A Bayesian Graphical Model to Discover Latent Events from Twitter

Wei Wei
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, PA, USA
weiwei@cs.cmu.edu

Kenneth Joseph
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, PA, USA
kjoseph@cs.cmu.edu

Wei Lo
Zhejiang University
38 Zheda Rd
Hangzhou, Zhejiang, China
spencer_w_lo@zju.edu.cn

Kathleen M. Carley
Carnegie Mellon University
5000 Forbes Ave
Pittsburgh, PA, USA
kathleen.carley@cs.cmu.edu

Abstract

Online social networks like Twitter and Facebook produce an overwhelming amount of information every day. However, research suggests that much of this content focuses on a reasonably sized set of ongoing events or topics that are both temporally and geographically situated. These patterns are especially observable when the data that is generated contains geospatial information, usually generated by a location-enabled device such as a smartphone. In this paper, we consider a data set of 1.4 million geo-tagged tweets from a country during a large social movement, where social events and demonstrations occurred frequently. We use a probabilistic graphical model to discover these events within the data in a way that informs us of their spatial, temporal and topical focus. Quantitative analysis suggests that the streaming algorithm proposed in the paper uncovers both well-known events and lesser-known but important events that occurred within the timeframe of the dataset. In addition, the model can be used to predict the location and time of texts that do not have these pieces of information, which accounts for the much of the data on the web.

1 Introduction

The perpetual availability of online content and our increasing reliance on the Internet have made social networking websites such as Twitter and Facebook an indispensable part of modern social life for many people. As of November 2014, it is estimated that roughly a half billion tweets are generated on a daily basis [1]. The content generated from these social networking/social media sites is not only voluminous; it also contains a selection of information that is new and interesting to individual users, corporate and government actors and researchers alike. This information is useful for many types of analysis, such as sentiment analysis (Pak and Paroubek 2010) and abnormality detection (Thom et al. 2012).

One particularly interesting line of work that draws on social media content is the problem of detecting events. In event detection, we wish to uncover abnormal subsets of content that may be referring to a particular occurrence of interest. A significant amount of this work focuses purely on the analysis of the textual content of social media messages (Benson, Haghighi, and Barzilay 2011; Kumaran and Allan 2004). While the inference of topical focus is an interesting problem in its own right, the idea that topical coherence is a signal for an "event" is slightly misleading. Such algorithms are essentially detecting topics, which are words that clustered together, rather than any coherent subset of content that has a unique geo-temporal realization, one we would expect of a typical event. For example, topics uncovered that are broadly related to online games and jokes have little or no link to the physical world and thus are difficult to consider events.

Having realized this, recent work has begun to focus on the geo-temporal aspects of event detection (Sakaki, Okazaki, and Matsuo 2010). However, much of this work fails to utilize the textual information that previous authors have capitalized on, information that is vital in interpreting the topical focus of a particular event (Ritter, Etzioni, and Clark 2012). For example, events that occur in a residence and a nearby night club at the same time will contain the same geospatial and temporal information but are, of course, different in important ways. A good definition of an event should thus contain a geographical approximation of where the event is happening, a temporal range over which the event lasts and also a specific set of words and/or phrases that can be used to describe what the event is about.

In this paper, we develop a probabilistic graphical model that learns the existence of events based on the location, time and text of a set of social media posts, specifically tweets. An event is described by a central geographical location and time, a variance in space and time and a set of words (a topic) that is representative of the terms that can be used to describe this event. By incorporating both a central location and time and a variance around it, we account for the fact some events are more concentrated within a specific region and time (e.g. a marathon) while others might be distributed across a broader area in time and/or space (e.g. Occupy Wallstreet). The use of a set of words that are frequently used in tweets from or about the event allows us to incorporate topic modeling to extract information from the actual tweet text, from which an understanding of the focus of the event can be derived.

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

Our contributions are twofold. First, we build an event detection model that successfully discovers latent events being discussed at different points in time and at different locations in a large, geo-tagged Twitter data set. We demonstrate the model's abilities by applying our method to a Twitter data set collected in an Arab country during a time period where demonstrations and social movements were frequent. Second, we build a location and time prediction tool based on our learned model that allows us to accurately predict the location or time of a tweet (when this information is held out) with considerably more accuracy than several baseline approaches.

2 Related Work

The problem of event detection is well studied. Here, we provide a brief survey of relevant methods, touching on a variety of approaches that have been taken in studying the problem.

Events Extraction from Text

As most information available on the web does not provide geospatial or temporal information, text based methods represent an important aspect of event detection methodology. Three general types of approaches are surveyed here.

