Soc. Netw. Anal. Min. (2015) 5:59
DOI 10.1007/s13278-015-0298-5

ORIGINAL ARTICLE

Real-time trending topics detection and description from Twitter content

Amina Madani[1] · Omar Boussaid[2] · Djamel Eddine Zegour[3]

Received: 11 March 2015 / Revised: 4 September 2015 / Accepted: 9 September 2015 / Published online: 5 October 2015

Abstract Twitter has become, over the last years, a major source of information. Twitter enables its users to send and read short text-based messages called tweets. Users are busy reporting news about what's going around and within their personal. Numerous researchers from various disciplines have examined Twitter, due to the heterogeneity and immense scale of the data. One of the challenging problems is to automatically identify trending topics in real time on Twitter. Trending topics detection in real time is, thus, of high value to journalists, news reporters, analysts, e-marketing specialists, real-time application developers, and social media researchers to understand what is happening, what emergent trending topics are exchanged between people. In this paper, we propose a new approach that discovers many different trending topics from tweets in real time. Our trending topics are detected for a specific geographic town and compared with the top trending topics shown on Twitter. Contrary to Twitter, our proposed approach distinguishes between different terms corresponding to the same trending topic. We exploit the semantic similarity between keywords composing tweets, by unifying them using a tweets thesaurus former created. Each trending topic has a description presented by keywords of ten tweets that are more representative.

Keywords Tweets mining · Tweets trending topics · Tweets clustering · Real-time trending topics detection · Topical clustering

1 Introduction

These last years have been marked by the emergence of microblogs. Currently Twitter is a popular microblogging service with hundreds of millions of users. They exchange and tell their last thoughts, moods or activities by tweets in some words.

Twitter has several unique characteristics that distinguish it from other microblogs. Important characteristic of Twitter is its real-time nature (Sakaki et al. 2010). A huge amount of tweets is produced several times on Twitter, as soon as there is news. Tweets are data stream that arrive at high speed and in real time.

Two other characteristics are the brevity of tweets and the public nature of the communication (Naaman et al. 2010). Tweets are text-based messages of up to 140 characters that are often available publicly.

Another characteristic is that tweets are usually presented in the form of semi-structured documents (Madani et al. 2014). Figure 1 presents an example of a tweet. While the content is ostensibly 140 unstructured characters, the anatomy of a tweet reveals lots of structural data such as time stamp, the name of the user. Even, the content contains some structural information like RT indicating retweet and #hashtags serving as topical metadata. Moreover, every user has personal information (name, location, biographical sketch).

✉ Amina Madani
a_madani@esi.dz; a_madani@univ-blida.dz

Omar Boussaid
omar.boussaid@univ-lyon2.fr

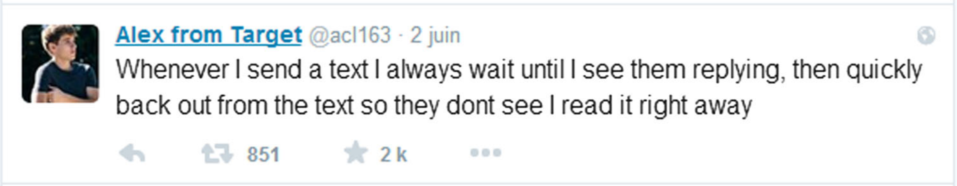
Djamel Eddine Zegour
d_zegour@esi.dz

[1] LRDSI Laboratory, Sciences Faculty Blida 1 University, BP 270, Soumaa Road Blida, Algeria

[2] ERIC Laboratory, Lumiere Lyon 2 University, Lyon, France

[3] LCSI Laboratory, National High School of Computer Science, Algiers, Algeria

Fig. 1 An example of a tweet



Alex from Target @acl163 · 2 juin
Whenever I send a text I always wait until I see them replying, then quickly back out from the text so they dont see I read it right away

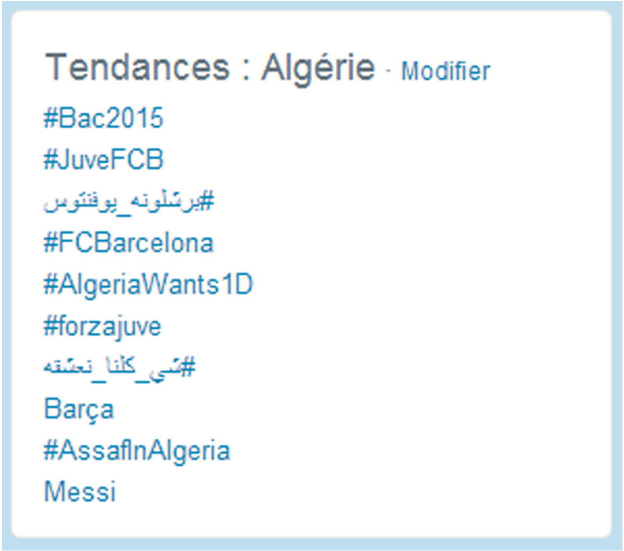
All these characteristics gave rise to an increasing interest in analyzing tweets by the data mining community. In the data mining context, text mining based on analysis of tweets is one of fundamental tasks that can be considered in conjunction with Twitter data (Bifet and Frank 2010; Madani et al. 2014). Analyze tweets to detect trending topics in real time is a big challenge. However, it is not practical for us to browse tweets manually all the time for searching about the latest most discussed issues and thus revealing the emerging topics of our interest. How to automatically understand, extract and summarize useful Twitter content to detect trending topics has therefore become an important and emergent research topic. Trending topics detection is a key step to understand what is exchanged between people on Twitter.

Trending topics detection has primarily involved analyzing the content of tweets (Madani et al. 2014). Twitter defines trending topics as “topics that are immediately popular, rather than topics that have been popular for a while or on a daily basis”[1]. We define a trending topic as an emerging keyword which links to a very recent event (Madani et al. 2014). This keyword is the most used in tweets. A trending topic experience sudden increases in user popularity and can be associated with two dimensions: geographical dimension and temporal dimension.

Trending topics are typically driven by emerging events, breaking news and general topics that attract the attention of a large fraction of Twitter users (Mathioudakis and Koudas 2010).

Twitter has its own trending topics detection mechanism, which is one of the several features of Twitter. Twitter shows a list of top ten most tweeted topics at any given moment on every user’s homepage by default (as can be seen in Fig. 2). These terms reflect the topics that are being discussed most in the latest minutes on the site’s stream of tweets. For viewing more details about these trending topics, we must browse related tweets manually all the time. It is important to automatically analyze, understand, extract and summarize useful tweets content for a given trending topic.

In this paper, we suggest a new approach that detects many different trending topics from tweets in a specific geographic area. Our main objective is to identify different trending topics and their description in real time using topical clustering, specifically focusing on Twitter messages content and using the structural information of tweets as the time and the localization. We create a tweets thesaurus using a bag of words generated from the whole collection of tweets. By considering this thesaurus as a source of synonyms, we manage semantically the keywords of tweets. Trending topics are detected and presented by keywords of few tweets that best describe them.



Tendances : Algérie · Modifier

#Bac2015
#JuveFCB
#برشلونه_يوفنتوس
#FCBarcelona
#AlgeriaWants1D
#forzajuve
#شي_كلنا_نعشقه
Barça
#AssafInAlgeria
Messi

Fig. 2 A screenshot of Twitter’s own trending topics list for Algeria town on 7th June 2015

The remainder of this paper is organized as follows. In the Sect. 2, we present our approach. Experiments are discussed in the Sect. 3. Conclusion and future work are given at the end of the paper.

2 Our approach

We propose a new approach that performs trending topics detection over the Twitter stream. The system identifies new information in trending topics on Twitter content and provides meaningful analytics that synthesize an accurate description of each trending topic.

[1] http://support.twitter.com/articles/101125-abouttrending-topics.

