Noname manuscript No.
(will be inserted by the editor)

Event Detection in Location-based Social Networks

Joan Capdevila · Jesús Cerquides · Jordi Torres

the date of receipt and acceptance should be inserted later

Abstract With the advent of social networks and the rise of mobile technologies, users have become ubiquitous sensors capable of monitoring various real-world events in a crowd-sourced manner. Location-based social networks have proven to be faster than traditional media channels in reporting and geo-locating breaking news, i.e. Osama Bin Laden's death was first confirmed on Twitter even before the announcement from the communication department at the White House. However, the deluge of user-generated data on these networks requires intelligent systems capable of identifying and characterizing such events in a comprehensive manner. The data mining community coined the term, *event detection*, to refer to the task of uncovering emerging patterns in data streams. Nonetheless, most data mining techniques do not reproduce the underlying data generation process, hampering to self-adapt in fast-changing scenarios. Because of this, we propose a probabilistic machine learning approach to event detection which explicitly models the data generation process and enables reasoning about the discovered events. With the aim to set forth the differences between both approaches, we present two techniques for the problem of event detection in Twitter: a data mining technique called Tweet-SCAN and a machine learning technique called WARBLE. We assess and compare both techniques in a dataset of tweets geo-located in the city of Barcelona during its annual festivities. Last but not least, we present the algorithmic changes and data processing frameworks to scale up the proposed techniques to big data workloads.

Joan Capdevila
Universitat Politècnica de Catalunya (UPC), Barcelona Supercomputing Center (BSC)
E-mail: jc@ac.upc.edu

Jesús Cerquides
Artificial Intelligence Research Institute (IIIA), Spanish National Research Council (CSIC)
E-mail: cerquide@iiia.csic.es

Jordi Torres
Universitat Politècnica de Catalunya (UPC), Barcelona Supercomputing Center (BSC)
E-mail: torres@ac.upc.edu

Keywords Event detection · Social Networks · Geolocation · Twitter · Anomaly Detection · DBSCAN · Topic Models · Probabilistic Modeling · Variational Inference · Apache Spark

1 Introduction

Sensor networks are systems composed of several tenths of spatially-distributed autonomous devices capable of monitoring their surroundings and communicating with their neighbors (Akyildiz et al, 2002). Detecting abnormal behaviors in these networks have attracted the interest of different communities ranging from communications (Rajasegarar et al, 2008) to data mining (Chandola et al, 2009). In particular, the task of detecting and characterizing anomalous subgroups of measurements that emerge in time has been coined as *event detection* and it has found many applications in surveillance systems, environmental monitoring, urban mobility, among many others (Wong and Neill, 2009).

In contrast, social networks came about to interconnect users mainly for communication purposes. However, the rise of mobile technologies and positioning systems have turned users into ubiquitous sensors capable of monitoring and reporting real-world events (i.e. music concert, earthquakes, political demonstration). Most of these events are very challenging to detect through sensor networks, but location-based social networks, which incorporate geo-tagging services, have shown to report them even faster than traditional media (Zheng, 2012). For example, Mumbai terrorist attacks were instantly described on Twitter by several eyewitness in the crime area (Stelter and Cohen, 2008) and Osama Bin Laden's death was first revealed on the same platform before the communication department at the White House had even confirmed his death (Newman, 2011).

Therefore, there has recently been a growing interest to build intelligent systems which are able to automatically detect and summarize interesting events from online social content (Panagiotou et al, 2016). In particular, Twitter has attracted most of the attention in both research and industry because of its popularity [1] and its accessibility [2] (Atefeh and Khreich, 2015). Tweet messages respond to the *What's happening?* question through a 140-character-long text message, and tweet meta-data might also contain details about the when, where and who (Yuan et al, 2013). Social networks in general, and Twitter in particular, are classic big data scenarios in which large volumes of heterogeneous data are generated in streaming by millions of uncoordinated users. Applications such as event detection have to consider these challenges in order to generate veracious knowledge from this data. In other words, event detection in Twitter has to deal with the 5 Vs defined in big data: *volume*, *velocity*, *variety*, *veracity* and *variability*.

In this chapter, we present two techniques for retrospective event detection, that is to say that both techniques seek to discover events from historical data, not from a stream. As a result, *velocity* is disregarded for this retrospective study, but left for future work in online or prospective setups. Both techniques deal with tweet *variety*

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

[2] https://dev.twitter.com/rest/public

by modeling the spatial, temporal and textual dimensions of a tweet independently. They could also be extended to take into account other forms of data (image, video, etc.).

The first technique, called Tweet-SCAN (Capdevila et al, 2016a), is based on the Density-based Spatial Clustering of Applications with Noise (DBSCAN) (Ester et al, 1996). This is a well-known algorithm for bottom-up event detection from the data mining community (Wong and Neill, 2009). This algorithm identifies as events groups of densely packed tweets which are about similar themes, location and time period. However, such techniques do not consider uncertain measurements (i.e. GPS errors) or partial information (i.e. tweets without location), compromising the veracity of the results. Moreover, these detection techniques lack of knowledge about the data generation process hampering them to adapt in *varying* scenarios. Nonetheless, parallel and distributed versions of DBSCAN (He et al, 2014a) are enabling to scale up event detection in large datasets (Capdevila et al, 2016d).

On the other hand, computational intelligent approaches like probabilistic models and learning theory can help to mitigate some of these issues by accounting for the uncertainty in a very principled way (Bishop, 2013). WARBLE (Capdevila et al, 2016b), the second technique presented here, follows this approach and tackles the event detection problem through heterogeneous mixture models (Banfield and Raftery, 1993). These are probabilistic models that represent sub-populations within an overall population, and each sub-population might be generated by a different statistical distribution form. Last but not least, recent advances in approximate inference have mitigated the high computational cost in learning probabilistic models in scenarios with large *volumes* of data (Hoffman et al, 2013).

The rest of this chapter is structured as follows. In section 2, we define the problem of event detection in location-based social networks. We then provide the necessary background regarding DBSCAN and mixture models in section 3. Section 4 contains detailed explanation about the two event detection techniques and their scaling in the presence of large data volumes. Tweet-SCAN is described in section 4.1 and WARBLE, in section 4.2. The experimental setup and results is in section 5. We first introduce "La Mercè" dataset for local event detection in section 5.1, we then present the metrics to evaluate the detection performance in section 5.2 and we ultimately evaluate both techniques in section 5.3. Finally, section 6 presents some conclusions out of this chapter and points out to several future steps.

2 Problem Definition

Event detection in social networks lacks of a formal definition for an event, hampering the progress of this field. Broadly speaking, McMinn et al (2013) defined an event as "a *significant* thing that happens at some specific time and place". However, this definition does not specify what *significant* means in the context of social networks. Lately, Panagiotou et al (2016) built on top of this definition to provide the following one:

Definition 1 Event (e): In the context of Online Social Networks (OSN), (significant) event (e) is something that cause (a large number of) actions in the OSN.

Note first that it does not constrain an event to happen at some specific time and place, in contrast to (McMinn et al, 2013). This enables to have a more general definition from which we can then distinguish several event types (Global, Local or Entity-related) depending on the constraints. However, this definition still lacks of some sort of formalization regarding the significant number of actions in the social network (e.g. post new content or accept a friend request). With the aim to unify and formalize this, we add to Definition 1 the idea that events are caused by abnormal occurrences:

Definition 2 Event (e): In the context of Online Social Networks (OSN), (significant) event (e) is something that cause an *abnormal* number of actions in the OSN.

This definition resembles that of event detection in sensor networks (Wong and Neill, 2009), in which events are anomalous occurrences that affect a subgroup of the data. Note also that this captures more complex events than Definition 1. For example, an abnormal decrease of actions in the social network, as it might initially happen during a shooting in a crowded area, should be also considered a significant event.

