entity-annotation systems. In: 22nd international world wide web conference, WWW 2013, pp. 249–260

Cornolti M, Ferragina P, Ciaramita M, Rüd S, Schütze H (2016) A piggyback system for joint entity mention detection and linking in web queries. In: Proceedings of the 25th international conference on world wide web. International World Wide Web Conferences Steering Committee, pp. 567–578

Cucerzan S (2007) Large-scale named entity disambiguation based on Wikipedia data. In: Joint conference on empirical methods in natural language processing and computational natural language learning, pp. 708–716

Cuzzola J, Bagheri E (2014) Derive: finding semantic concepts with property-values from natural language text. In: International conference on computer science and software engineering, CASCON '14, pp. 331–334

Daiber J, Jakob M, Hokamp C, Mendes PN (2013) Improving efficiency and accuracy in multilingual entity extraction. In: I-SEMANTICS 2013—9th international conference on semantic systems, pp. 121–124

Derczynski L, Maynard D, Rizzo G, van Erp M, Gorrell G, Troncy R, Petrak J, Bontcheva K (2015) Analysis of named entity recognition and linking for tweets. Inf Process Manag 51(2):32–49

Dumais ST (2004) Latent semantic analysis. Ann Rev Inf Sci Technol 38(1):188–230

Feng Y, Fani H, Bagheri E, Jovanovic J (2015) Lexical semantic relatedness for twitter analytics. In: International conference on tools with artificial intelligence 2015

Ferragina P, Scaiella U (2010) TAGME: on-the-fly annotation of short text fragments (by Wikipedia entities). In: 19th ACM conference on information and knowledge management, CIKM 2010, pp. 1625–1628

Ferrer M, Valveny E, Serratosa F, Bardají I, Bunke H (2009) Graph-based *k*-means clustering: a comparison of the set median versus the generalized median graph. In: Computer analysis of images and patterns, 13th international conference, CAIP 2009, Münster, Germany, September 2–4, 2009, proceedings, pp. 342–350

Gale WA, Church KW, Yarowsky D (1992) One sense per discourse. In: Proceedings of the workshop on speech and natural language, pp. 233–237

Ganea O-E, Ganea M, Lucchi A, Eickhoff C, Hofmann T (2016) Probabilistic bag-of-hyperlinks model for entity linking. In: Proceedings of the 25th international conference on world wide web. International World Wide Web Conferences Steering Committee, pp. 927–938

Gao N, Cucerzan S (2017) Entity linking to one thousand knowledge bases. In: European conference on information retrieval, Springer, New York, pp. 1–14

Gomaa WH, Fahmy AA (2013) A survey of text similarity approaches. Int J Comput Appl 68(13):13–18

Habib MB, van Keulen M (2012) Unsupervised improvement of named entity extraction in short informal context using disambiguation clues. In: Proceedings of the workshop on semantic web and information extraction (SWAIE 2012), Galway, Ireland, October 9, 2012, pp. 1–10

Habib MB, van Keulen M (2016) Twitterneed: a hybrid approach for named entity extraction and disambiguation for tweet. Nat Lang Eng 22(3):423–456

Han L, Kashyap A, Finin T, Mayfield J, Weese J (2013) Umbc ebiquity-core: semantic textual similarity systems. Proc Second Jt Conf Lex Comput Semant 1:44–52

Han X, Sun L, Zhao J (2011) Collective entity linking in web text: a graph-based method. In: Proceeding of the 34th international ACM SIGIR conference on research and development in information retrieval, pp. 765–774

Hoffart J, Yosef MA, Bordino I, Fürstenau H, Pinkal M, Spaniol M, Taneva B, Thater S, Weikum G (2011) Robust disambiguation of named entities in text. In: Proceedings of the 2011 conference on empirical methods in natural language processing, EMNLP 2011, 27–31 July 2011, John McIntyre Conference Centre, Edinburgh, UK, A meeting of SIGDAT, a Special Interest Group of the ACL, pp. 782–792

Huang J, Peng M, Wang H, Cao J, Gao W, Zhang X (2017) A probabilistic method for emerging topic tracking in microblog stream. World Wide Web 20(2):325–350

Huang H, Cao Y, Huang X, Ji H, Lin C-Y (2014) Collective tweet wikification based on semi-supervised graph regularization. In: Proceedings of the 52nd annual meeting of the association for computational linguistics, pp. 380–390

Inches G, Carman MJ, Crestani F (2010) Statistics of online user-generated short documents. In: Advances in information retrieval, 32nd European conference on IR research, pp. 649–652

Jansen BJ, Zhang M, Sobel K, Chowdury A (2009) Twitter power: tweets as electronic word of mouth. JASIST 60(11):2169–2188

Jovanovic J, Bagheri E, Cuzzola J, Gasevic D, Jeremic Z, Bashash R (2014) Automated semantic tagging of textual content. IT Prof 16(6):38–46

Kapanipathi P, Jain P, Venkatramani C, Sheth AP (2014) User interests identification on twitter using a hierarchical knowledge base. In: The semantic web: trends and challenges—11th international conference, ESWC 2014, Anissaras, Crete, Greece, May 25–29, 2014. proceedings, pp. 99–113

