Tweet-SCAN: An event discovery technique for geo-located tweets

Joan CAPDEVILA [a,b], Jesús CERQUIDES [c], Jordi NIN [a,b], Jordi TORRES [a,b]
[a] *Department of Computer Architecture, Universitat Politècnica de Catalunya (UPC)*
[b] *Barcelona Supercomputing Center (BSC-CNS)*
[c] *Institut d'Investigació en Intel·ligència Artificial (IIIA-CSIC)*

Abstract. Twitter has become one of the most popular Location-Based Social Networks (LBSNs) that enables bridging physical and virtual worlds. Tweets, 140-character-long messages published in Twitter, are aimed to provide basic responses to the *What's happening?* question. Occurrences and events in the real life are usually reported through geo-located tweets by users on site. Uncovering event-related tweets from the rest is a challenging problem that necessarily requires exploiting different tweet features. With that in mind, we propose Tweet-SCAN, a novel event discovery technique based on the density-based clustering algorithm called DBSCAN. Tweet-SCAN takes into account four main features from a tweet, namely content, time, location and user to cluster homogeneously event-related tweets. This new technique models textual content through a probabilistic topic model called Hierarchical Dirichlet Process and introduces Jensen-Shannon distance for the task of neighborhood identification in the textual dimension. As a matter of fact, we show Tweet-SCAN performance in a real data set of geo-located tweets posted during Barcelona local festivities in 2014, for which some of the events were known beforehand. By means of this data set, we are able to assess Tweet-SCAN capabilities to discover events, justify using a textual component and highlight the effects of several parameters.

Keywords. Twitter, Unsupervised learning, event discovery, DBSCAN, probabilistic topic models, Hierarchical Dirichlet Process (HDP), Jensen-Shannon Distance (JSD)

1. Introduction

Twitter[1] is one of the most popular social networks and microblogging sites offering location-based services to identify the geographical location of social content, e.g. tweets. A tweet is a 140-character-long status message that responds to the question *What's happening?*. This update message is associated to a user, a posting time and might contain some sort of geographical localization. In fact, several studies have already shown that one out of five tweets is geo-located or its location can be inferred from user metadata. Given that 500 millions tweets are generated per day[2], understanding some of the physical world behaviors from geo-located tweets seems feasible. There are numer-

[1] http://www.twitter.com/
[2] https://blog.twitter.com/2013/new-tweets-per-second-record-and-how

ous research papers supporting this viewpoint in various fields ranging from politics [1] to epidemics [2] or seismology [3].

Consequently, we can also view Twitter as a rich source of data about the physical world generated by millions of distributed users acting as sensors that report what is happening right now. An event happening in the real world (such as a demonstration, a music concert, an accident or a street fight) will be likely reported on Twitter by means of geo-located tweets posted by a subset of users close to the event location. These events provide an explanation for a large subset of tweets, and thus its identification from the tweet stream provides good insights about what is going on. Nonetheless, these events are usually masked by tweets which do not contribute to any particular pattern and can be considered as noise for the event identification task. Therefore, the problem of event discovery in Location-based Social Networks (LBSNs), and specifically in Twitter, consists in uncovering and determining these events while excluding the undesired observations [4].

To tackle event discovery in Twitter, we framed the problem in a clustering paradigm in which clusters could be arbitrarily shaped and not necessarily all points have to be associated to a cluster but to noise. DBSCAN (Density-Based Spatial Clustering of Applications with Noise) [5], seems suitable for this problem, but not all tweet features can be directly mapped to the metric space defined by this algorithm. Therefore, we introduce Tweet-SCAN, a novel event discovery technique which adapts the DBSCAN algorithm to cope with Twitter objects, considering its spatial, temporal, textual and user dimensions. Tweet-SCAN implements independent neighborhood identification in each separate dimension and groups common neighbors into a cluster which is finally associated to an event. Previous works [6] have extended DBSCAN to deal with spatio-temporal dimensions. However, this is the first time that the algorithm is extended to deal also with textual information.