Moreover, location-based social networks have enabled to narrow down the scope of events to geo-located events (Zheng, 2012), enabling the identification of many real-world occurrences such as music concerts, earthquakes or political demonstrations. Moreover, by restricting the geographical dimension of such events, we are able to identify local events taking place in urban environments, which will be the application of the techniques presented in this chapter.

Therefore, the task of event detection in a social network consists of identifying and characterizing a set of events that are anomalous with respect to a baseline. This task can be performed either retrospectively or prospectively. While the former aims to retrieve events from historical data in a batch mode, the latter seeks to identify them in streaming data in an online fashion. In the following sections, we will present two different approaches to retrospectively uncover these anomalous patterns:

1. Tweet-SCAN: A data mining approach based on DBSCAN (Ester et al, 1996) in which events are groups of posts (i.e. tweets) that are more densely packed than the baseline.
2. WARBLE: A probabilistic approach based on heterogeneous mixture models (Banfield and Raftery, 1993) in which events are groups of posts (i.e. tweets) generated by a statistical distribution different from that of non-event tweets.

Both techniques follows the anomaly-based approach to event detection by assuming that events are groups of similar tweets (in space, time and textual meaning) and they are masked by tones of non-event tweets such as *memes*, user conversations or re-post activities. While Tweet-SCAN considers distance as the metric for similarity, WARBLE uses probability to assess the pertinence to event or non-event.

3 Background

In this section, we present digested background regarding DBSCAN and mixture models, methods that are used by the later proposed techniques. Both methods have

been used for clustering in applications with noise. We instead propose them for retrospective event detection, given that the noise component can be used to for modeling the baseline or expected behavior.

3.1 DBSCAN

DBSCAN (Ester et al, 1996) was initially proposed to uncover clusters with arbitrary shapes whose points configure a dense or packed group. This means that for each point in a cluster its neighborhood at a ϵ distance must contain at least a minimum number of points, $MinPts$. Formally, this implies the definition of two predicates:

1. $NPred(o, o') \equiv N_\epsilon(o, o') = |o - o'| \leq \epsilon$.
2. $MinWeight(o) \equiv |\{o' \in D \mid |o - o'| \leq \epsilon\}| \geq MinPts$.

The fulfillment of both predicates allows to define the notion of a point p being directly density-reachable from another point q, see (left) Fig. 1, where ϵ is given by the circle radius and $MinPts$ is set to 2. In this scenario, q is a *core point* because it satisfies both predicates and p is a *border point* since it breaks the second predicate. The notion of being direct reachable is extended to density-reachable points when p and q are far apart, but there is a chain of points in which each pair of consecutive points are directly density-reachable, as it is the case in (middle) Fig. 1. Finally, it might happen that p and q are not density-reachable, but there is a point o from which they are both density-reachable, that is when p and q are said to be density-connected, for example in (right) Fig. 1. Note that both points, p and q, are here *border points*, while o is a *core point*.

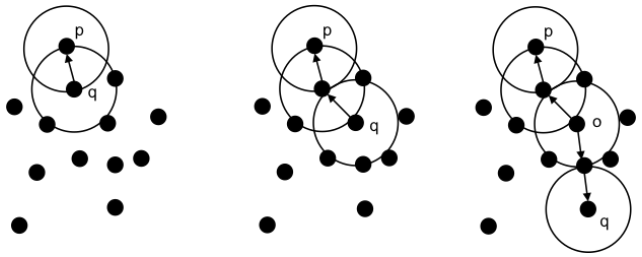


Fig. 1: Directly density-reachable (left), density-reachable (middle) and density-connected (right) points.

Consequently, a cluster in DBSCAN is defined to be a set of density-connected points that contains all possible density-reachable points. Furthermore, *noise points* can now be defined as those points which do not belong to any cluster since they are not density-connected to any.

GDBSCAN (Sander et al, 1998) generalizes DBSCAN by redefining the above-mentioned predicates to cope with spatially extended objects. For example, the neighborhood of a set of polygons is defined by the intersect predicate instead of a distance

function. It is also the case for a set of points with financial income attributes within a region whose $MinWeight$ predicate is a weighted sum of incomes instead of mere point cardinality, so that clusters become regions with similar income. Therefore, both predicates can be generalized as follows:

1. $NPred(o, o')$ is binary, reflexive and symmetric.
2. $MinWeight(o) \equiv wCard(\{o' \in D \mid NPred(o, o')\}) \geq MinCard$, where $wCard$ is a function that $2^D \rightarrow \mathbb{R}^{\geq 0}$

These new predicates enable to extend the concept of density-connected points to objects and thus generalize density-based clustering to spatially extended objects, like geo-located tweets. Moreover, we note that DBSCAN-like techniques have been considered for event detection in sensor networks as a *bottom-up* approach (Wong and Neill, 2009).

3.2 Mixture Models

Mixture models are probabilistic models for representing the presence of subpopulations within an overall population and they have been very popular for clustering and unsupervised learning. Mixture of Gaussians or Gaussian Mixture Models (GMM) are the most widely used mixture model (Murphy, 2012). In this model, each mixture component is a multivariate Gaussian with mean μ_k and covariance Σ_k parameters. This means that given the component assignment c_n, the generative process for the n-th observations is,

$$x_n \sim N\left(\mu_{c_n}, \Sigma_{c_n}\right). \tag{1}$$

Banfield and Raftery (1993) proposed a more general model in which not all mixture components share the same distributional form. In particular, observations from one of the mixture components came from a Poisson process associated with noise. Therefore, the generative process can be rewritten as,

$$x_n \sim \begin{cases} N\left(\mu_{c_n}, \Sigma_{c_n}\right) & c_n < K \\ U(x_{min}, x_{max}) & c_n = K \end{cases} \tag{2}$$

where $U(x_{min}, x_{max})$ corresponds to a multivariate uniform distribution with x_{min}, the most south-western point and x_{max}, the most north-eastern point. This model has shown to perform reasonably well in cluster recovery from noisy data in both synthetic and real datasets (Fraley and Raftery, 2002).

Heterogeneous mixture models enable to propose generative models for event detection, in which event-related observations, i.e. tweets, are drawn from a distributional $(c_n < K)$ form that entails locality in the temporal, spatial and textual dimensions, while non-event data points are generated from the background distribution $(c_n = K)$.

4 Event Detection Techniques

4.1 Tweet-SCAN: a Data Mining Approach

Tweet-SCAN (Capdevila et al, 2016a) is defined by specifying the proper neighborhood and MinWeight predicates introduced in Section 3.1 for GDBSCAN in order to associate density-connected sets of tweets to real-world events. Next, we introduce both predicates and the text model for the textual component of a tweet.

4.1.1 Neighborhood predicate

Most event-related tweets are generated throughout the course of the event within the area where it takes place. Consequently, we need to find sets of tweets density-connected in space and time, as well as in meaning.

We also note that closeness in space is not comparable to time, nor to meaning. Because of this, Tweet-SCAN is defined to use separate positive-valued ϵ_1, ϵ_2, ϵ_3 parameters for space, time and text, respectively. Moreover, specific metrics will be chosen for each dimension given that each feature contains different type of data.

The neighborhood predicate for a tweet o in Tweet-SCAN can be expressed as follows,

$$NPred(o, o') \equiv |o_1 - o'_1| \leq \epsilon_1,\ |o_2 - o'_2| \leq \epsilon_2,\ |o_3 - o'_3| \leq \epsilon_3 \tag{3}$$

where $|o_i - o'_i|$ are distance functions defined for each dimension, namely space, time and text. The predicate symmetry and reflexivity are guaranteed as long as $|o_i - o'_i|$ are proper distances. Particularly, we propose to use the Euclidean distance for the spatial and temporal dimensions given that latitude and longitude coordinates as well as timestamps are real-valued features and the straight line distance seems a reasonable approximation in this scenario. The metric for the textual component will be defined later once we present the text model for Tweet-SCAN.