Clustering is one of most important techniques in dealing with the event detection problem using text. Clustering approaches attempt to find latent events by uncovering common patterns of texts that appear in the document set. These efforts generally fall into two distinct types of approaches: similarity based methods and statistical ones. Similarity-based methods usually compare documents by applying metrics such as cosine similarity (Kumaran and Allan 2004). These models are usually efficient but ignore statistical dependencies between both observable and latent underlying variables. A statistical method such as a graphical model (Benson, Haghighi, and Barzilay 2011) can incorporate more complicated variable dependencies and hierarchical structure to event inference.

Another type of event detection model utilizes the fact that the arrival of new events will change the distribution of the existing data. Such approaches are thus concerned with developing criterion for detecting abnormal changes in the data. For example, Matuszka, Vinceller, and Laki (2013) assumes a life cycle for each possible keyword for an event, penalizing the term if it appears consistently in the data. The result is an event defined by keywords that only appear in some specific subset of the observed data. Zubiaga et al. (2012) use techniques such as outlier detection to detect abnormalities in the data set which is considered a potential consequence of a new event.

The third type of work defines events indirectly by linking documents together. Models such as the one proposed by Štajner and Grobelnik (2009) define each document as a node in a graph and then build connections between them once they are classified as being a part of the same event. Finally, there is also a large amount of work focusing on using information retrieval techniques such as TF-IDF as features to extract events(Brants, Chen, and Farahat 2003).

Events Extractions from Space and Time

Beyond the extraction of events purely from text, there have also been several efforts to incorporate temporal and geospatial information. Sakaki, Okazaki, and Matsuo (2010) analyzed the statistical correlations between earthquake events in Japan and Twitter messages that were sent during the distaster time frame. An abrupt change of volume of tweets in a specific geo region indicated a potential disaster in that area. Hong et al. (2012) constructed a probabilistic graphical model that contains both a geographical component and a topical component to discover latent regions from Twitter data. Their efforts, however, are not strictly focused on event detection, as they do not consider the temporal domain. In contrast, Ritter, Etzioni, and Clark (2012) and Panisson et al. (2014) extract events into a hierarchy of types, in part utilizing the temporal information in both the text and the timestamp of the tweet itself. However, their work does not consider the spatial information explicit in geospatially tagged tweets.

Graphical Models and Sampling Techniques

Graphical models are powerful tools that can be used to model and estimate complex statistical dependencies among variables. For a general overview, we refer the reader to (Jordan 1998), which contains a much richer discussion than is possible here. By constructing statistical dependencies among both observed and latent variables, graphical models can be used to infer latent representations that are not observed in the data. Latent Dirichlet allocation (Blei, Ng, and Jordan 2003), used to discover such latent topics/events from text, is perhaps the most widely known example in this area.

One issue often raised in graphical models is the difficulty in estimation. As the complexity of the model increases, exact inference become difficult or even impossible. Various sampling strategies such as Gibbs sampling (Casella and George 1992) has thus been developed to find approximate solutions.

3 Model

We use a probabilistic graphical model to characterize the relationship between events and tweets (referred to here as documents). Using plate notation, Figure 1 illustrates the structure of the model. Note that there are D documents and E events, where E is a value pre-determined by the researcher. The model has three major components. First, an **event model** contains information about a specific event, such as the parameters that characterize its spatial and temporal distributions. Second, a **document model** contains the location, time and event index of each document. Third, there is a **language model**, which contains information about the topical content of the documents. Table 1 gives a summary of all notation that will be used as we describe the model in this section.

Event Model

An important observation incorporated into our model is that events are in many ways natural extensions of topics; events

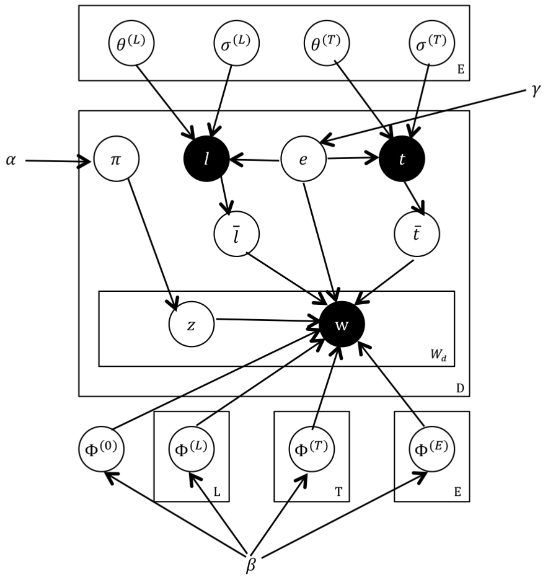

Figure 1: Illustrations of the model in plate notations

have a topical focus but also include a spatial and temporal region in which they are likely to occur. We thus assume events are defined by three things. First, each event has a geographical center $\theta_e^{(L)}$ as well as a geographical variance controled by a diagonal covariance matrix with each value defined by $\sigma_e^{(L)}$. The location of a document that belongs to event e is assumed to be drawn from a two dimensional Gaussian distribution governed by these parameters.