The textual part of a tweet is modeled through a probabilistic topic model [7], named Hierarchical Dirichlet Process (HDP) [8], which can be seen as the nonparametric extension of Latent Dirichlet Allocation (LDA) [9]. This nonparametric topic model represents the textual dimension of each tweet as a probability distribution over topics. To assess the similarity of the textual part of two tweets, we propose the usage of the Jensen-Shannon distance [10] between their topic distributions.

To our best knowledge, this is the first attempt to model spatial, temporal, textual and user dimensions for event discovery problems in LBSNs. Despite authors in [11] have presented a spatio-temporal topic model that uses all preceding features, their problem settings are different from ours since they seek to model individual users' mobility behaviors from them. Others in [12] have proposed a hierarchical probabilistic model which lacks of user dimension and requires transfer learning from external data sets. Despite not being formulated in probabilistic terms, our approach does not require other source of data rather than Twitter and it benefits from taking into account user dimension.

The algorithm capabilities for uncovering events are assessed in a real data set composed of geo-located tweets from Barcelona during its local festivities in September of 2014. This data set has been crawled from the Twitter Streaming API via a distributed system called Hermes [13]. Furthermore, some tweets have been manually tagged and assigned to an event based on our expert knowledge about the local festivities. This tagging process allows to quantitatively evaluate the algorithm and interpret the algorithm parameters.

Authors in [14] have also used Barcelona city as a test bed for their event detection technique surely due to the high penetration of social networks and the vast number of events in the city. They presented a model for the normal city behavior to characterize abnormal situation or events. Then, they applied this model into a data set composed of aggregated observations from Twitter, Instagram and Foursquare. The diversity of using posts from different social networks shows promising results. Our work does not account for social network diversity to boost performance, but we focus on improving the tweets modeling instead.

The rest of the paper is organized as follows: First, Tweet-SCAN technique is described in deep detail in Section 2. In Section 3, we perform a descriptive analysis of the Twitter data set. Then, we assess Tweet-SCAN discovering capabilities by studying different parameter settings, see Section 4. To conclude, we summarize the major contributions of this work and identify next challenges in Section 5.

2. Tweet-SCAN technique

Tweet-SCAN technique can be viewed as a two step process with first a pre-processing and modeling step and later a clustering algorithm sub-process.

2.1. Tweet-SCAN pre-processing and modeling

Tweets considered in this work are composed of a text message written by a user who has posted at a given time and tagged it with its geographical location. They might also contain *hashtags* or mentions indicating trending topics or users at which the tweet refers to, respectively. However, both are part of the 140-character-long message and to simplify we will consider them as plain text from now on.

The **posting time** is real-valued scalar feature which can be measured with distinct units (sec, min, days) with respect to a time reference. In our case, we will be using the oldest date in the data set as reference and time will be a real-valued feature measured in seconds.

The **geographical location** is a real-valued bi-dimensional vector that encodes latitude and longitude coordinates. These two variables are encoded in a given Coordinate Reference System (CRS) which determines how distances are computed in this metric space.

The **user** is represented by an integer value that uniquely identifies the user in the whole social network. Users will be considered here as cluster's attributes instead of user's features. As we will describe later, clusters are rejected if they do not contain a minimum number of unique users.

The **text message** is 140-character-long free-text which is too high dimensional to simply model each character/word in it. Because of this, we propose here to use a probabilistic topic model [7], which reduces the space dimension while still considering word co-occurrences. Moreover, text messages can now be expressed as probability distribution over topics and a distance metric can be defined over this lower dimension space.

We rely on a state-of-the-art nonparametric Bayesian model called Hierarhical Dirichlet Process (HDP) [8]. The HDP model overcomes the limitation of its parametric counterpart, Latent Dirichlet Allocation (LDA) [9], by using Dirichlet Process instead

of Dirichlet Distributions. The graphical representation of HDP is shown in Fig. 1. This topic model consist of two nested Dirichlet Process (DP): G_o, whose base distribution is H and γ is its concentration parameter, and G_i, whose base distribution is drawn from G_o and α_o is its concentration parameter. Although the number of mixture components, K , does not need to be estimated, the hyperparameters γ and α_o might rule the number of components. Because of this, priors over these parameters are often considered to avoid performing model selection. Typically, vague informative gamma priors are chosen: $\gamma \sim\ Gamma(1, 0.1)$ and $\alpha_o \sim\ Gamma(1, 1)$