Kapanipathi P, Orlandi F, Sheth AP, Passant A (2011) Personalized filtering of the twitter stream. In: Proceedings of the second workshop on semantic personalized information management: retrieval and recommendation 2011, Bonn, Germany, October 24, 2011, pp. 6–13

Kulkarni S, Singh A, Ramakrishnan G, Chakrabarti S (2009) Collective annotation of Wikipedia entities in web text. In: Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining, pp. 457–466

Lawler GF, Limic V (2010) Random walk: a modern introduction, vol 123. Cambridge University Press, Cambridge

Li Y, Tan S, Sun H, Han J, Roth D, Yan X (2016) Entity disambiguation with linkless knowledge bases. In: Proceedings of the 25th international conference on world wide web, pp. 1261–1270

Li Y, Tan S, Sun H, Han J, Roth D, Yan X (2016) Entity disambiguation with linkless knowledge bases. In: Proceedings of the 25th international conference on world wide web. International World Wide Web Conferences Steering Committee, pp. 1261–1270

Liu X, Li Y, Wu H, Zhou M, Wei F, Lu Y (2013) Entity linking for tweets. In: Proceedings of the 51st annual meeting of the association for computational linguistics, pp. 1304–1311

Mannor S, Menache I, Hoze A, Klein U (2004) Dynamic abstraction in reinforcement learning via clustering. In: Machine learning, proceedings of the twenty-first international conference (ICML 2004), Banff, Alberta, Canada, July 4–8, 2004

Massoudi K, Tsagkias M, de Rijke M, Weerkamp W (2011) Incorporating query expansion and quality indicators in searching microblog posts. In: Advances in information retrieval—33rd European conference on IR research, pp. 362–367

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

Mihalcea R, Csomai A (2007) Wikify!: linking documents to encyclopedic knowledge. In: ACM conference on information and knowledge management, pp. 233–242

Pak A, Paroubek P (2010) Twitter as a corpus for sentiment analysis and opinion mining. In: Proceedings of the international conference on language resources and evaluation, LREC 2010, 17–23 May 2010, Valletta, Malta

Saleiro P, Eduarda MR, Soares C, Oliveira E (2017) Texrep: a text mining framework for online reputation monitoring. New Gener Comput 35(4):365–389

Santamaría C, Gonzalo J, Artiles J (2010) Wikipedia as sense inventory to improve diversity in web search results. In: Proceedings of the 48th annual meeting of the association for computational Linguistics. Association for Computational Linguistics, pp. 1357–1366

Sarmento L, Kehlenbeck A, Oliveira EC, Ungar LH (2009) An approach to web-scale named-entity disambiguation. In: Machine learning and data mining in pattern recognition, 6th international conference, MLDM 2009, Leipzig, Germany, July 23–25, 2009. Proceedings, pp. 689–703

Shen W, Wang J, Luo P, Wang M (2013) Linking named entities in tweets with knowledge base via user interest modeling. In: International conference on knowledge discovery and data mining, KDD 2013, pp. 68–76

Shirakawa M, Wang H, Song Y, Wang Z, Nakayama K, Hara T, Nishio S (2011) Entity disambiguation based on a probabilistic taxonomy. In: technical report MSR-TR-2011-125

Tran AT, Tran NK, Asmelash TH, Jäschke R (2015) Semantic annotation for microblog topics using Wikipedia temporal information. In: Proceedings of the 2015 conference on empirical methods in natural language processing, pp. 97–106

Turney PD (2008) The latent relation mapping engine: algorithm and experiments. J Artif Intell Res (JAIR) 33:615–655

Varga A, Basave AEC, Rowe M, Ciravegna F, He Y (2014) Linked knowledge sources for topic classification of microposts: a semantic graph-based approach. J Web Semant 26:36–57

Vitale D, Ferragina P, Scaiella U (2012) Classification of short texts by deploying topical annotations. In: Advances in information retrieval—34th european conference on IR research, ECIR 2012, Barcelona, Spain, April 1–5, 2012, proceedings, pp. 376–387

Yamada I, Takeda H, Takefuji Y (2015) An end-to-end entity linking approach for tweets. In: Proceedings of the the 5th workshop on making sense of microposts co-located with the 24th international world wide web conference, pp. 55–56

Yosef MA, Hoffart J, Bordino I, Spaniol M, Weikum G (2011) AIDA: an online tool for accurate disambiguation of named entities in text and tables. PVLDB 4(12):1450–1453

Zarrinkalam F, Fani H, Bagheri E, Kahani M, Du W (2015) Semantics-enabled user interest detection from twitter. In: IEEE/WIC/ACM international conference on web intelligence and intelligent agent technology, WI-IAT 2015, pp. 469–476

Zhao G, Wu J, Wang D, Li T (2016) Entity disambiguation to Wikipedia using collective ranking. Inf Process Manag 52(6):1247–1257

Zou X, Sun C, Sun Y, Liu B, Lin L (2014) Linking entities in tweets to Wikipedia knowledge base. In: Natural language processing and Chinese computing—third CCF Conference, pp. 368–378