4.1.2 MinWeight predicate

Tweet-SCAN seeks to group closely related tweets generated by a diverse set of users instead of a reduced set of them. User diversity is imposed to avoid that a single user continuously posting tweets from nearby locations could trigger a false event in Tweet-SCAN. Forcing a certain level of user diversity within a cluster can be achieved through two conditions in the $MinWeight$ predicate that must be satisfied at the same time,

$$MinWeight(o) \equiv |N_{NPred}(o)| \geq MinPts,\ UDiv(N_{NPred}(o)) \geq \mu \tag{4}$$

where $N_{NPred}(o)$ is the set of neighboring tweets of o such that $\{o' \in D \mid NPred(o, o')\}$ w.r.t. the previously defined Tweet-SCAN neighborhood predicate. The first condition from the MinWeight predicate establishes that neighboring tweets must have a minimum cardinality $MinPts$ as in DBSCAN. While in the second condition, the user diversity $UDiv()$ ratio, which is defined as the proportion of unique users within the set $N_{NPred}(o)$, must be higher than a given level μ of user diversity.

4.1.3 Text model

The text message in a tweet is a 140-character-long field in which users type freely their thoughts, experiences or conversations. The fact that users tweet in different languages, argots and styles dramatically increases the size of the vocabulary, making the use of simple Bag of Words (BoW) models (Salton et al, 1975) not viable. Therefore, we propose to use probabilistic topic models, which are common dimensionality reduction tools in text corpus (Blei, 2012). In this approach, a tweet message is encoded into a K-dimensional vector which corresponds to the Categorical probability distribution over the K topics. K is often much smaller than the vocabulary size and the resulting topics are represented by semantically similar words.

Nonparametric Bayesian models like Hierarchical Dirichlet Process (HDP) (Teh et al, 2006) can automatically infer the number of topics K, overcoming the limitation of their parametric counterparts like Latent Dirichlet Allocation (LDA) (Blei et al, 2003). The HDP topic model basically consists of two nested Dirichlet Process: G_o, with base distribution H and concentration parameter γ, and G_i, with base distribution G_o and concentration parameter α_o. Although the number of topics is automatically inferred, the hyperparameters γ and α_o might strongly influence the number of components. Because of this, vague informative gamma priors such as, $\gamma \sim\ Gamma(1, 0.1)$ and $\alpha_o \sim\ Gamma(1, 1)$ are usually considered (Escobar and West, 1995; Teh et al, 2006).

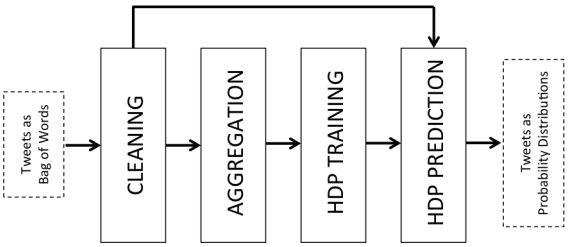


Fig. 2: Text model scheme. Stages are highlighted in bold in the text.

The straightforward use of HDP models on raw tweets does not provide meaningful topic distributions (Hong and Davison, 2010) due to the lack of word co-occurrence in short texts like tweets. Because of this, we propose the scheme from Fig. 2 which aims to alleviate these shortcomings. First, raw tweets, modeled as Bag of Words, are pre-processed and **cleaned** through classical data cleaning techniques from Natural Language Processing (NLP): lowering case, removing numbers and special characters, and stripping white-spaces. Then, processed tweets are **aggregated** to build longer training documents from a group of concatenated tweets. These aggregated documents are used to **train** the HDP model. Finally, the trained HDP model is employed to **predict** the topic distributions per each single tweet in order to obtain the Categorical probability distributions over the K topics that summarize each tweet message.

In the aggregation stage, we consider the aggregation scheme by top key terms proposed in (Hong and Davison, 2010). This consists in first identifying a set of top key terms through the TF-IDF statistic (Salton and Buckley, 1988), and then aggregating tweets that contains each of these top keywords. Thus, there will be as many training documents as top key terms and very few tweets will be unassigned as long as we choose a reasonable number of top keywords.

Finally, we propose to use the Jensen-Shannon (JS) distance for the textual component in Tweet-SCAN neighborhood predicate. JS is a proper distance metric for probability distributions (Endres and Schindelin, 2003). It is defined as,

$$JS(p,q) = \sqrt{\frac{1}{2}D_{KL}(p||m) + \frac{1}{2}D_{KL}(q||m)} \tag{5}$$

where p, q and m are probability distributions and $D_{KL}(p||m)$ is the Kullback-Leibler divergence between probability distribution p and m written as,

$$D_{KL}(p||m) = \sum_i p(i) log_2 \frac{p(i)}{m(i)} \quad m = \frac{1}{2}(p+q) \tag{6}$$

where m is the average of both distributions.

In Tweet-SCAN, p and q from equation (5) are two Categorical probability distributions over topics which are associated to two tweet messages. Given that Jensen-Shannon distance is defined through base 2 logarithms, JS distance will output a real value within the $[0, 1]$. Documents with the similar topic distribution will have a Jensen-Shannon distance close to 0 and those topic distributions which are very far apart, distance will tend to 1.

4.1.4 Scaling up to large datasets

To scale up Tweet-SCAN to large datasets, we propose to build on current parallel versions of DBSCAN such as MR-DBSCAN (He et al, 2014b) which parallelizes all the critical sub-procedures of DBSCAN. The MR-DBSCAN workflow, shown in Fig. 3, first partitions the full dataset, then performs local DBSCAN clustering in each partition, and finally merges the local clusters into global ones, which correspond to events in our case.

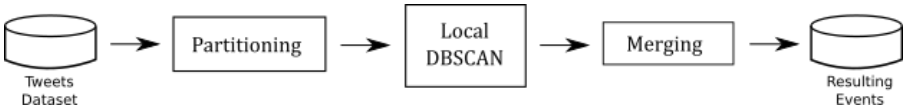


Fig. 3: Simplified MR-DBSCAN workflow

An implementation of MR-DBSCAN in Apache Spark named RDD-DBSCAN was proposed by Cordova and Moh (2015). Apache Spark (Zaharia et al, 2010) is a computing framework in which distributed data collections, called Resilient Distributed Datasets (RDD), can be cached into memory for fast map-reduce operations.

The extension of DBSCAN algorithm for large scale event detection based on RDD-DBSCAN was developed by (Capdevila et al, 2016d) and preliminary results show that by increasing parallelism we can reduce computation time.

4.2 WARBLE: a Machine Learning Approach

Next, we introduce WARBLE (Capdevila et al, 2016b) a probabilistic model and learning scheme to uncover events from tweets through heterogeneous mixture models introduced in section 3.2.

4.2.1 Probabilistic Model

McInerney and Blei (2014) proposed a probabilistic model for event detection based on homogeneous mixture models in which each mixture component shares the same distributional form. Formally, they assume that the n-th tweet $\mathbb{T}_n$ is generated according to,

$$\mathbb{T}_n \sim \ f\left(\beta_{e_n}\right) \tag{7}$$

where f is the probability distribution function (pdf), common for all mixture components and β_k are the distribution parameters corresponding to the k-th mixture component.

As argued in the introduction, a vast majority of tweets is not event-related. Therefore, we would like to address rarity of event data by introducing a new mixture component, to which we will refer as **background**, which contains those tweets which are not part of any event. In probabilistic terms, it seems clear that the distribution of tweets inside the background component should be widely different from that inside events.