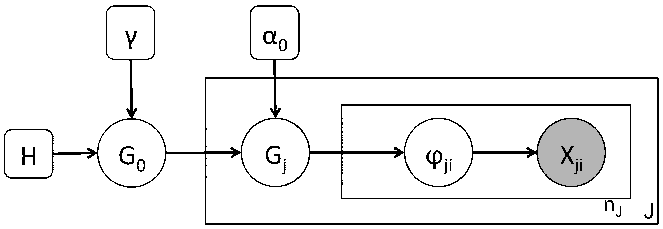


Figure 1. Hierarchical Dirichlet Process (HDP) graphical model.

The n_J observed words, $x_{j,i}$, for each of the J tweets that feed the HDP model are firstly pre-processed to reduce noise. Classical data cleaning techniques from Natural Language Processing (NLP) are applied to the whole corpus: lowering case, removing numbers, characters and stripping white-spaces. Secondly, training documents are built from tweets aggregated by *hashtag* and these are used to train the HDP topic model. This aggregation strategy was already proposed in [15] to address some issues of short text modeling with classical topic models (e.g. LDA or HDP). Last, the trained HDP model is employed to infer the topic mixture distributions per each single tweet. This topic distributions are vectors of length K, which each k-component represents the probability that the tweet belongs to topic k.

2.2. Tweet-SCAN clustering algorithm

The clustering algorithm presented here is based on the well-known DBSCAN algorithm [5]. The basic elements composing DBSCAN can be found in [5] with all details. Therefore, we reference the reader to this article to consult the formal definition of algorithms components such as *core object*, *border object*, *noise* or *density-reachable object*.

Algorithm 1 basically starts with the first document d in the database $D_{1:N}$ and retrieves all neighboring tweets. The neighborhood identification, lines 4 and 14, will be treated in further detail later since each dimension precises a different distance metric. If the neighborhood size is greater than parameter $Minpts$ and the user diversity of this subset is greater than parameter u, d is considered a *core object* and a new temporal cluster is created. Otherwise, document d is tagged as *noise*. Given that user diversity is a ratio ranging from 0 to 1, user diversity could be completely disabled by simply setting u to 0.

Then, for each neighbor in the cluster, we search for its neighboring tweets and check again whether the neighborhood size is larger than $Minpts$. Objects in this new neighborhood are *density-reachable objects* from the original point. They can still be *core objects*, if their ϵ-neighborhood has at least $Minpts$ objects, or simply *border objects*, if

Algorithm 1: Tweet-SCAN algorithm.

input : $\{D_{1:N}, \epsilon_1, \epsilon_2, \epsilon_3, Minpts, u\}$
output: $\{CLabels\}$

```
1  CLabel = 0;
2  for d ← D_{1:N} do
3      if d ∉ Cluster then
4          Neigh = FindNeighbors(d, D_{1:N}, ε1, ε2, ε3);
5          if |Neigh| < Minpts or diversity < u then
6              Mark d as Noise;
7          else
8              Mark d with CLabel;
9              Cand.Push(Neigh)
10             while |Cand| > 0 do
11                 o = Cand.Pop();
12                 if o ∉ Cluster then
13                     Mark o with CLabel;
14                     NewNeigh = FindNeighbors(o, D_{1:N}, ε1, ε2, ε3);
15                     if |NewNeigh| ≥ Minpts and diversity ≥ u then
16                         Cand.Push(NewNeigh)
17                     end
18                 end
19             end
20         end
21         CLabel = CLabel + 1
22     end
23 end
```

they do not have a neighborhood large enough. All *density-reachable* objects are tagged with the same cluster label and are pushed into a list in which each of them will be evaluated in its turn, see line 16. Once the list is emptied, the algorithm increments the cluster label and continues with the next not assigned object until completion.

Neighborhood identification

Identifying the correct neighborhood is essential in any clustering problem and even more relevant when using density-based approaches. Spatial clustering with DBSCAN typically uses the Euclidean distance to look for the neighborhood which is ϵ apart, although other distances, such as Manhattan or Hamming, could also be employed.