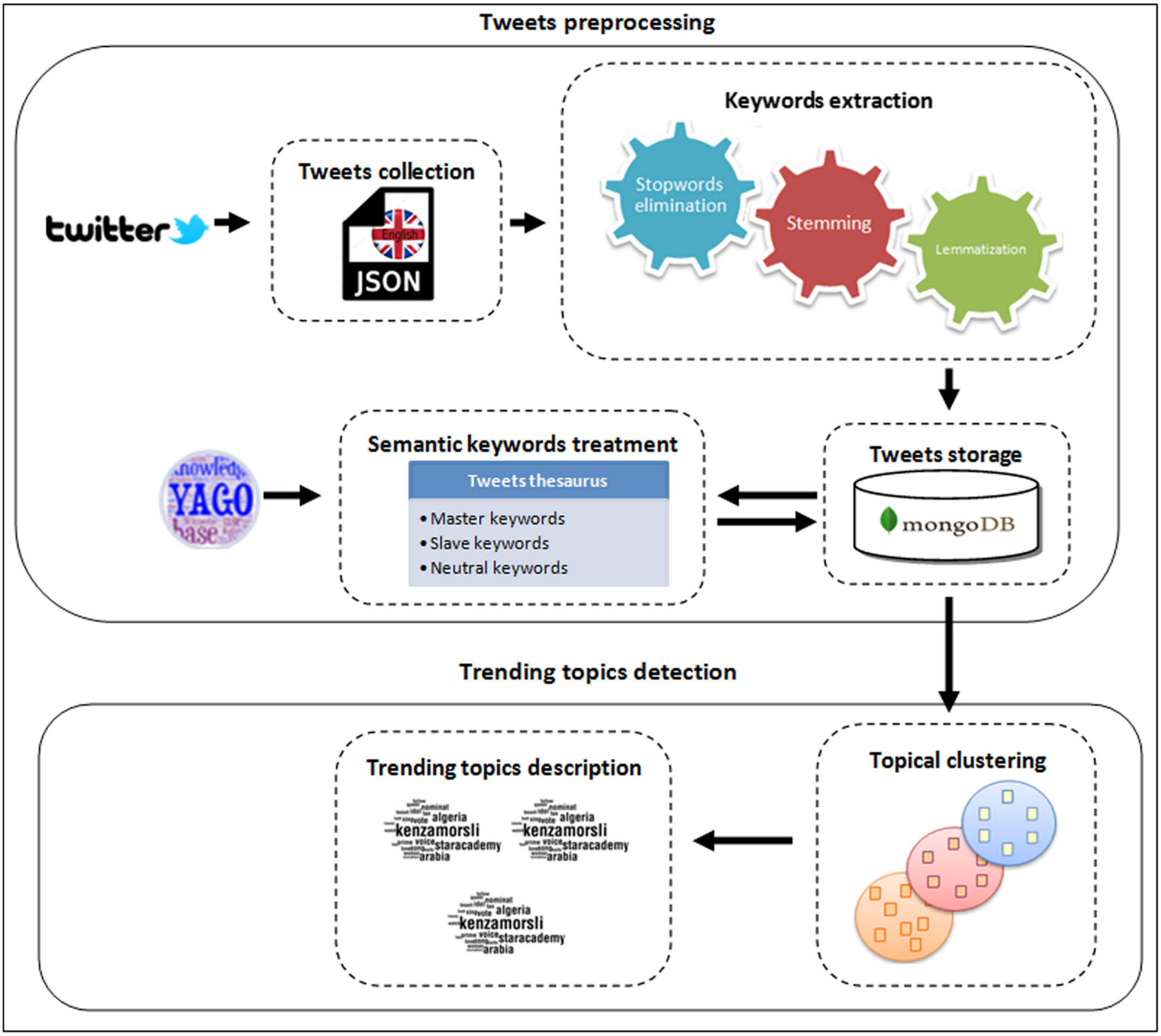


Fig. 3 Global architecture of our approach

Our approach performs trending topics detection in two parts: preprocessing of tweets and trending topics detection (Fig. 3). The important steps of the first part are resumed bellow:

- Collect tweets in JSON format.
- Filter tweets by retrieving only English tweets and eliminating tweets of other language using different stop-words lists appropriated to many languages.
- Extract keywords from tweets using *Natural Language Processing* (NLP). For each tweet:
  - Remove punctuations and special characters.
  - Remove stop-words.
  - Do stemming words.
  - Do lemmatization words.
- Save tweets with their cleaned keywords in mongoDB database.
- Construct a tweets thesaurus containing three types of keywords: *master* keywords, *slave* keywords and *neutral* keywords.
- Semantic keywords treatment by considering the tweet thesaurus as a source of synonyms to unify all keywords of tweets that have semantically the same concept.

The second part of our approach consists of trending topics detection. Their key points are explained below:

- Discover a vector of topics by applying the generative modeling approach of *Hierarchical Dirichlet Processes* on tweets.

- For each tweet a distribution of topics is computed using the vector of topics. The topic that has the highest probability in a distribution of topics is a trending topic. Tweets of the same trending topic are assigned into a same cluster.
- For each cluster:
  - Build a tweets dependency graph where nodes correspond to tweets and edges correspond to a measure of similarity between tweets.
  - Measure the importance of tweets based on TextRank and generate a wordcloud comporting keywords of the ten most important tweets.

We define some challenges that we want our trending topics detection approach to satisfy:

- Volume of tweets: Twitter has users of the order of hundreds of millions. They exchange a huge amount of content. In our approach, we use NoSQL *(Not only Structured Query Language)* database to store and process these voluminous amounts of data.
- Nature of tweets: users on Twitter have space limitation. A tweet can be of maximum 140 characters long. It is short, noisy, with a lot of slang and personal style, and of informal nature. To clean tweets and extract keywords, we use NLP.
- Semantic of tweets: semantic is the study of meaning in language (Hurford 1983). In tweets, semantic treatment (lexical not grammatical) has for goal to study the semantic relationships between words. Hence, the problem is how to distinguish between many different senses that a word may have (polysemy) or between different words that can have the same significance (synonymy) Our approach exploits the semantic similarity of terms composing the textual content of tweets using use external semantic resource.
- Real-time detection: our approach can detect trending topics within minutes of them happening.
- Unsupervised detection: our approach is unsupervised because no training data or user interaction is needed.
- Trending topics description: our goal is to extract keywords of few tweets that best describe a trending topic.
- Structural information: our approach identifies trending topics relying on the content of tweets and using the structural information of tweets as the time and the localization.

The two parts of our approach are described in detail in the following paragraphs.

2.1 Tweets preprocessing

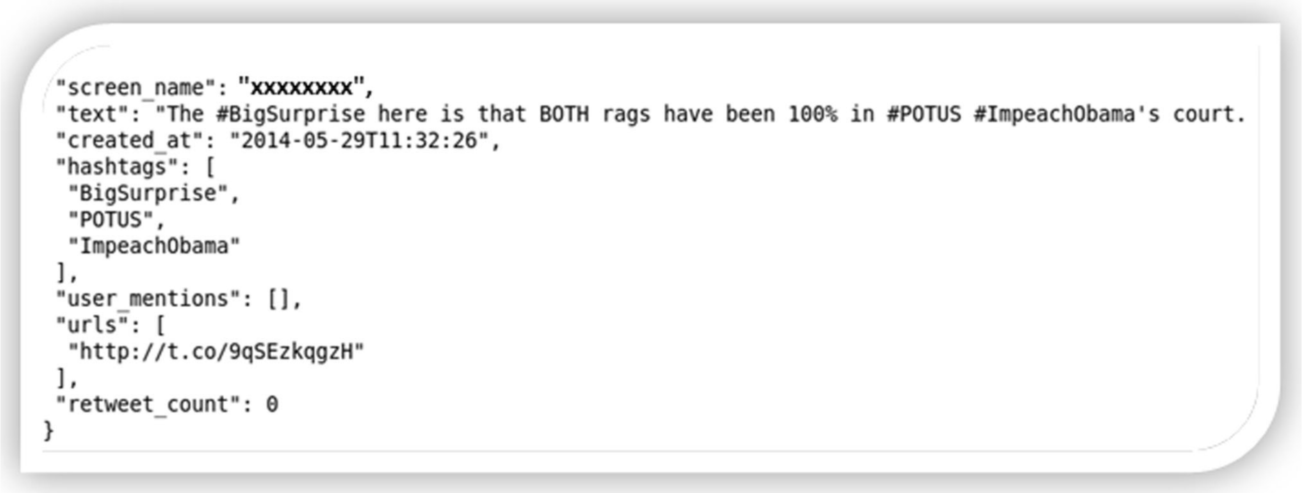
The first part of our approach consists of preprocessing of tweets and it is composed of four steps presented below.