Accordingly, the WARBLE model generalizes McInerney and Blei's model to handle heterogeneous components as introduced in section 3. To do that, for each component k, we enable a different base function f_k as

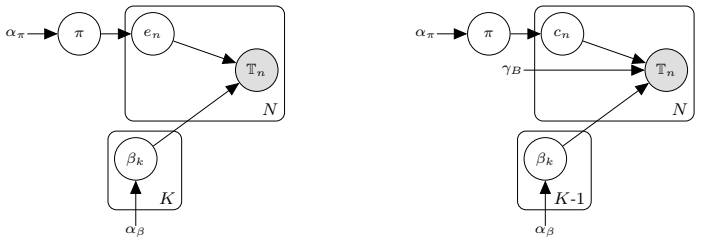
$$\mathbb{T}_n \sim \ f_{c_n}\left(\beta_{c_n}\right) \tag{8}$$

where the latent variables are now symbolized as c_n to denote that a tweet might be generated by event component ($c_n < K$) or by background ($c_n = K$).

Fig. 4 shows simplified probabilistic graphical models (PGMs) (Koller and Friedman, 2009) for McInerney and Blei's and our proposals. The proposed WARBLE model uses a different distributional form γ_B for the K-th mixture component.

Moreover, geo-located tweets tends to be unevenly distributed through space and time. For example, it is known that users are more likely to tweet during late evening and from highly populated regions (Li et al, 2013). Consequently, the background component ($c_n = K$) needs to cope with density **varying spatio-temporal distributions**.

In particular, we propose to model the distributional form γ_B for the background component through two independent histogram distributions for time and space with



(a) McInerney and Blei's model (b) WARBLE model

Fig. 4: Simplified Probabilistic Graphical Models (PGMs)

parameters T_B and L_B, respectively. The temporal histogram distribution is represented through a piecewise-continuous function which takes constant values (T_{B_1},T_{B_2}, ... $T_{B_{I_T}}$) over the I_T contiguous intervals of length b. Similarly, the spatial background is modeled through a 2d-histogram distribution over the geographical space, which is represented in a Cartesian coordinate system. The 2d-piecewise-continuous function is expressed through I_L constant values (L_{B_1},L_{B_2}, ... $L_{B_{I_L}}$) in a grid of squares with size b x b each.

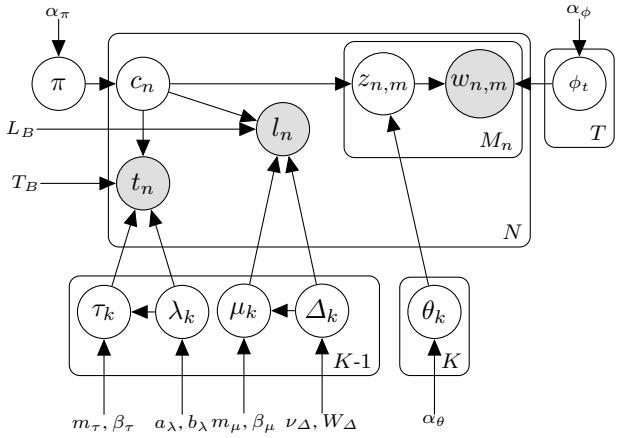


Fig. 5: The complete WARBLE model

Fig. 5 shows the complete probabilistic graphical model for the WARBLE model, where tweets $\mathbb{T}_n$ are represented by their temporal t_n, spatial l_n and textual $w_{n,\cdot}$ features.

The event-related components ($k < K$) generate the temporal, spatial and textual features from a Gaussian distribution with mean τ_k and precision λ_k, a Gaussian distribution with mean μ_k and precision matrix Δ_k and a Categorical distribution with proportions θ_k, respectively. Moreover, priors over these distributions are assumed with hyperparameters m_τ, β_τ, a_λ, b_λ, m_μ, β_μ, ν_Δ, W_Δ and α_θ.

The background component ($k = K$) accounts for the spatio-temporal features of non-event tweets, which are drawn from the histogram distributions with parameters (L_B) and (T_B) introduced earlier. However, textual features of the K-th component are not constrained by any textual background, but drawn from a Categorical distribution with proportions θ_K and hyperparameter α_θ.

Finally, we consider T topic distributions over words $\phi = \{\phi_1, \ldots, \phi_T\}$ generated from a Dirichlet distribution with hyperparameter α_ϕ. The topic distributions ϕ are learned simultaneously with component assignments c_n which has lately been found very promising in modeling short and sparse text (Quan et al, 2015) and we refer here as **simultaneous topic-event learning**. In contrast to traditional topic modeling, where distributions over topics are document-specific (Blei et al, 2003), the WARBLE model assumes that topics $z_{n,m}$ are drawn from component-specific distributions θ_k. This enables to directly obtain topics that are event-related or background-related, providing also an interesting approach for automatic event summarization.

4.2.2 Learning from tweets

Next, we describe how we can learn the WARBLE model from tweets to identify a set of events in a region during a period of interest. We first show how to learn the background model and later explain the assignment of tweets to events or background components.

Learning the background model. To learn the spatio-temporal background, we propose to collect geo-located tweets previous to the period of interest in order to add a sense of 'normality' to the model.

From the collected tweets, the temporal background is built by first computing the daily histogram with I_T bins. Then, the daily histogram is smoothed by means of a low pass Fourier filter in order to remove high frequency components. The cut-off frequency f_c determines the smoothness of the resulting signal. The normalized and smoothed histogram provides the parameters for the temporal background $T_{B_1}, T_{B_2}, ...T_{B_{I_T}}$.

The spatial background is built following the same procedure. However, geographical location has to be first projected into a Cartesian coordinate system in order to consider locations in a 2-d Euclidean space. The spatial range limits can be determined from the most southwestern and northeastern points. We consider now a two dimensional Gaussian filter with a given variance σ. The resulting 2d-histogram provides the parameter for the spatial background $L_{B_1}, L_{B_2}, ...L_{B_{I_L}}$.

We suggest to set the number of bins for the temporal and spatial histograms as well as the cut-off frequency and variance empirically. Future work will examine how to automatically adjust these parameters.

Assigning tweets to mixture components. To assign tweets to mixture components, we need to find the most probable assignment of tweets to mixture components, given the data at hand. That is finding c^*,

$$c^* = \underset{c}{\operatorname{argmax}}\, p(c|l, t, w; \Gamma) \tag{9}$$

where Γ stands for the model hyperparameters L_B, T_B, α_π, α_θ, α_ϕ, m_τ, β_τ, a_λ, b_λ, m_μ, β_μ, ν_Δ and W_Δ. Exactly assessing c^* is computationally intractable for the WARBLE model.

Therefore, we propose to first use mean-field variational Bayesian inference (Fox and Roberts, 2012; Jordan et al, 1999) to approximate $p(X|D;\Gamma)$ (where X stands for the set of random variables containing c, z, π, τ, λ, μ, Δ, θ and ϕ, and D stands for our data, namely l, t, and w) by a distribution $q(X;\eta)$ (where η stands for the variational parameters). Then, assess c^* from the approximation, that is

$$c^* = \underset{c}{\operatorname{argmax}}\, q(c;\eta) = \underset{c}{\operatorname{argmax}} \int_{X-c} q(X;\eta). \tag{10}$$

The functional forms for the mean-field approximation $q(X;\eta)$ and the updates for the variational parameters can be found in a separate technical report (Capdevila et al, 2016c). Variational parameters are updated in an iterative fashion one at a time as in coordinate descent.

4.2.3 Scaling up to large datasets

Recent advances in approximate inference are enabling to scale up inference of probabilistic models to large high-dimensional datasets (Hoffman et al, 2013). In particular, the application of stochastic optimization techniques to variational inference has enabled to process datasets in an online fashion (Hoffman et al, 2010), avoiding to have the whole dataset cached in memory or even in a local machine.