In the proposed algorithm, we motivate to use Euclidean distance for the spatial and temporal dimensions, but keeping different parameters (ϵ_1, ϵ_2) for each dimension. ST-DBSCAN [6] also adds these two parameters arguing that clusters are not anymore constraint to have the same density along both axes. Moreover, we introduce a third parameter (ϵ_3) to identify the textual neighborhood independently from the spatial and temporal ones.

Regarding the identification of the textual neighborhood, we propose to use the Jensen-Shannon distance, Eq. (1a), which fulfills the triangle inequality and it was originally thought to measure the similarity between probability distributions [10], such as the topic distributions per tweet. The Jensen-Shannon distance makes use of the Kullback-Leibler divergence, Eq. (1b), which is an asymmetric measure between probability distributions.

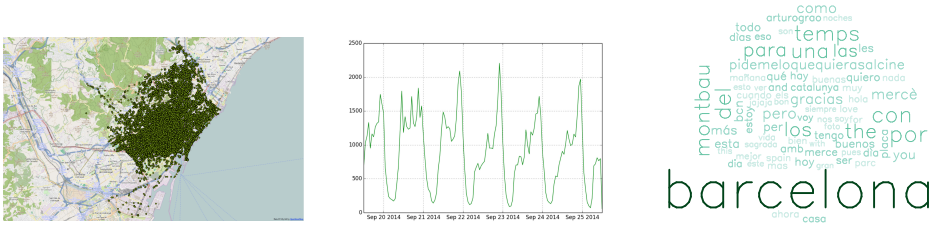
$$JSD(p,q) = \sqrt{\frac{1}{2}D_{KL}(p||m) + \frac{1}{2}D_{KL}(q||m)} \tag{1a}$$

$$D_{KL}(p||m) = \sum_i p(i) log_2 \frac{p(i)}{m(i)} \quad m = \frac{1}{2}(p+q) \tag{1b}$$

In Tweet-SCAN, p and q from Eq. (1a) are two probability distributions over topics which are associated to two tweet messages. The Jensen-Shannon distance will output an real value within the $[0, 1]$. Documents with the same topic distribution will have a Jensen-Shannon distance equal to 0 and distance for those distributions which are very far apart will be close to 1.

3. Barcelona data set

A handcrafted Twitter data set was created with the aim to evaluate the event discovery features of the proposed Tweet-SCAN algorithm. Through HERMES [13], a cloud computing service used to perform social media analytics, we were capable of retrieving geo-located tweets generated within a bounding box that include the whole city of Barcelona[3]. Within the city region, 45.623 tweets were collected from the 19th to the 25th of September of 2014, period that matches with the annual local festivities called *la Mercè*. Fig. 2 shows the spatial, temporal and textual patterns present in this raw data set.



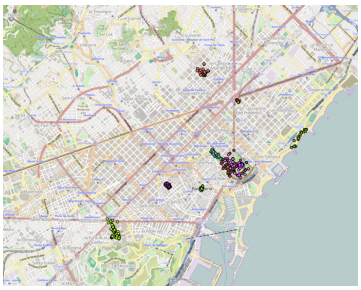
(a) Spatial dimension (b) Temporal dimension (c) Textual dimension

Figure 2. Tweets from Barcelona festivities on September 2014.

For the sake of evaluation, we have also tagged 12 known events taking place in Barcelona during those days, see Fig. 3. Based on these tagged events, we will assess the

[3]Barcelona has 1,602,386 citizens in the 101.4 km^2 sized municipality according to Idescat

algorithm capabilities to distinguish them from noise. The known events are very diverse in nature ranging from cultural or leisure events to political acts or gastronomic tastings. As it can be seen in Fig. 3a, some of the events share its location in the city center whilst others are geographically isolated.