2.1.1 Tweets collection

Our analysis is based on Twitter data, gathered by querying the Twitter search API (*Application Programming Interface*). Twitter's API allows receiving streaming data in different formats.

First, we collect Twitter messages in JSON (*Java Script Objet Notation*) format (Fig. 4).

Next, the language is detected. We filter tweets by retrieving only English tweets and eliminating tweets of other language. For each tweet *t*, we extract its textual content and breaking it up into words to obtain tokens



```
 "screen_name": "xxxxxxxx",
 "text": "The #BigSurprise here is that BOTH rags have been 100% in #POTUS #ImpeachObama's court.
 "created_at": "2014-05-29T11:32:26",
 "hashtags": [
  "BigSurprise",
  "POTUS",
  "ImpeachObama"
 ],
 "user_mentions": [],
 "urls": [
  "http://t.co/9qSEzkqgzH"
 ],
 "retweet_count": 0
}
```

Fig. 4 JSON format of a tweet

composing it. Different stop-words lists appropriated to many languages were used to detect language of tweets. We compare stop-words of the tweet *t* with the stop-words lists to calculate the number of stop-words of the tweet *t* that corresponds to the stop-words of each list. The list having the maximum number of stop-words of the tweet *t* permits to detect the language of this tweet.

Last, we extract six entities from tweets that are: name of Twitter user (screen_name), textual content (text), timestamp (created at), hashtags (#), mentioned urls and mentioned users.

2.1.2 Keywords extraction

This part proposes to clean tweets and extract keywords using NLP. NLP use tools and techniques based on linguistic theories to automate the translation process between computers and human languages. Twitter message presents many intriguing opportunities for applications of NLP.

Keywords are terms that contain most important information. Automatic keyword extraction is the task to identify a small set of words composing tweets. Keywords of a tweet can describe its meaning. A keyword is a string not containing the white-space character.

We obtain tokens composing tweets by removing special characters and punctuations such as point, comma, etc. We use the English stop-words list of NLTK (*Natural Language ToolKit*)[2] to remove stop-words.

Words would all be stemmed and lemmatized to remove noise in tweets collection. We use Porter Algorithm (Porter 1980, 2001) to do stemming words. Stemming transforms the variants of words and reduces them into a single stem by removing suffixes or prefixes. For lemmatization, we use WordNet (Fellbaum 1988)[3] to help map the different word forms to a base form. We need to find the root of verbs and then we convert plural words to their singular form.

2.1.3 Tweets storage

Storing large amounts of tweets is becoming a necessity for scientists in research area. Different solutions have been proposed, among them methods of structuring and storing information like databases. Relational databases quickly reach their limits and add more servers don't increases the performance. Many new technologies have emerged such as NoSQL databases which are based on CAP theorem (*Consistency*, *Availability*, *Partition tolerance*) (Brewer 2000).

NoSQL databases are proposed to manage the continuous growth of data volumes and thus increases the performance and availability of services.

Four classes of NoSQL databases exist, specific to different needs that are key-value stores, column stores, document stores and graph stores. For each class, several implementations exist.

In order to stock tweets in JSON format, we used document stores. In this class, a document in semistructured format is stored with hierarchical tree structure. We chose MongoDB[4] implementation[5] that is major representative for the class of document databases for the following reasons:

- Direct storage from Twitter API: they permit to stock tweets in hierarchical semi-structured format of JSON type.
- Content management: they are used mainly in CMS development (*Content Management System*) while managing larger amounts of data.
- Requests simplicity: requests are syntactically well different from the SQL but they are semantically very similar.

2.1.4 Semantic keywords treatment

The objective of this part is to exploit the semantic similarity of terms composing the textual content of tweets. We are interested to distinguish between different keywords that can have the same significance (synonymy) using external semantic resource.

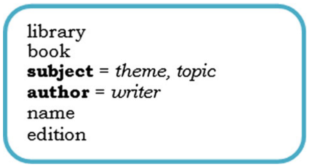
Based on the work of Madani et al. (2011) that proposes to construct a thesaurus from a collection of XML documents, we define a tweets thesaurus. We create a bag of words containing all keywords extracted beforehand (in tweets preprocessing part) of the whole collection of tweets. The bag of words is then processed in order to construct a thesaurus and avoiding redundancy. Each keyword must only appear one time in the bag of words. The tweets thesaurus contains three types of keywords that we define as follows:

[2] http://www.nltk.org.

[3] WordNet is a free lexical resource of English language available on web. It groups terms denoting a given concept (names, verbs, adjectives and adverbs) in sets of synonyms named synsets (Brigitte et al. 2007).

[4] http://www.mongodb.org/.

[5] MongoDB is a schema-free document database written in C++ and developed in an open-source project which is mainly driven by the company 10gen Inc that also offers professional services around MongoDB. According to its developers the main goal of MongoDB is to close the gap between the fast and highly scalable key-value-stores and feature-rich traditional RDBMSs relational database management systems (Strauch 2011).



library
book
subject = *theme, topic*
author = *writer*
name
edition

Fig. 5 Some keywords of tweets thesaurus

- *Master* keyword: the different labels that have the same concept are all included under only one keyword named *master*.
- *Slave* keyword: the synonymous keywords that are included under a *master keyword* are called *slave* keywords.
- *Neutral* keyword: a keyword that is neither not a *master* nor a *slave* is called a *neutral* keyword.

For example, Fig. 5 shows that the two *slave* keywords “theme” and “topic” will be replaced by their *master* keyword “subject”. The *slave* keyword “writer” will be replaced by its *master* keyword “author”. “library”, “book”, “name” and “edition” are *neutral* keywords.

To define the tweets thesaurus (*master* and *slave* keywords), we have used YAGO ontology.[6] YAGO is a large and extensible ontology that builds on entities and relations from Wikipedia. Facts in YAGO have been automatically extracted from Wikipedia and unified with WordNet, using rule-based and heuristic methods (Suchanek et al. 2007).

The tweets thesaurus allows unifying all different keywords existing in tweets that represent semantically the same concept. For each tweet, we replace all *slave* keywords by their *master* keywords using the tweets thesaurus.

Now, we are going to present the second part of our approach that consists of detecting trending topics and their descriptions.

2.2 Trending topics detection

The second part of the approach consists of applying topical clustering on tweets in order to identify trending topics and their description.

2.2.1 Topical clustering

A topic is a distribution (variable relations) that links words in a fixed vocabulary and their occurrence in documents. Topic models are suite of algorithms that automatically discover thematic information (topics) by providing a statistical solution to the problem of managing and analyzing the semantic content of large corpus of documents. Topic models are bag-of-words models exploiting the correlations among the words and latent semantic themes (Blei and Lafferty 2007). Hong and Davison (2010) show that topic modeling is a powerful tool for short text messages as tweets.

Many topic models have been developed by researchers like *Latent Semantic Indexing* (LSI) (Deerwester et al. 1990), *Latent Semantic Analysis* (LSA) (Kubota Ando and LEE 2001) and Probabilistic *Latent Semantic Indexing* (PLSI) or *Probabilistic Latent Semantic Analysis* (PLSA) (Hofmann 1999). The most prominent topic model is *Latent Dirichlet Allocation* (LDA), which was introduced by Blei et al. (2003). LDA becomes one of the hottest research spot in machine learning and information retrieval. It was also introduced in emerging trends detection (Lu and Yang 2012) for summarizing and extracting topics from documents. LDA is a very simple model which is essentially the Bayesian version of PLSA model. LDA is competitor to other topic models in the setting of dimensionality reduction for document collections and other discrete corpora. It is also intended to be illustrative of the way in which probabilistic models can be scaled.