The stochastic variational inference paradigm (Hoffman et al, 2013) sets a variational objective function which also uses the factorized mean-field distribution $q(X;\eta)$. However, the variational updates are now computed from noisy estimates of the objective function instead of the true gradient. As a result, the computation of noisy gradients does not require the local variational parameters for the whole dataset, but only those associated with the randomly sampled data point.

Although stochastic algorithm are sequential in nature, their parallelization have been actively researched in order to preserve the statistical correctness while speeding up the run time of the algorithm in multicore machines (Agarwal and Duchi, 2011). The straightforward application of such techniques on distributed systems with commodity hardware is not obvious due to the high latency introduced by the network. Recently, some have distributed the inference of specific probabilistic models such as Latent Dirichlet Allocation (LDA) (Newman et al, 2007), but their parallel scheme is tailored to this model.

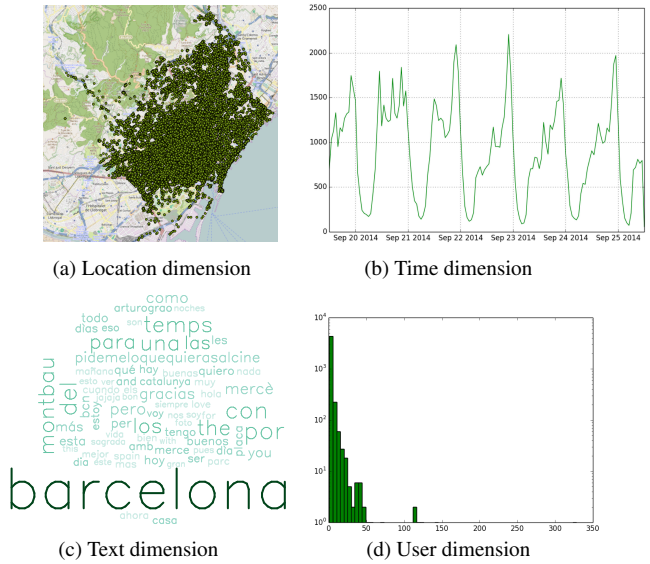
System for Parallelizing Learning Algorithm with Stochastic Methods (Splash) has been introduced as general framework for parallelizing stochastic algorithms on distributed systems (Zhang and Jordan, 2015). It is build on top of Apache Spark (Zaharia et al, 2010) and it benefits from the abstraction of this data processing engine. Splash consist of a programming interface in which the user defines the sequential stochastic algorithm and a execution engine in which it averages and reweights local updates to build the global update.

Our approach to scale up WARBLE is to use the general Splash framework built on top of Apache Spark.

5 Experimental Setup and Results

5.1 "La Mercè": a Dataset for Local Event Detection

We have collected data through the Twitter streaming API[3] via Hermes (Cea et al, 2014). In particular, we have established a long standing connection to Twitter public stream which filters all tweets geo-located within the bounding box of Barcelona city. This long standing connection was established during the local festivities of "La Mercè", that took place during few days in September 2014 and 2015[4].



(a) Location dimension (b) Time dimension

(c) Text dimension (d) User dimension

Fig. 6: Tweets dimensions from "La Mercè" 2014.

"La Mercè" festivities bring with several social, cultural and political events that happen in different locations within a considerably short period of time. This scenario is a suitable test bed for evaluating the accuracy of Tweet-SCAN on discovering these local events from tweets. Moreover, the abundance of events during these days causes that some of them overlap in time and space, making text more relevant to distinguish them. However, these events are apparently not distinguishable by analyzing tweet dimensions separately as shown in Fig. 6, where event patterns are not visible. Fig. 6a shows the spatial distribution of tweets within the borders of Barcelona city, where different tweet density levels can be appreciated in the map. Fig. 6b represents the time series of tweets from the 19th to the 25th of September and daily cycles are

[3] http://dev.twitter.com/streaming/overview

[4] Dataset published in https://github.com/jcapde87/Twitter-DS

recognizable. Fig. 6c is a wordcloud in which more frequent words are drawn with larger font size, such as "Barcelona". The multilingualism at Twitter is also reflected at this wordcloud although this work does not considered translating between different languages. Last, Fig. 6d is a histogram of the number of tweets per user, which shows that most of the users tweet very few times, while there are a few, although non-negligible number of users, who tweet very often. All four dimensions play a key role in Tweet-SCAN to uncover events.

Table 1: "La Mercè" local festivities data sets

	Tweets	Tagged tweets	Tagged events
"La Mercè" 2014	43.572	511	14
"La Mercè" 2015	12.159	476	15

As shown in Table 1, we have also manually tagged several tweets with the corresponding events as per the agenda in "La Mercè" website[5] and our own expert knowledge as citizens. With this tagged subset of tweets, we will experimentally evaluate the goodness of Tweet-SCAN. We also note that the number of tweets collected in 2015 is much less than in 2014. This is because Twitter released new smart-phone apps in April 2015 for Android and IOS that enable to attach a location to a tweet (such as a city or place of interest) apart from the precise coordinates[6]. Since tweets generated during "La Mercè" 2014 data set did not contain this functionality, we only consider tweets whose location is specified through precise coordinates for "La Mercè" 2015 data set (12.159 tweets).

5.2 Detection Performance Metrics

Clustering evaluation metrics have been applied in retrospective event detection given that this problem is defined to look for groups of tweets which are clustered together. The task of evaluating clustering against a tagged data set or *gold standard* is known as extrinsic cluster evaluation, in contrast to intrinsic evaluation, which is based on the closeness/farness of objects from the same/different clusters. Among extrinsic measures, we find out that purity, inverse purity and, specially, the combined F-measure have been extensively used for event discovery (Yang et al, 1998).

Purity is the weighted average of the maximum proportion of tweets from cluster C_i labeled as L_j over all clusters C_i, and it is expressed as follows,

$$Purity = \sum_i \frac{\mid C_i \mid}{N} max_j \frac{\mid C_i \cap L_j \mid}{\mid C_i \mid} \tag{11}$$

where higher purity means that more tweets clustered as C_i are from the same labeled event, and lower purity represents that they are from more different labels.

[5] http://lameva.barcelona.cat/merce/en/

[6] https://support.twitter.com/articles/78525

Given that the number of clusters is not fixed, we note that purity is trivially maximum when each object is set to a different cluster, but it is minimum when all objects are set to the same cluster.

To compensate the trivial solution of purity, inverse purity is introduced. Inverse purity is the weighted average of the maximum proportion of tweets labeled as event L_i that belongs to cluster C_j over all labels L_i, and it is defined as follows,

$$Inv.\,Purity = \sum_i \frac{\mid L_i \mid}{N} max_j \frac{\mid C_j \cap L_i \mid}{\mid L_i \mid} \tag{12}$$

where higher inverse purity means that more tweets labeled as event L_i are from the same cluster, and lower inverse purity represents that they are from more different clusters. Hence, Inverse Purity is trivially maximum when grouping all tweets into a unique cluster, but it is minimum if each tweet belongs to a different cluster.

Van Rijsbergen (1974) combined both measures through the harmonic mean into the Van Rijsbergen's F-measure to mitigate the undesired trivial solutions from purity and inverse purity.

The F-measure score is defined as,

$$F = \sum_i \frac{\mid L_i \mid}{N} max_j\, 2 \cdot \frac{Rec(C_j, L_i) \cdot Prec(C_j, L_i)}{Rec(C_j, L_i) + Prec(C_j, L_i)} \tag{13}$$

where L_i is the set of tweets labeled as event i and C_j is the set of tweets clustered as j and N is the total number of tweets. Recall and precision are defined over these sets as the proportions $Rec(C_j, L_i) = \frac{|C_j \cap L_i|}{|L_i|}$ and $Prec(C_j, L_i) = \frac{|C_j \cap L_i|}{|C_j|}$.