(a) Tagged events over Barcelona

Event	Number of tweets	Most common hashtags
CaixaForum conference	21	#mlearningcat
ATypl2014 conference	99	#ATypl2014
Bogatell Beach concerts	61	#Merce2014 #Txarango #mediterrània ment
Fàbrica Damm concerts	89	#AntigaFàbrica, #BAM14 #Merce2014
MACBA concerts	131	#macba, #BAM14 #Merce2014
Maria Cristina concerts	30	#40merce,#Merce2014
Referendum law	167	#lleiconsultes, #9N, #Parlament
Human towers	47	#castellers, #humantowers, #Merce2014
Fireworks	59	#piromusical, #tricentenari, #Merce2014
Mapping Barcekholm	80	#Barcekholm, #projecció, #Merce2014
Van market	281	#VanVanMarket, #ParcdelaCiutadella, #food
Wine Tasting	119	#vins, #mostracat, #Merce2014

(b) Events overview

Figure 3. Tagged events from Barcelona festivities on September 2014.

4. Tweet-SCAN assessment

Tweet-SCAN is evaluated in terms of sensitivity, a well known statistical measure in binary classification problems [16]. However, we build here the confusion matrix for twelve known events in the rows, and let the algorithm fix the number of estimated events in the columns. We then calculate the sensitivity for each of the tagged events by considering that the event was correctly identified only if most of the tweets belong to an estimated cluster which is not the noise label nor a label associated to another event. These twelve sensitivity measures are then averaged in an unweighted manner. Nonetheless, this sensitivity figure can be a little bit misleading since we do not enforce the non-tagged tweets to belong to a specific class. As we will see, high sensitivity figures results in large number of estimated events. Since we know that there should not be a huge number of events in this data set, we decide to incorporate the number of estimated events into this assessment.

First of all, we aim to find proper spatio-temporal parameters values that perform best in terms of high sensitivity and reasonable number of estimated events. The plot at the left side of Fig. 4 represents the averaged sensitivity and number of events as a function of $Minpts$ for several different ϵ_1, ϵ_2 settings. In fact, we here consider two possible values for $\epsilon_1 = 250m, 500m$ and two more for $\epsilon_2 = 1800s, 3600s$ which suits the size of the city and type of events. In particular, we observe that $\epsilon_1 = 250m, \epsilon_2 = 1800s$ curves for a value of $Minpts = 10$ accomplishes our restrictions quite satisfactory.

Considering this parameter settings, the map at the right side of Fig. 4 plots each geo-located tweet associated to an event with different coloring and excludes tweets considered noise. Clearly, events are apparently well-identified in areas with low tweet density, whereas events are still masked in higher density regions. By increasing $Minpts$, we would be able to reduce the number of estimated events as shown in plot from Fig. 4, but this will simply cause losing those clusters less dense spatial and temporarily speak-

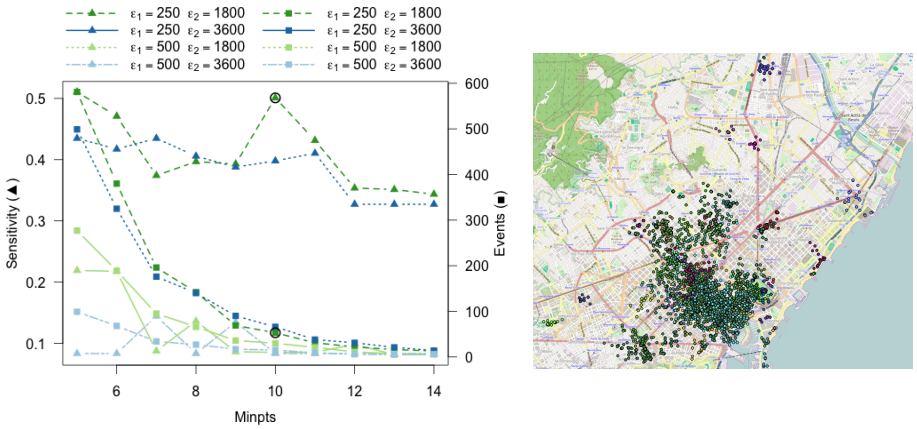


Figure 4. Tweet-SCAN evaluation for different spatio-temporal settings ($\epsilon_3 = 1$).

ing. This motivates adding the textual dimension to rule the homogeneity of clusters in terms of textual similarities.