LDA is a probabilistic model for unsupervised clustering of data. The data are documents and the components are distributions of terms that reflect recurring patterns (topics) in the collection. LDA assumes that the ordering of the terms within a document is unimportant (bag of words). LDA postulates a latent structure consisting of a set of topics. Each document is a mixture of various topics and each term is attributable to one of the document’s topics.

Numerous extensions to the standard LDA model exist such as *Hierarchical Dirichlet Processes* (HDP) (Teh et al. 2006), *Dynamic Topic Models* (DTM) (Blei and Lafferty 2006) and *Correlated Topic Models* (CTM) (Blei and Lafferty 2007), etc. We apply the generative modeling approach of *Hierarchical Dirichlet Processes* for topical clustering of tweets. We choose HDP for its nonparametric nature to solve the problem of determining the appropriate parameters.

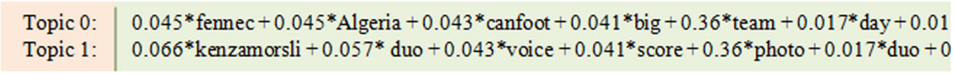
HDP is a nonparametric Bayesian model that can be applied to tweets collections. Tweets are viewed as groups of observed words, mixture components (topics) are distributions over terms and each tweet exhibits the topics with different proportions.

In order to estimate the model parameters, we use Gibbs sampling (Steyvers and Griffiths 2005) inference algorithm, which approximates the joint distribution of multiple variables by drawing a sequence of samples.

The output of HDP is a vector of topics which is a probability distribution over the vocabulary, often visualized as the list of top probability words, as shown in Fig. 6.

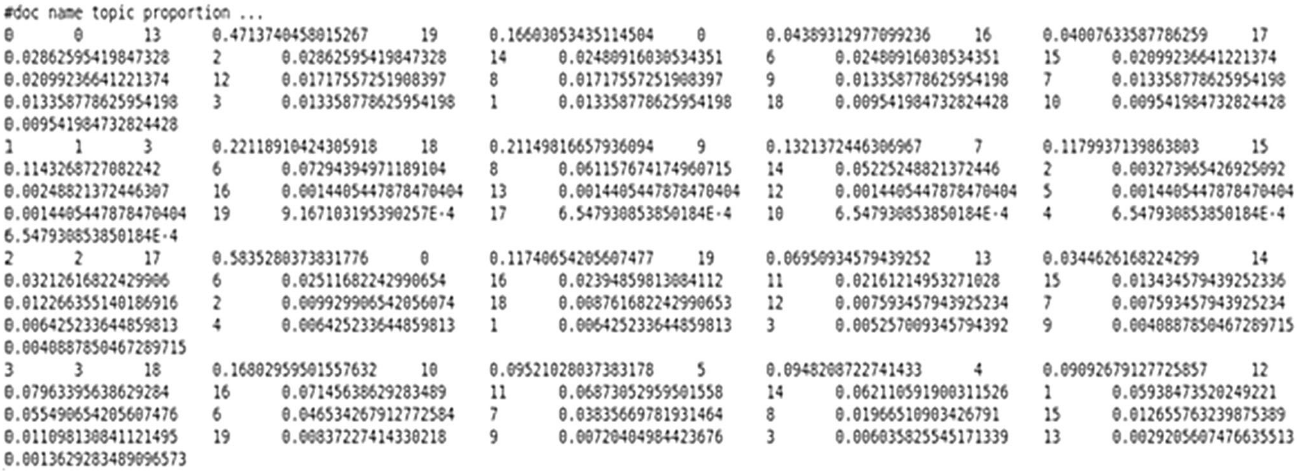
Using the vector of topics and tweets, a distribution of topics is computed for each tweet of the collection. A distribution of topics contains different topics and their

[6] http://www.mpii.mpg.de/~suchanek/yago.



Topic 0:	0.045*fennec + 0.045*Algeria + 0.043*canfoot + 0.041*big + 0.36*team + 0.017*day + 0.01
Topic 1:	0.066*kenzamorsli + 0.057* duo + 0.043*voice + 0.041*score + 0.36*photo + 0.017*duo + 0

Fig. 6 Sample of topics and their words probability distribution



```
#doc name topic proportion ...
0       0       13      0.4713740458015267      19      0.16603053435114504     0       0.04389312977099236     16      0.04007633587786259     17
0.02862595419847328     2       0.02862595419847328     14      0.02480916030534351     6       0.02480916030534351     15      0.02099236641221374
0.02099236641221374     12      0.01717557251908397     8       0.01717557251908397     9       0.013358778625954198    7       0.013358778625954198
0.013358778625954198    3       0.013358778625954198    1       0.013358778625954198    18      0.009541984732824428    10      0.009541984732824428
0.009541984732824428
1       1       3       0.22118910424305918     18      0.21149816657936094     9       0.1321372446306967      7       0.1179937139863803      15
0.11432687270082242     6       0.07294394971189104     8       0.06115767417496715     14      0.05225248821372446     2       0.003273965426925092
0.00248821372446307     16      0.0014405447878470404   13      0.0014405447878470404   12      0.0014405447878470404   5       0.0014405447878470404
0.0014405447878470404   19      9.167103195390257E-4    17      6.547930853850184E-4    10      6.547930853850184E-4    4       6.547930853850184E-4
6.547930853850184E-4
2       2       17      0.5835280373831776      0       0.11740654205607477     19      0.06950934579439252     13      0.0344626168224299      14
0.03212616822429906     6       0.02511682242990654     16      0.02394859813084112     11      0.02161214953271028     15      0.013434579439252336
0.012266355140186916    2       0.009929906542056074    18      0.008761682242990653    12      0.007593457943925234    7       0.007593457943925234
0.006425233644859813    4       0.006425233644859813    1       0.006425233644859813    3       0.005257009345794392    9       0.0040887850467289715
0.0040887850467289715
3       3       18      0.16802959501557632     10      0.09521028037383178     5       0.0948208722741433      4       0.09092679127725857     12
0.07963395638629284     16      0.07145638629283489     11      0.06873052959501558     14      0.062110591900311526    1       0.05938473520249221
0.055490654205607476    6       0.046534267912772584    7       0.03835669781931464     8       0.01966510903426791     15      0.012655763239875389
0.011098130841121495    19      0.00837227414330218     9       0.00720404984423676     3       0.006035825545171339    13      0.0029205607476635513
0.0013629283489096573
```

Fig. 7 Distribution of topics for four tweets

probabilities of words partners for a tweet. In Fig. 7, we show an example of distribution of topics for four tweets.

The topic that has the highest probability in a distribution of topics is a trending topic. Tweets of the same trending topic are assigned into a same cluster.

2.2.2 *Trending topics description*

Twitter force users to click on their trending topics to understand why a topic is trending. Users must read a set of related tweets returned by Twitter. A trending topic description would be extremely helpful for the users to understand what this trending topic is about and why it is trending. A trending topic description consists of the selection of representative tweets for a given trending topic.

In order to measure the importance of tweets in a cluster, we use TextRank (Mihalcea and Tarau 2004). TextRank rely on Google's PageRank (Page et al. 1999; Brin and Page 1998). Google uses the PageRank algorithm to rank web pages in their search results (Page et al. 1999). PageRank is a method for rating web pages by computing a ranking for every web page based on the graph of the web (Page et al. 1999).

TextRank is a graph based ranking model for natural language processing. We use TextRank because it is fully unsupervised, and relies only on the given text to derive an extractive summary, which represents a summarization model closer to what humans are doing when producing an abstract for a given document (Mihalcea and Tarau 2004). We apply TextRank to identify the most important tweets in a cluster, which can be used to build trending topics description.

To apply TextRank, we first need to build a tweets dependency graph TDG = (T,D) associated with each cluster, where: T is the set of nodes that correspond to tweets, D is the set of edges that present a measure of similarity between tweets.