5.3 Assessment

This section assesses Tweet-SCAN and WARBLE techniques in "La Mercè" data sets presented in section 5.1 through the detection metrics introduced earlier in section 5.2. In particular, we will first show how to determine Tweet-SCAN and WARBLE parameters and we will then evaluate both tuned up techniques during the main day of "La Mercè" 2014.

5.3.1 Determining Tweet-SCAN density thresholds

We aim to determine the best performing neighborhood sizes for Tweet-SCAN in terms of its spatio-temporal, textual and user diversity parameters.

First, we assess Tweet-SCAN in terms of F-measure scores when varying ϵ_1, ϵ_2 and ϵ_3. Fig. 7 shows four possible ϵ_1, ϵ_2 configurations as function of ϵ_3 for "La Mercè" 2014 and 2015 data sets. Note that, we consider a value of $MinPts$ equal to 10, which implies that an event will have at least 10 tweets[7].

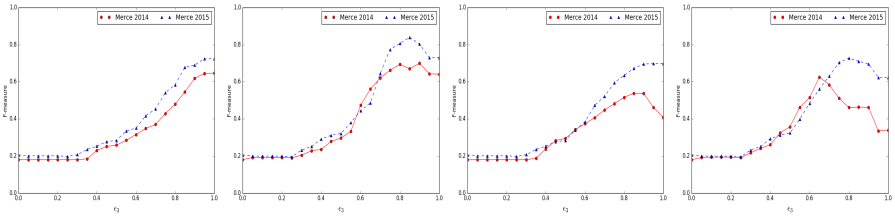
[7] Although we have tested several different $MinPts$ values, $MinPts = 10$ outperforms all others given that labeled events had at least 10 tweets.

A Tweet-SCAN configuration for short distances in time and space ($\epsilon_1 = 250m$, $\epsilon_2 = 1800s$) optimizes F-measure for $\epsilon_3 = 1$, see Fig. 7a. This means that Tweet-SCAN disregards the textual component for this spatio-temporal setup and it can be explained by the fact that these $\epsilon_1\epsilon_2$-neighborhoods are too narrow for the tagged events.

For larger temporal neighborhoods ($\epsilon_1 = 250m$, $\epsilon_2 = 3600s$), the optimum value for ϵ_3 is achieved within the range 0.8-0.9 in both data sets, see Fig. 7b. Now, we can also see that this spatio-temporal configuration performs the best.

If we increase the spatial component, but we keep the temporal short ($\epsilon_1 = 500m$, $\epsilon_2 = 1800s$), F-measure score is lower in both data sets, but the optimum value for ϵ_3 is attained within 0.8-0.9 in "La Mercè" 2014, and $\epsilon_3 = 1$ in "La Mercè" 2015, see Fig. 7c.

Last, we increase both dimensions to ($\epsilon_1 = 500m$, $\epsilon_2 = 3600s$) as shown in Fig. 7d. Although the optimum F-measure score for this setup is lower than the best performing configuration, we observe that the textual component becomes more relevant. This is due to the fact that large $\epsilon_1\epsilon_2$-neighborhoods need textual discrimination to identify meaningful events.

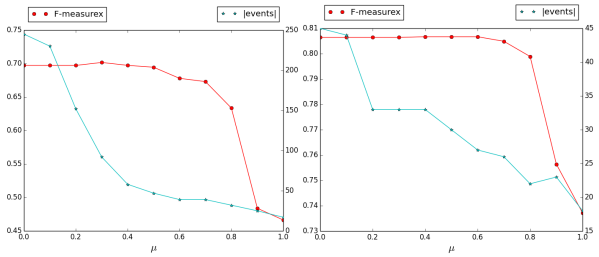


(a) $\epsilon_1 = 250m, \epsilon_2 = 1800s$ (b) $\epsilon_1 = 250m, \epsilon_2 = 3600s$ (c) $\epsilon_1 = 500m, \epsilon_2 = 1800s$ (d) $\epsilon_1 = 500m, \epsilon_2 = 3600s$

Fig. 7: F-measure for different ϵ_1, ϵ_2, ϵ_3 and $MinPts = 10$, $\mu = 0.5$.

Next, we examine the effect of different user diversity levels μ in terms of F-measure and number of discovered events. To do that, we fix the spatio-temporal and textual parameters to the best performing parameter set ($\epsilon_1 = 250m$, $\epsilon_2 = 3600s$, $\epsilon_3 = 0.8$, $MinPts = 10$) and we compute F-measure as function of the user diversity level μ. Low user diversity levels will cause that few users could generate an event in Tweet-SCAN, while higher values will entail that events are generated by many different users. Since different μ values influences the number of detected clusters by Tweet-SCAN, we will also add the number of events into the figure.

Fig. 8 plots the F-measure and number of clusters as a function of μ for both data sets. It is clear from the figures that F-measure starts decreasing after a level of μ around 0.6. Similarly, the number of discovered clusters decreases but much faster and sooner than F-measure. We observe that a user diversity level of 50% ($\mu = 0.5$) gives high figures of F-measure and reasonable number of events (≈50 events for "La Mercè" 2014 and ≈30 events for "La Mercè" 2015). Given that the size of "La Mercè" 2015 data set is nearly four times smaller, make sense to obtain less number of events for the same μ level.

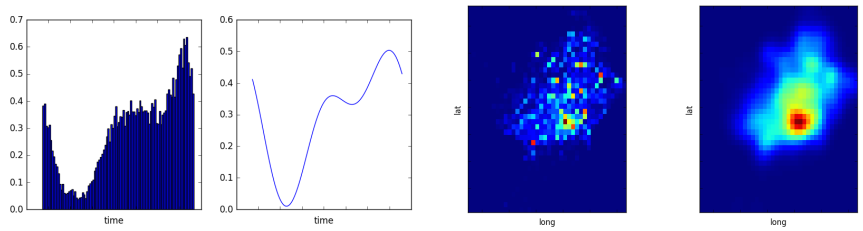


(a) "La Mercè" 2014 (b) "La Mercè" 2015

Fig. 8: F-measure for different μ values.

5.3.2 Learning WARBLE *background component*

In what follows, we learn the background component for the WARBLE model from "La Mercè" dataset. In particular, we consider all geo-located tweets from the 20th to the 23th of September 2014 to build the spatio-temporal backgrounds, L_B and T_B, to be used later in the 24th of September for event detection.



(a) Temporal background (b) Spatial background

Fig. 9: Spatio-temporal backgrounds

Fig. 9a (left) shows the daily histogram of tweets in which we observe a valley during the early morning and a peak at night, indicating low and high tweeting activity during these hours, respectively. The 1-d histogram has been computed with $I_T = 100$ bins. Fig. 9a (right) is the filtered histogram signal that will be used for setting the temporal background parameters $T_{B_1}, T_{B_2}, ...T_{B_{I_T}}$.

Fig. 9b (left) is the spatial histogram of all tweet locations. The smoothed version, Fig. 9b (right), provides the parameters for the spatial background $L_{B_1}, L_{B_2}, ...L_{B_{I_L}}$. The 2-d histogram has been computed with $I_L = 1600$ bins. We observe that the most likely areas in the filtered histogram (in red) correspond to highly dense regions of Barcelona like the city center, while city surroundings are colored in blue indicating lower density of tweets.

We note that WARBLE considers priors over most of model variables. We have considered non-informative priors and we have not experimented substantial differences in the results when varying its hyper parameters.

5.3.3 Comparative evaluation

Finally, we compare the detection performance of Tweet-SCAN and WARBLE tuned up as described in the previous sections during the main day of "La Mercè 2014", which was the 24th of September. During that day, 7 events happened in the city of Barcelona: a music concert at *Bogatell* beach area and its revival the morning after, human towers exhibition at *Plaça Sant Jaume*, open day at *MACBA* museum, a food market at *Parc de la Ciutadella*, a wine tasting fair at *Arc de Triomf* and fireworks near *Plaça d'Espanya*.