To understand how the textual dimension works, we must set ϵ_3 to some value less than 1. The lower ϵ_3 is, the closer tweets must be in their textual component in order to form a cluster. Because of this, we plot at the left side of Fig. 5 the averaged sensitivity and number of estimated events for different values of ϵ_3 within the $[0, 1]$ interval. As it can be seen in this plot, the system reaches its best performance in terms of in terms of sensitivity for ϵ_3 values of 1. However, the drawback for $\epsilon_3 = 1$ is that the number of estimated events is quite high, which we know it is not the case for this specific data set. On the other hand, we also see that lower ϵ_3 values, like those around 0.7, keep the number of events in a reasonable range, while still getting an acceptable sensitivity for some $Minpts$ values. In particular, we see that $\epsilon_3 = 0.7$ curves for a value of $Minpts = 10$ would result in acceptable evaluation figures.

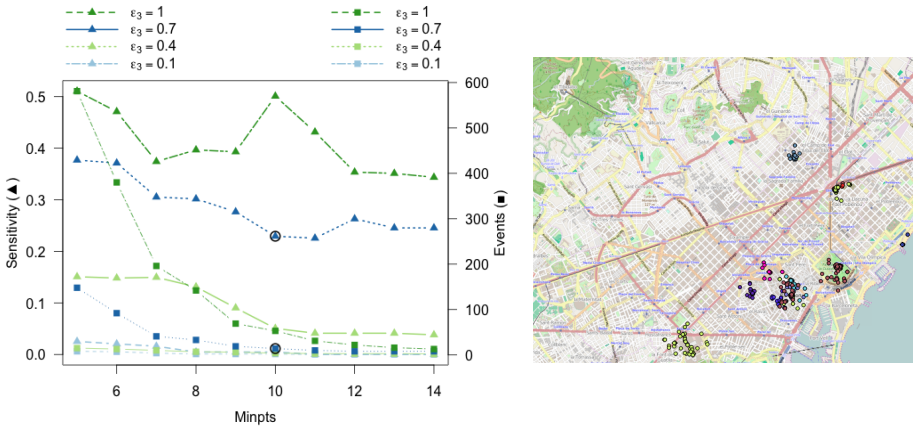


Figure 5. Tweet-SCAN evaluation for different textual settings ($\epsilon_1 = 250m$ and $\epsilon_2 = 1800s$).

The map at the right side of Fig. 5 was plotted with $\epsilon_3 = 0.7$ and $Minpts = 10$ and the resulting events resembles the tagged events from Fig. 3. Clearly, higher density areas, e.g. the city center, are now better discriminated while some of the events in lower density areas are not lost. In general, all clusters has become more homogeneous and meaningful in terms of text. Nonetheless, we must advocate for robust document models which can best represent tweets textual content to not loss sensitivity when using a textual component.

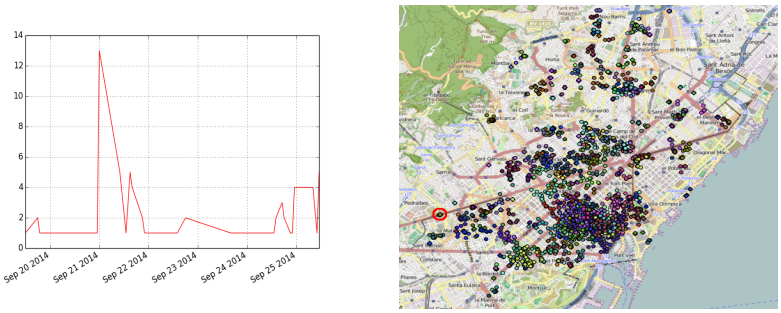


Figure 6. User temporal pattern (left). Estimated events for $u = 0$ (right).

Last, we have stated that Tweet-SCAN is capable of mitigating user interference through the user diversity parameter u. Up to this point, we have used a value for u equal to 0.5 which means that clusters at least contain 50% of unique users. To show how user interference mitigation is disabled, we set the user diversity parameter to zero, and look at the estimated events on the map from Fig. 6. Definitely, the number of estimated events greatly increased compared with $u = 0.5$. Moreover, we now observe spatially concentrated clusters such as the circled one, which are due to a unique user who is exhaustingly posting geo-located tweets about a similar topic. Plot at the left side of Fig. 6 draws the temporal profiling of the user generating the circled cluster. Clearly, its tweeting behavior the first hours of September 21th explains the estimated event when the user diversity parameter u is completely disabled.