Given two tweets T_i and T_j, with a tweet being represented by a set of keywords, their similarity is defined as:

$$\mathrm{Sim}(T_i, T_j) = \frac{\Sigma_{k \in T_i, T_j}(\mathrm{freq}(k, T_i) + \mathrm{freq}(k, T_j)}{(\log(|T_i|) + \log(|T_j|))} \qquad (1)$$

$\mathrm{freq}(k, T_i)$ is the frequency of keyword k in tweet T_i. $\log(|T_i|)$ is the total number of keywords in tweet T_i.

Based on TextRank, the score of a given node T_i is defined as follows:

$$S(T_i) = \Sigma_{T_j \in \mathrm{nested}(T_i)} \frac{\mathrm{sim}(T_i, T_j)}{\Sigma_{T_k \in \mathrm{nested}(T_j)} \mathrm{sim}(T_k, T_j)} S(T_j) \qquad (2)$$

A trending topic description is constituted of keywords of ten top tweets with the highest score.

In the next section, we will describe the prototype evaluation, the data collection of tweets, and the results of the experiments of our approach.

3 Experiments

To validate our approach, we have developed with the Java language a prototype that assures the following tasks:

- Import a collection of tweets for a specific geographic area chosen by the user.

- Extract keywords using NLP.
- Save tweets with extracted keywords in mongoDB database.
- Construct a tweets thesaurus using YAGO ontology.
- Unify all keywords of tweets using the tweets thesaurus.
- Discover clusters using topical clustering.
- Generate a wordcloud for each cluster comporting keywords of ten representative tweets.

We begin with a description of the data used in our experiments.

The dataset used for our study consists of over 4,140,000 tweets posted by Algeria town users of Twitter between 10th November 2014 and 1st January 2015. We collected the tweets using Twitter's search API (*Application Programming Interface*) method that was requested every 4 h for the most recent tweets posted by Algeria town users.

Using Twitter's top trending topics and search API methods, we collected over 690 trending topics published by Twitter for the Algeria town area between 10th November 2014 and 1st January 2015. The data included the hundred top tweets associated with each published trending topic.

Two different terms identified by Twitter as trending topics can be semantically equivalent. A repetition can be founded in the list of top ten trending topics on Twitter. We remark that different variations in phrasing the same topic create two or more trending topic strings on Twitter. For example, *KenzaWithHaifa*, *#KenzaMorsli*, *#KenzaMorsliArmy*, *#staractowardcandidatkenzamorsli* and *#WelcomeBackkenza* are five terms listed by Twitter as trending topics but they present the same topic.

Contrary to Twitter, our approach distinguishes between different terms corresponding to the same trending topic. In Table 1, we present samples of different trending topics shown on Twitter (presented by different keywords or hashtags) but describe the same topic compared to trending topics generated by our approach.

Our trending topics detected for Algeria town are compared with the top conversations shown as trending topics on Twitter. To verify that our approach detects trending topics in real time, for ten trending topics, we compare their timestamp on Twitter with their timestamp of detection on our system.

From Table 2, we remark that there is a time lag of about few minutes between Twitter and our proposed system. Sometimes, our system detects trending topics before they appear on the homepage of Twitter.

Table 1 Some Twitter's trending topics describing the same topic compared to our approach trending topics

Twitter's trending topics	Our trending topics
#NightChangesVideo	Nightchanges
Night Changes	
#MTVStars	Mtvstars
Justin Bieber	
Little Mix	
#CAN2015	Canfoot
#ALGETH	
#TeamDZ	
Se ne gal	
Afrique du Sud	
Noel	Newyear
Happy New Year	
#KenzaWithHaifa	Kenzamorsli
#staractowardcandidatkenzamorsli	
#KenzaMorsli	
#KenzaMorsliArmy	
#WelcomeBackkenza	
#HaifaInStarac	Staracarabia
#StaracArabia	
#MohammadChahine	
#mohamedhussein	
#LaithAbuJoda	
#Rayan_AbdulRahman	
#LeaMakhoul	
Tunisie	Tunisie
Marzouki	

Table 2 Comparison between timestamp of ten trending topics on Twitter and their timestamp on our system

Trending topic	Timestamp on Twitter	Our timestamp
Renault	11-11-2014/23:01:09	11-11-2014/23:00:36
Maroc	11-11-2014/10:38:05	11-11-2014/10:40:32
Qatar	14-11-2014/06:22:40	14-11-2014/06:27:05
Kenzamorsli	15-11-2014/15:08:49	15-11-2014/15:08:45
Nightchanges	21-11-2014/19:43:39	21-11-2014/19:44:50
Tunisie	24-11-2014/11:09:59	24-11-2014/11:20:02
Newyear	06-12-2014/17:47:13	06-12-2014/17:45:58
Canfoot	10-12-2014/08:56:47	10-12-2014/09:00:10
Arabidol	14-12-2014/21:11:26	14-12-2014/21:10:26
ArabsGoTalent	30-12-2014/13:28:58	30-12-2014/13:31:04

Our experiment was evaluated using *Precision*, *Recall* and *F-measure* scores in comparison to the trending topics identified by Twitter. *Precision* and *Recall* are defined as follows:

$$P = \frac{\Sigma_i a_i}{\Sigma_i a_i + \Sigma_i b_i} \tag{3}$$

Table 3 Precision, Recall, and *F*-measure scores for our trending topics

Month	Day	Without semantic			With semantic		
		R	*P*	*F*1	*R*	*P*	*F*1
November 2014	10	0.34	0.60	0.43	0.90	0.98	0.93
	11	0.47	0.59	0.52	1.00	0.93	0.96
	12	0.62	0.55	0.58	0.93	0.90	0.91
	13	0.70	0.57	0.62	0.91	0.80	0.85
	14	0.27	0.33	0.297	1.00	1.00	1.00
	15	0.41	0.56	0.47	0.82	0.80	0.80
	16	0.19	0.43	0.26	0.80	1.00	0.88
	17	0.50	0.42	0.45	1.00	0.99	0.99
	18	0.47	0.60	0.52	0.82	0.73	0.77
	19	0.23	0.27	0.24	0.88	0.79	0.83
	20	0.38	0.43	0.40	1.00	0.98	0.98
	21	0.22	0.17	0.19	0.86	0.77	0.81
	22	0.44	0.35	0.38	0.90	1.00	0.94
	23	0.28	0.13	0.17	0.94	0.91	0.92
	24	0.33	0.37	0.34	0.93	0.91	0.91
	25	0.17	0.29	0.21	1.00	0.89	0.94
	26	0.20	0.50	0.28	0.90	0.89	0.89
	27	0.49	0.43	0.45	1.00	1.00	1.00
	28	0.51	0.11	0.18	0.95	1.00	0.97
	29	0.31	0.24	0.27	0.97	0.99	0.97
	30	0.19	0.20	0.19	1.00	0.90	0.94
December 2014	1	0.15	0.21	0.17	0.94	1.00	0.96
	2	0.23	0.18	0.20	0.97	1.00	0.98
	3	0.51	0.53	0.51	1.00	1.00	1.00
	4	0.27	0.38	0.31	0.97	0.95	0.95
	5	0.19	0.22	0.20	0.94	0.78	0.85
	6	0.42	0.39	0.40	0.78	0.96	0.86
	7	0.17	0.19	0.17	0.86	0.89	0.87
	8	0.26	0.25	0.25	0.97	0.91	0.93
	9	0.39	0.35	0.36	0.90	0.98	0.93
	10	0.24	0.19	0.21	0.94	0.90	0.91
	11	0.30	0.10	0.15	1.00	0.97	0.98
	12	0.18	0.33	0.23	0.88	1.00	0.93
	13	0.24	0.47	0.31	0.91	0.92	0.91
	14	0.15	0.36	0.21	1.00	1.00	1.00
	15	0.43	0.41	0.41	0.98	0.99	0.98
	16	0.28	0.18	0.21	0.75	0.91	0.82
	17	0.61	0.30	0.40	0.78	1.00	0.87
	18	0.27	0.28	0.27	0.97	0.98	0.97
	19	0.41	0.29	0.33	1.00	0.91	0.95
	20	0.17	0.26	0.20	1.00	0.99	0.99
	21	0.21	0.39	0.27	1.00	1.00	1.00
	22	0.39	0.21	0.27	0.99	0.92	0.95
	23	0.51	0.39	0.44	0.97	0.98	0.97
	24	0.45	0.51	0.47	1.00	1.00	1.00
	25	0.37	0.16	0.22	0.98	1.00	0.98
	26	0.25	0.18	0.20	1.00	1.00	1.00
	27	0.15	0.07	0.09	0.96	0.94	0.94
	28	0.30	0.14	0.19	1.00	0.97	0.98
	29	0.32	0.42	0.36	0.98	0.90	0.93
	30	0.19	0.27	0.22	0.99	1.00	0.99
	31	0.51	0.61	0.55	1.00	0.80	0.88
January 2015	1	0.38	0.43	0.40	1.00	0.97	0.98

$$R = \frac{\Sigma_i a_i}{\Sigma_i a_i + \Sigma_i c_i} \tag{4}$$

For an extracted cluster C_i, we calculate:

a_i as the number of keywords that were identified as trending topics both by Twitter and our method in cluster C_i.