Together with Tweet-SCAN and WARBLE, we will also consider McInerney & Blei model (McInerney and Blei, 2014) and two WARBLE variants for comparison. Next, we enumerate event detection techniques under assessment,

(A) McInerney & Blei model (McInerney and Blei, 2014), which does not consider background and does not perform simultaneous topic-event learning.
(B) The WARBLE model without simultaneous topic-event learning.
(C) The WARBLE model without modeling background.
(D) The WARBLE model.
(E) Tweet-SCAN.

For McInerney & Blei, WARBLE and its variants we consider the number of components K to be 8 so that the model is able to capture the 7 events occurring. Moreover, we also consider the number of topics T to be 30 for all models. Regarding those models that do not perform simultaneous topic-event learning (B and E), the Latent Dirichlet Allocation model (Blei et al, 2003) is separately trained with tweets aggregated by key terms as proposed earlier in section 4.1.3.

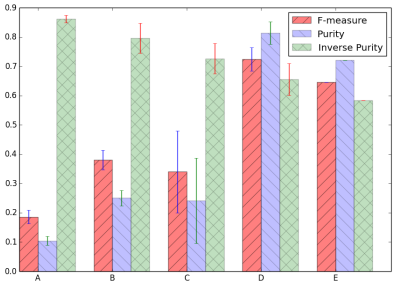


Fig. 10: Detection performance. (A) McInerney & Blei model (B) WARBLE w/o simultaneous topic-event learning (C) WARBLE w/o background model (D) WARBLE model (E) Tweet-SCAN

Fig. 10 shows the results for each event detection technique introduced earlier in terms of set matching metrics. Results show that the complete WARBLE model outperforms in terms of F-measure and purity. Moreover, by analyzing the results of

models B and C we see a clear synergy between background modeling and simultaneous topic-event learning. Neither of them separately achieves a large increase of the F-measure, but when combined they do.

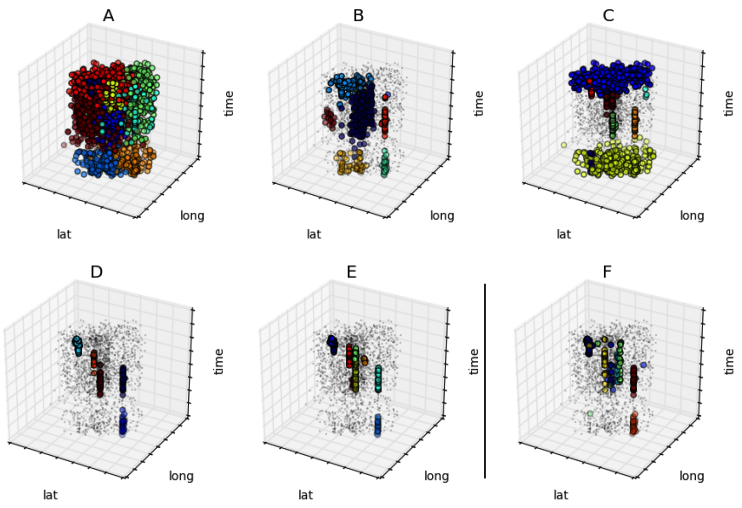


Fig. 11: Resulting real-world events. (A) McInerney & Blei model (B) WARBLE w/o simultaneous topic-event learning (C) WARBLE w/o background model (D) WARBLE model (E) Tweet-SCAN (F) Labeled events

Fig. 11 provides visual insight on the quality of the events detected by each of the alternatives, by drawing tweets in a 3-dimensional space corresponding to the spatial (lat, long) and temporal (time) features. Each tweet is colored with the maximum likelihood event assignment (c_n^*) for that tweet. Moreover, to improve visualization, the most populated cluster, which usually is the background, is plotted with tiny dots for all models, except model A, which fails to capture a clear background cluster. The figure shows that the similarity between hand-labeled data (F) and the WARBLE model (D) can only be compared to that of Tweet-SCAN (E).

6 Conclusions and Future Work

6.1 Conclusions

In this chapter, we have introduced the problem of event detection in location-based social networks and we have motivated a computational intelligent approach that combines probabilistic methods and learning theory to identify and characterize a set of interesting events from Twitter. Following this paradigm, we have presented a machine learning-based technique called WARBLE which is based on heterogeneous mixture models. To show the differences with the classical data mining approach,

we have also presented a DBSCAN-like algorithm for event detection called Tweet-SCAN. Both approaches are inspired on the anomaly-based event detection paradigm, in which events are groups of data points which are anomalous with respect to a baseline or background distribution.

On the one hand, the formulation of Tweet-SCAN within the framework of DBSCAN defines events as density-connected set of tweets in their spatial, temporal and textual dimension. This technique allows the discovery of arbitrary-shaped events, but restricts the definition of 'normality' to simply be sparse regions of tweets and has no notion of the data generation process. On the other hand, WARBLE can define richer background models and account for seasonality and uneven population densities, but the spatio-temporal shape for events is explicitly constrained to be Gaussian.

The experimental results show that both techniques performs similarly well, although WARBLE does slightly better. For Tweet-SCAN, we have also shown that the technique performs much better when incorporating the textual and user features. More importantly, we have shown that Tweet-SCAN and WARBLE significantly outperforms the geographical topic model presented by McInerney and Blei (2014). This result encourages explicitly modeling 'normality' in a separate clustering component, either in a data mining approach like DBSCAN or in probabilistic models like mixture models.

We have shown that both approaches can scale up to large data volumes by means of distributed processing frameworks such as Apache Spark. A parallel version of Tweet-SCAN splits data into separate partitions which might reside in separate computers and apply local event detection and subsequent merging to obtain the same results as the sequential Tweet-SCAN. The scaling of WARBLE benefits from stochastic optimization to avoid having all data cached in memory or in the same local machine. Moreover, general frameworks like Splash enable parallel and distributed learning on top of Apache Spark.

6.2 Future Work

Future work will put together larger Twitter datasets to corroborate our preliminary findings regarding the accuracy of both techniques and validate our approach to scale them up through the proposed parallel schemes and general purpose data processing engines, such as Apache Spark.

Moreover, we will consider non-parametric approaches for the proposed WARBLE model in which the number of events and topics can be automatically inferred from data. For instance, existing work in mixture models uses Dirichlet Process (Blei et al, 2006) as a prior distribution and that of topic modeling uses Hierarchically-nested Dirichlet process (Teh et al, 2006).

Probabilistic approaches to event detection also provide a mechanism to reason about unseen observations or partially observed data in a principled way. For example, posts that have not been geo-referenced, words that have been misspelled or pictures without captions, can be taken into account by these models.

Finally, online or prospective event detection has to be addressed in such a way that events can be detected as early and reliably as possible and deal with the fact that the 'normality' might change over time.

Our vision is that computational intelligent approaches that combine probabilistic modeling and learning theory can pave the way to build event detection systems which self-adapt to fast changing scenarios and that are capable to reason with partially observed and noisy data.

Acknowledgements This work is partially supported by Obra Social "la Caixa", by the Spanish Ministry of Science and Innovation under contract (TIN2015-65316), by the Severo Ochoa Program (SEV2015-0493), by SGR programs of the Catalan Government (2014-SGR-1051, 2014-SGR-118), Collectiveware (TIN2015-66863-C2-1-R) and BSC/UPC NVIDIA GPU Center of Excellence. We would also like to thank the reviewers for their constructive feedback.