5. Conclusions and Future work

To our best knowledge, Tweet-SCAN provides a first step in using spatio-temporal, textual and user features for the purpose of uncovering real world event from Location-based Social Networks. Moreover, the extension of a well known clustering technique, such as DBSCAN, to incorporate search within textual objects could boost the applications of this algorithm in many other areas. Last but not least, the initial results of Tweet-SCAN for event detection are very promising and further tuning of the system could lead to a very powerful unsupervised discovery system.

Despite that, the assessment of the algorithm could only be performed on a subset of tagged events what hampers a full evaluation in terms of classification measures. We leave for future work this full assessment of Tweet-SCAN in terms of sensitivity and specificity for the data set entirely tagged, events, noise and interference. Furthermore, other document modeling techniques could be employed to represent the textual com-

ponent of tweets, which could lead to even more promising results. Under this rigorous evaluation scheme, we could assess and compare these text modeling techniques in order to gain more insights.

Acknowledgments

This work is partially supported by Obra Social "la Caixa", by the Spanish Ministry of Economy and Competitivity under contract TIN2012-34557, by the BSC-CNS Severo Ochoa program (SEV-2011-00067), by the SGR programme (2014-SGR-1051) of the Catalan Government and by COR (TIN2012-38876-C02-01) project.

References

[1] Javier Borge-Holthoefer, Alejandro Rivero, Iñigo García, Elisa Cauhé, Alfredo Ferrer, Darío Ferrer, David Francos, David Iñiguez, María Pilar Pérez, Gonzalo Ruiz, et al. Structural and dynamical patterns on online social networks: the spanish may 15th movement as a case study. *PloS one*, 6(8):e23883, 2011.

[2] Eui-Ki Kim, Jong Hyeon Seok, Jang Seok Oh, Hyong Woo Lee, and Kyung Hyun Kim. Use of hangeul twitter to track and predict human influenza infection. *PloS one*, 8(7):e69305, 2013.

[3] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*, pages 851–860. ACM, 2010.

[4] Yu Zheng. Tutorial on location-based social networks. In *WWW 2012*. ACM, May 2012.

[5] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231, 1996.

[6] Derya Birant and Alp Kut. St-dbscan: An algorithm for clustering spatial–temporal data. *Data and Knowledge Engineering*, 60(1):208 – 221, 2007. Intelligent Data Mining.

[7] David M Blei. Probabilistic topic models. *Communications of the ACM*, 55(4):77–84, 2012.

[8] Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. Hierarchical dirichlet processes. *Journal of the american statistical association*, 101(476), 2006.

[9] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[10] Dominik Maria Endres and Johannes E Schindelin. A new metric for probability distributions. *IEEE Transactions on Information theory*, 2003.

[11] Quan Yuan, Gao Cong, Zongyang Ma, Aixin Sun, and Nadia Magnenat Thalmann. Who, where, when and what: Discover spatio-temporal topics for twitter users. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pages 605–613, New York, NY, USA, 2013. ACM.

[12] James McInerneya and David M Blei. Discovering newsworthy tweets with a geographical topic model. 2014.

[13] Daniel Cea, Jordi Nin, Rubén Tous, Jordi Torres, and Eduard Ayguadé. Towards the cloudification of the social networks analytics. In *Modeling Decisions for Artificial Intelligence*, pages 192–203. Springer, 2014.

[14] Dario Garcia-Gasulla, Sergio Alvarez-Napagao, Arturo Tejeda-Gómez, Luis Oliva-Felipe, Javier Vázquez-Salceda, Ignasi Gómez-Sebastià, and Javier Bejar. Social network data analysis for event detection. In *21st European Conference on Artificial Intelligence (ECAI2014)*, volume 263, pages 1009–1010. IOS Press, 2014.

[15] Liangjie Hong and Brian D Davison. Empirical study of topic modeling in twitter. In *Proceedings of the First Workshop on Social Media Analytics*, pages 80–88. ACM, 2010.

[16] Vincent Labatut and Hocine Cherifi. Accuracy measures for the comparison of classifiers. *arXiv preprint arXiv:1207.3790*, 2012.