b_i as the number of keywords identified as trending topics by our method in cluster C_i, and that were not identified as trending topics by Twitter.

c_i as the number of keywords identified as trending topics by Twitter that were not identified as trending topics by our method in cluster C_i.

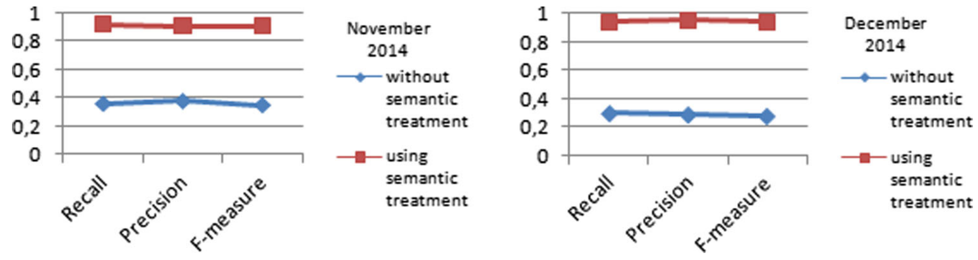
The *F-measure* is the harmonic average of *Recall* and *Precision*, defined as:

$$F1 = \frac{2 \times P \times R}{P + R} \tag{5}$$

The obtained results according to the evaluation measures are summarized in Table 3. We can say that our approach using semantic keywords treatment performs better than without semantic keywords treatment for finding trending topics (Fig. 8). Our results are very interesting; Recall and Precision reached excellent values with the semantic keywords treatment of tweets. We find that exploiting the semantic similarity of keywords composing the textual content of tweets greatly affects the quality of detected trending topics.

Twitter lists a freshly updated set of top terms being discussed currently on Twitter as trending topics. For more details about a trending topic, we must read their related tweets. It is not practical to read several tweets in order to browse information revealing a trending topic. However, it is more important to automatically and quickly understand and summarize useful information related to a trending topic. Using our method, we generate keywords of the ten top tweets related to emergent trending topics. Our approach provides automatically a description of the trending topic and presents their properties by a set of keywords. Keywords of top ten tweets of the trending topic *kenzamorsli* generated by our approach are provided in Fig. 9.

Fig. 8 Average of *Recall*, *Precision* and *F-measure* for our approach without semantic keywords treatment and using semantic keywords treatment on November and December 2014



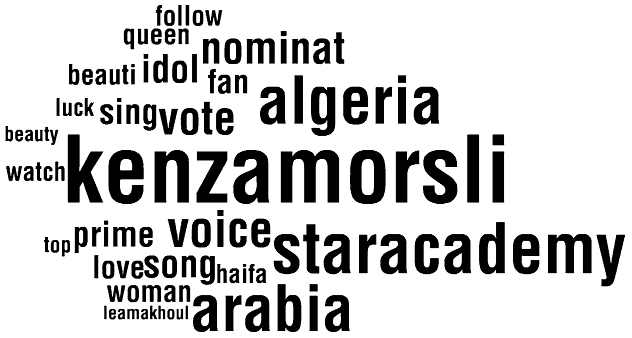


Fig. 9 Keywords of top ten tweets of the trending topic *kenzamorsli*

We compare our approach with the work of Benhardus and Kalita (2013). They use both of the Twitter Streaming API and the Edinburgh Twitter corpus (Petrovic and Lavrenko 2010), a collection of approximately 97 million tweets collected between November 2009 and February 2010. The Edinburgh Twitter corpus was used to provide baseline measurement against the data from the Twitter Streaming API. The corpus was divided into 1212 sections corresponding to one hour's worth of tweets, consisting of two bag-of-words dictionaries for each section, one containing unigrams and one containing bigrams.

To compare our approach with the method of Benhardus and Kalita (2013), we run their first experiment using only unigrams. Their first experiment independently evaluated groups of documents consisting of tweets collected over ten minutes and groups of documents consisting of tweets collected over one hour of streaming. The resulting documents were used independently of one another. For the first experiment, a specified number of baseline documents were used to compute average normalized term Frequency. Their experiment was evaluated using *Recall*, *Precision* and *F-measure* scores in comparison to the trending topics identified by Twitter.

Table 4, Figs. 10 and 11 show the results of comparison between our approach and the approach of Benhardus and Kalita (2013) using Precision, Recall and *F*-measure scores for unigrams of tweets (six 1-h segments and six 10-min segments).

It can be observed that our approach performs better than the other for the two groups of documents (the first collected over 10 min and the second consisting of tweets collected over 1 h of streaming).

Table 4 Comparison of Precision, Recall and *F*-measure scores for hourly unigrams (HU) and 10 minute unigrams (10mU) of tweets

Data set	Benhardus and Kalita (2013)			Our scores		
	R	*P*	*F*1	*R*	*P*	*F*1
HU 1	0.2667	0.2500	0.2581	0.4730	0.5530	0.5099
HU 2	0.3000	0.3103	0.3051	0.7803	0.6137	0.6870
HU 3	0.1333	0.1290	0.1311	0.6450	0.4108	0.5019
HU 4	0.2333	0.2188	0.2258	0.8678	0.9762	0.9188
HU 5	0.2000	0.2143	0.2069	0.4312	0.6166	0.5075
HU 6	0.1667	0.1786	0.1724	0.8398	0.7655	0.8009
Average	0.2167	0.2168	0.2166	0.6728	0.6559	0.6643
10mU 1	0.4333	0.4063	0.4194	0.7589	0.6912	0.7234
10mU 2	0.2667	0.2424	0.2540	1	0.9054	0.9503
10mU 3	0.4000	0.3871	0.3934	0.5908	0.5611	0.5756
10mU 4	0.3667	0.3333	0.3492	0.8593	0.6377	0.7321
10mU 5	0.2667	0.2500	0.2581	0.9537	0.6504	0.7734
10mU 6	0.3000	0.2813	0.2903	0.8750	0.5667	0.6879
Average	0.3389	0.3167	0.3274	0.8396	0.6687	0.7445

4 Related work

Our work is related to automatic real-time detection of trending topics from tweets content. Determining trending topics can be considered a subset of the larger problem known as *Topic Detection and tracking* (TDT) (Allan et al. 2000, 1998; Brants et al. 2003; Kontostathis et al. 2003; Mei and Zhai 2005; Wang et al. 2007; Yang et al. 1998, 2002). TDT is a large problem and has long been a research topic. The real-time trending topic detection problem is closely related to that of stream clustering and topic modeling (Aggarwal 2006; Blei and Lafferty 2006; He et al. 2007; Liu et al. 2008; Surendran and Sra 2006; Wang et al. 2007; Zhong 2005; Wartena and Brussee 2008) found that topic identification by clustering a set of keywords works fairly well. They present an evaluation of topic detection on a Wikipedia corpus.