References

Agarwal A, Duchi JC (2011) Distributed delayed stochastic optimization. In: Advances in Neural Information Processing Systems, pp 873–881

Akyildiz IF, Su W, Sankarasubramaniam Y, Cayirci E (2002) A survey on sensor networks. IEEE Communications Magazine 40(8):102–114, DOI 10.1109/MCOM.2002.1024422

Atefeh F, Khreich W (2015) A survey of techniques for event detection in twitter. Computational Intelligence 31(1):132–164

Banfield JD, Raftery AE (1993) Model-based gaussian and non-gaussian clustering. Biometrics pp 803–821

Bishop CM (2013) Model-based machine learning. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 371(1984):20120,222

Blei DM (2012) Probabilistic topic models. Communications of the ACM 55(4):77–84

Blei DM, Ng AY, Jordan MI (2003) Latent dirichlet allocation. the Journal of machine Learning research 3:993–1022

Blei DM, Jordan MI, et al (2006) Variational inference for dirichlet process mixtures. Bayesian analysis 1(1):121–144

Capdevila J, Cerquides J, Nin J, Torres J (2016a) Tweet-scan: An event discovery technique for geo-located tweets. Pattern Recognition Letters pp –

Capdevila J, Cerquides J, Torres J (2016b) Recognizing warblers: a probabilistic model for event detection in twitter, iCML Anomaly Detection Workshop

Capdevila J, Cerquides J, Torres J (2016c) Variational forms and updates for the WARBLE model. Tech. rep., `https://www.dropbox.com/s/0qyrkivpsxxv55v/report.pdf?dl=0`

Capdevila J, Pericacho G, Torres J, Cerquides J (2016d) Scaling dbscan-like algorithms for event detection systems in twitter. In: Proceedings of 16th International Conference, ICA3PP, Granada, Spain, December 14-16, 2016, Springer, vol 10048

Cea D, Nin J, Tous R, Torres J, Ayguadé E (2014) Towards the cloudification of the social networks analytics. In: Modeling Decisions for Artificial Intelligence, Springer, pp 192–203

Chandola V, Banerjee A, Kumar V (2009) Anomaly detection: A survey. ACM Comput Surv 41(3):15:1–15:58, DOI 10.1145/1541880.1541882, URL http://doi.acm.org/10.1145/1541880.1541882

Cordova I, Moh TS (2015) Dbscan on resilient distributed datasets. In: High Performance Computing Simulation (HPCS), 2015 International Conference on, pp 531–540, DOI 10.1109/HPCSim.2015.7237086

Endres DM, Schindelin JE (2003) A new metric for probability distributions. IEEE Transactions on Information theory

Escobar MD, West M (1995) Bayesian density estimation and inference using mixtures. Journal of the american statistical association 90(430):577–588

Ester M, Kriegel HP, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: Kdd, vol 96, pp 226–231

Fox CW, Roberts SJ (2012) A tutorial on variational Bayesian inference. Artificial Intelligence Review 38(2):85–95, DOI 10.1007/s10462-011-9236-8

Fraley C, Raftery AE (2002) Model-based clustering, discriminant analysis, and density estimation. Journal of the American statistical Association 97(458):611–631

He Y, Tan H, Luo W, Feng S, Fan J (2014a) Mr-dbscan: a scalable mapreduce-based dbscan algorithm for heavily skewed data. Frontiers of Computer Science 8(1):83–99, DOI 10.1007/s11704-013-3158-3, URL http://dx.doi.org/10.1007/s11704-013-3158-3

He Y, Tan H, Luo W, Feng S, Fan J (2014b) Mr-dbscan: a scalable mapreduce-based dbscan algorithm for heavily skewed data. Frontiers of Computer Science 8(1):83–99, DOI 10.1007/s11704-013-3158-3, URL http://dx.doi.org/10.1007/s11704-013-3158-3

Hoffman M, Bach FR, Blei DM (2010) Online learning for latent dirichlet allocation. In: advances in neural information processing systems, pp 856–864

Hoffman MD, Blei DM, Wang C, Paisley J (2013) Stochastic variational inference. The Journal of Machine Learning Research 14(1):1303–1347

Hong L, Davison BD (2010) Empirical study of topic modeling in twitter. In: Proceedings of the First Workshop on Social Media Analytics, ACM, pp 80–88

Jordan MI, Ghahramani Z, Jaakkola TS, Saul LK (1999) An introduction to variational methods for graphical models. Machine learning 37(2):183–233

Koller D, Friedman N (2009) Probabilistic graphical models: principles and techniques. MIT press

Li L, Goodchild MF, Xu B (2013) Spatial, temporal, and socioeconomic patterns in the use of twitter and flickr. Cartography and Geographic Information Science 40(2):61–77

McInerney J, Blei DM (2014) Discovering newsworthy tweets with a geographical topic model. NewsKDD: Data Science for News Publishing workshop Workshop in conjunction with KDD2014 the 20th ACM SIGKDD Conference on Knowledge Discovery and Data Mining

McMinn AJ, Moshfeghi Y, Jose JM (2013) Building a large-scale corpus for evaluating event detection on twitter. In: Proceedings of the 22nd ACM international

conference on Information & Knowledge Management, ACM, pp 409–418
Murphy KP (2012) Machine learning: a probabilistic perspective. MIT press
Newman D, Smyth P, Welling M, Asuncion AU (2007) Distributed inference for latent dirichlet allocation. In: Advances in neural information processing systems, pp 1081–1088
Newman N (2011) Mainstream media and the distribution of news in the age of social discovery. Reuters Institute for the Study of Journalism, University of Oxford
Panagiotou N, Katakis I, Gunopulos D (2016) Detecting events in online social networks: Definitions, trends and challenges. Solving Large Scale Learning Tasks: Challenges and Algorithms
Quan X, Kit C, Ge Y, Pan SJ (2015) Short and sparse text topic modeling via self-aggregation. In: Proceedings of the 24th International Conference on Artificial Intelligence, AAAI Press, pp 2270–2276
Rajasegarar S, Leckie C, Palaniswami M (2008) Anomaly detection in wireless sensor networks. IEEE Wireless Communications 15(4):34–40, DOI 10.1109/MWC.2008.4599219
Salton G, Buckley C (1988) Term-weighting approaches in automatic text retrieval. Inf Process Manage 24(5):513–523
Salton G, Wong A, Yang CS (1975) A vector space model for automatic indexing. Communications of the ACM 18(11):613–620
Sander J, Ester M, Kriegel HP, Xu X (1998) Density-based clustering in spatial databases: The algorithm gdbscan and its applications. Data Mining and Knowledge Discovery 2(2):169–194
Stelter B, Cohen N (2008) Citizen journalists provided glimpses of mumbai attacks. URL `http://www.nytimes.com/2008/11/30/world/asia/30twitter.html`
Teh YW, Jordan MI, Beal MJ, Blei DM (2006) Hierarchical dirichlet processes. Journal of the american statistical association 101(476)
Van Rijsbergen CJ (1974) Foundation of evaluation. Journal of Documentation 30(4):365–373
Wong WK, Neill DB (2009) Tutorial on event detection. In: KDD
Yang Y, Pierce T, Carbonell J (1998) A study of retrospective and on-line event detection. In: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, ACM, pp 28–36
Yuan Q, Cong G, Ma Z, Sun A, Thalmann NM (2013) 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, ACM, pp 605–613
Zaharia M, Chowdhury M, Franklin MJ, Shenker S, Stoica I (2010) Spark: cluster computing with working sets. In: Proceedings of the 2nd USENIX conference on Hot topics in cloud computing, vol 10, p 10
Zhang Y, Jordan MI (2015) Splash: User-friendly programming interface for parallelizing stochastic algorithms. arXiv preprint arXiv:150607552
Zheng Y (2012) Tutorial on location-based social networks. In: Proceedings of the 21st international conference on World wide web, WWW, ACM