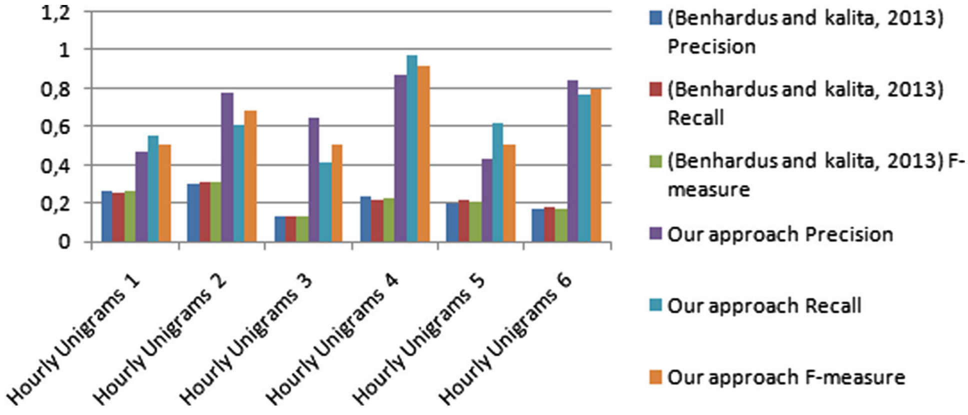


Fig. 10 Comparison of performance for the first group of unigrams of tweets

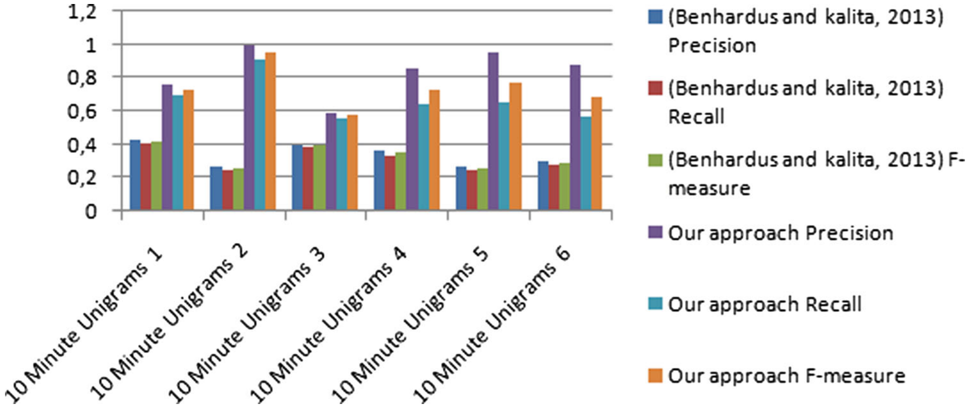


Fig. 11 Comparison of performance for the second group of unigrams of tweets

Our approach attempts to detect trending topics in the text stream, especially in tweets. Twitter serves more as an information spreading medium than an online social networking service (Kwak et al. 2010). Kwak et al. (2010) analyze the tweets of top trending topics and report on the temporal behavior of trending topics and user participation. They then classify the trending topics based on the active period and the tweets and show that the majority of topics are headline or persistent news in nature.

Several studies have been focused for trending topics detection in social media in general and on Twitter in particular. Sankaranarayanan et al. (2009) use clustering methods to identify trending topics that correspond to news events and their associated messages on Twitter. They use noisy data along with a nave Bayesian classifier to improve the quality of the noisy data by throwing away a large portion of the tweets noise. In our case, we use NLP tools to clean tweets and extract keywords.

Cataldi et al. (2010) propose a novel approach to detect in real-time emerging topics on Twitter. They extract the contents (set of terms) of the tweets and model the term life cycle according to a novel aging theory intended to mine terms that frequently occur in the specified time interval and they are relatively rare in the past. Moreover, considering that the importance of content also depends on its source, they studied the social relationships in the user network in order to determine the authority of the users. Finally, they formalized a keyword-based topic graph which connects the emerging terms with their other semantically related keywords, in order to obtain a set of emerging topics under user-specified time constraints.

Mathioudakis and Koudas (2010) propose TwitterMonitor, a system that identifies emerging topics on Twitter in real time. TwitterMonitor provides meaningful analytics that synthesize an accurate description of each topic. It extracts additional information from the tweets that belong to the trend, aiming to discover interesting aspects of it. Users interact with the system by ordering the identified trends using different criteria and submitting their own description for each trend. Contrary to TwitterMonitor, our approach generates automatically trending topics description that is presented by keywords of ten representative tweets.

Naaman et al. (2010) develop a taxonomy of the trends present in a large dataset of Twitter messages from one geographic area. They identify important dimensions according to which trends can be categorized, as well as the key distinguishing features of trends that can be derived from their associated messages. They quantitatively examine the computed features for different categories of trends, and establish that significant differences can be detected across categories. Our trending topics are detected for a specific geographic area chosen by the user. A tweets thesaurus is former created from the whole collection of tweets to manage semantically keywords.

Budak et al. (2011) introduce new methods for identification of important topics that utilize the network topology. They propose two novel trend definitions called coordinated and uncoordinated trends that detect topics that are popular among highly clustered and distributed users, respectively. A novel information diffusion model called *Independent Trend Formation Model* (ITFM) has also been introduced to distinguish viral diffusion of information from diffusion through external entities, such as news media, and to capture the diffusion of an arbitrary number of topics in a social network.

Benhardus and Kalita (2013) outlines methodologies of detecting and identifying trending topics from streaming data. *Term Frequency-Inverse Document Frequency* (TF-IDF) analysis and relative normalized term frequency analysis are performed on the tweets to identify the trending topics. Relative normalized term frequency analysis identifies unigrams, bigrams, and trigrams as trending topics, while term frequency-inverse document frequency analysis identifies unigrams as trending topics. TF-IDF does not make much sense for short text analysis. Our approach is based on Topic models such as LDA because it might work even better.

Zubiaga et al. (2013) propose a method that provides an efficient way to immediately and accurately categorize trending topics without need of external data, enabling news organizations to track and discover breaking news in real time, or quickly identify viral memes that might enrich marketing decisions, among others. The analysis of social features as observed in social trends also reveals social patterns associated with each type of trend, such as tweets related to ongoing events being shorter as many of the tweets were likely sent from mobile devices, or memes having more retweets originating from fewer users than for other kinds of trends.

Our work differs from the above approaches by managing the continuous growth of tweets and storing them in NoSQL databases. In contrast to the above approaches, we propose a description for each emergent trending topic represented by keywords of ten tweets. Our approach create a tweets thesaurus and use it to manage semantically keywords of tweets which resolves the problem of repetition of words founded in Twitters own trending topics. We focus here on tweets that are produced and shared within a specific geographic town and trending topics detected in that content.

5 Conclusion

Tweets are being posted with vast amount of new information and changes continuously. They are a live stream that contains a great wealth of information where topics of discussion shift dynamically with time. Twitter content is a good resource for detecting trending topics.

Our goal is to propose a novel detection approach that permits to retrieve in real time the most emergent trending topics expressed by tweets. Our method is independent of the method used by Twitter.

Our approach presents the following advantages: (i) it identifies trending topics in real time relying on the content of tweets and using the structural information of tweets as the time and the localization, (ii) to manage the continuous growth of tweets, it stores them in NoSQL databases. (iii) it provides automatically a description of each trending topic and present it in a wordcloud, contrary to Twitters own trending topics that we must search their related tweets to view more details, (iv) using external semantic resource, it create a tweets thesaurus and use it to manage semantically keywords of tweets which resolves the problem of repetition of words founded in Twitters own trending topics.

The experiments provide evidence that the proposed approach could result in better describing trending topics of tweets than Twitter's trending topics.

Based on the initial performance of the proposed approach, there are several possible extensions and improvements for it. One potential extension would be to find trending topics using tweets in different languages not only in English.

However, it will be interesting to apply other methods of data mining. Among these methods, we mention the supervised classification, while generating the pattern of the common keywords of tweets for each pre-definite cluster and applying the classification of tweets according to these patterns.

References

Aggarwal CC (2006) Data streams: models and algorithms (advances in database systems). Springer-Verlag Inc, New York

Allan J, Papka R, Lavrenko V (1998) On-line new event detection and tracking. In: Proceedings of the 21st annual international ACM SIGIR conference on research and development in information retrieval, ACM, SIGIR '98, pp 37–45

Allan J, Lavrenko V, Jin H (2000) First story detection in tdt is hard. In: Proceedings of the 9th international conference on information and knowledge management, ACM, CIKM '00, pp 374–381

Benhardus J, Kalita J (2013) Streaming trend detection in twitter. Int J Web Based Communities 9(1):122–139

Bifet A, Frank E (2010) Sentiment knowledge discovery in twitter streaming data. In: Proceedings of the 13th international conference on discovery science, Springer-Verlag, DS'10, pp 1–15

Blei D, Lafferty J (2007) A correlated topic model of science. Ann Appl Stat 1(1):17–35

Blei D, Ng AY, Jordan MI (2003) Latent dirichlet allocation. J Mach Learn Res 3:993–1022

Blei DM, Lafferty JD (2006) Dynamic topic models. In: Proceedings of the 23rd international conference on machine learning, ACM, ICML '06, pp 113–120

Brants T, Chen F, Farahat A (2003) A system for new event detection. In: Proceedings of the 26th annual international ACM SIGIR conference on research and development in informaion retrieval, ACM, SIGIR '03, pp 330–337

Brewer EA (2000) Towards robust distributed systems (abstract). In: Proceedings of the 19th annual ACM symposium on principles of distributed computing, ACM, PODC '00, pp 7–19

Brigitte S, Chantal R, Francois-Elie C (2007) Techniques d'alignement d'ontologies bases sur la structure d'une ressource complementaire. In: 1eres Journees Francophones sur les Ontologies, JFO 2007

Brin S, Page L (1998) The anatomy of a large-scale hypertextual web search engine. Comput Netw ISDN Syst 30(1–7):107–117

Budak C, Agrawal D, El Abbadi A (2011) Structural trend analysis for online social networks. Proc VLDB Endow 4(10):646–656

Cataldi M, Di Caro L, Schifanella C (2010) Emerging topic detection on twitter based on temporal and social terms evaluation. In: Proceedings of the 10th international workshop on multimedia data mining, ACM, MDMKDD '10, pp 4:1–4:10

Deerwester S, Dumais ST, Furnas GW, Landauer TK, Harshman R (1990) Indexing by latent semantic analysis. J Am Soc Inf Sci 41(6):391–407

Fellbaum C (1988) WordNet : an electronic lexical database. MIT Press, Cambridge

He Q, Chang K, Lim EP, Zhang J (2007) Bursty feature representation for clustering text streams. In: SDM conference, pp 491–496

Hofmann T (1999) Probabilistic latent semantic indexing. In: Proceedings of the 22nd annual international ACM SIGIR conference on research and development in information retrieval, ACM, SIGIR '99, pp 50–57

Hong L, Davison BD (2010) Empirical study of topic modeling in twitter. In: Proceedings of the 1st workshop on social media analytics, ACM, SOMA '10, pp 80–88

Hurford JR (1983) Semantics: a coursebook. Cambridge University Press, Cambridge

Kontostathis A, Galitsky L, Pottenger W, Roy S, Phelps D (2003) A survey of emerging trend detection in textual data mining. In: Berry MW (ed) Survey of text mining. Springer, New York, pp 185–224

Kubota Ando R, Lee L (2001) Iterative residual rescaling: an analysis and generalization of lsi. In: Proceedings of SIGIR, New Orleans, pp 154–162

Kwak H, Lee C, Park H, Moon S (2010) What is twitter, a social network or a news media? In: Proceedings of the 19th international conference on World Wide Web, ACM, WWW '10, pp 591–600

Liu Y, Cai JR, Yin J, Fu AC (2008) Clustering text data streams. JCST 32:112–128

Lu R, Yang Q (2012) Trend analysis of news topics on twitter. Int J Mach Learn Comput 2

Madani A, Boussaid O, Zegour DE (2011) Clust-xpaths: clustering of xml paths. In: Proceedings of the 7th international conference on machine learning and data mining in pattern recognition. Springer-Verlag, MLDM'11, pp 294–305

Madani A, Boussaid O, Zegour DE (2014) Whats happening: a survey of tweets event detection. In: Proceedings of the 3rd international conference on communications, computation, networks and technologies, INNOV 2014, pp 16–22

Mathioudakis M, Koudas N (2010) Twittermonitor: trend detection over the twitter stream. In: Proceedings of the 2010 ACM SIGMOD international conference on management of data, ACM, SIGMOD '10, pp 1155–1158

Mei Q, Zhai CX (2005) Discovering evolutionary theme patterns from text—an exploration of temporal text mining. In: KDD conference, Chicago, pp 198–207

Mihalcea R, Tarau P (2004) Textrank: bringing order into texts. In: Proceedings of empirical methods for natural language processing, pp 404– 411

Naaman M, Boase J, Lai CH (2010) Is it really about me?: message content in social awareness streams. In: Proceedings of the 2010 ACM conference on computer supported cooperative work, ACM, CSCW '10, pp 189–192

Page L, Brin S, Motwani R, Winograd T (1999) The pagerank citation ranking: bringing order to the web. Technical report 1999-66, Stanford InfoLab

Petrovic OM S, Lavrenko V (2010) The Edinburgh twitter corpus. In: Proceedings of NAACL workshop on social media

Porter MF (1980) An algorithm for suffix stripping. Program 14(3):130–137

Porter MF (2001) Snowball: a language for stemming algorithms. Published online. http://snowball.tartarus.org/texts/introduction.html

Sakaki T, Okazaki M, Matsuo Y (2010) Earthquake shakes twitter users: real-time event detection by social sensors. In: Proceedings of the 19th international conference on World Wide Web, ACM, WWW '10, pp 851–860

Sankaranarayanan J, Samet H, Teitler BE, Lieberman MD, Sperling J (2009) Twitterstand: news in tweets. In: Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems, ACM, GIS '09, pp 42–51

Steyvers M, Griffiths T (2005) Probabilistic topic models. In: Landauer T, Mcnamara D, Dennis S, Kintsch W (eds) Latent semantic analysis: a road to meaning. Laurence Erlbaum

Strauch C (2011) Nosql databases. Lecture selected topics on software-technology ultra-large scale sites. Manuscript, Stuttgart Media University. http://www.christof-strauch.de/nosqldbs.pdf

Suchanek FM, Kasneci G, Weikum G (2007) Yago: a core of semantic knowledge. In: Proceedings of the 16th international conference on World Wide Web, ACM, WWW '07, pp 697–706

Surendran AC, Sra S (2006) Incremental aspect models for mining document streams. In: Proceedings of the 10th European conference on principle and practice of knowledge discovery in databases, Springer-Verlag, PKDD'06, pp 633–640

Teh Y, Jordan M, Beal M, Blei D (2006) Hierarchical dirichlet processes. J Am Stati Assoc 101:1566–1581

Wang X, Zhai C, Hu X, Sproat R (2007) Mining correlated bursty topic patterns from coordinated text streams. In: Proceedings of the 13th ACM SIGKDD international conference on knowledge discovery and data mining, ACM, KDD '07, pp 784–793

Wartena C, Brussee R (2008) Topic detection by clustering keywords. In: Proceedings of the 2008 19th international conference on database and expert systems application, IEEE Computer Society, DEXA '08, pp 54–58

Yang Y, Pierce T, Carbonell J (1998) A study of retrospective and online event detection. In: Proceedings of the 21st annual international ACM SIGIR conference on research and development in information retrieval, ACM, SIGIR '98, pp 28–36

Yang Y, Zhang J, Carbonell J, Jin C (2002) Topic-conditioned novelty detection. In: Proceedings of the 8th ACM SIGKDD international conference on knowledge discovery and data mining, ACM, KDD '02, pp 688–693

Zhong S (2005) 2005 special issue: efficient streaming text clustering. Neural Netw 18(5–6):790–798

Zubiaga A, Spina D, Fresno V, Martínez R (2013) Real-time classification of twitter trends. J Am Soc Inf Sci Technol (JASIST) 66(3):462–473