$$l{\sim}\mathrm{N}(\theta_e^{(L)}, I{\cdot}\sigma_e^{(L)}) \tag{1}$$

Second, each event is defined by a temporal domain. Similar to the spatial distribution of an event, event time is also modeled as a Gaussian distribution, except with mean $\theta_e^{(T)}$ and a variance of $\sigma_e^{(T)}$:

$$t{\sim}\mathrm{N}(\theta_e^{(T)}, I{\cdot}\sigma_e^{(T)}) \tag{2}$$

The mean and standard deviations of both Gaussian distributions are latent variables and will need to be inferred by the model. Finally, events are determined by a topic (or distribution over words) that characterizes the event. The details of this are implemented within the document model and Language model, discussed later in this section.

Document Model

A document contains the information we obtain for a specific tweet. In our model, we only consider tweets that have both a geo-location tag (latitude/longitude pair) l and a time stamp t. Tweets also consist of a word array w which contains the actual words that appear in the tweet.

Several latent variables are also present in the document model. First, an event identity e defines which single event out of the E possible events in the event model that this specific document belongs to. We assume a multinomial prior γ for each e in each document.

$$e \sim \mathrm{Mult}(\gamma) \tag{3}$$

Second, each word w_i in the document has a corresponding category variable z_i that determines which of 4 categories of topics this word has been drawn from. Category "0" is a global category, which represents global topics that frequently occur across all tweets. Category "L" defines a set of regionally specific topics that are specific to particular geospatial subareas within the data. Category "T" represents a set of temporally aligned topics that contain words occurring within different temporal factions of the data. Category "E" defines topics that are representative of a particular event e, distinct from both other events and more specific to the event than topics in the other categories. By controlling for global, temporal and spatial topics, these event-specific topics allow us to uncover the defining terms of this particular event beyond those specific to a general spatial or temporal region. The variable z is controlled by a multinomial distribution whose parameter is a per document category distribution π:

$$z{\sim}\mathrm{Mult}(\pi) \tag{4}$$

For each document a π is generated by a prior α from a Dirichlet distribution:

$$\pi{\sim}\mathrm{Dir}(\alpha) \tag{5}$$

To index into the topics of the location and time categories, each location l and time t is converted into a location index $\bar{l}$ and a time index $\bar{t}$, respectively. These conversions are conducted by applying two functions $f(l)$ and $g(t)$. These resulting indices are used for the language model to retrieve the corresponding topics from these categories in a manner that will be introduced later.

$$\bar{l} = f(l) \tag{6}$$

$$\bar{t} = g(t) \tag{7}$$

Language Model

The language model defines how words within a document are drawn from topics (within specific categories) based on the full set of parameters associated with the document. Topic distributions for each category are generated using a Dirichlet prior β:

$$\Phi_*^{(*)}{\sim}\mathrm{Dir}(\beta) \tag{8}$$

Each topic contains the probability of each word in the vocabulary occurring within it. While this is the traditional representation of LDA, note that our approach is a generalization of the original model (Blei, Ng, and Jordan 2003), since now topics are also hierarchically organized by the four different categories. For a model with one global topic, L location topics, T time topics and E event topics, the total number of topics across the four categories is thus $K = 1 + L + T + E$.

Each word w_i is chosen from a corresponding topic based on its category variable z and the corresponding geo, temporal and event indices $\bar{l}$, $\bar{t}$ and e, respectively, depending on which category is being used. This is represented mathematically in Equation 9 below:

$$\begin{aligned} &P(w_i|\bar{l},\bar{t},e,z_i,\Phi^{(0)},\Phi^{(L)},\Phi^{(T)},\Phi^{(E)}) \\ &\quad = P(w_i|\Phi^{(0)})^{I(z_i=0)}\cdot P(w_i|\Phi^{(L)},\bar{l})^{I(z_i=L)}\cdot \\ &\qquad P(w_i|\Phi^{(T)},\bar{t})^{I(z_i=T)}\cdot P(w_i|\Phi^{(E)},e)^{I(z_i=E)} \end{aligned} \tag{9}$$

Spatial and Temporal Boundaries

To generate the location index (i.e. $\bar{l}$) and time index (i.e. $\bar{t}$), we need to define two transformation functions that map from a real vector space to an integer space. To do so, we first divide the geographical and temporal space into a lattice within a pre-determined boundary. For geospace, a preset boundary $B^L = (x_{low}, x_{hi}, y_{low}, y_{hi})$ is determined based on the data. The geoarea is then divided evenly by the number of locations L to form a $\sqrt{L} \times \sqrt{L}$ square lattice. Each cell in the lattice has a unit length of $U^L = (x, y)$, with $U^L_x = (B^L_{x_{hi}} - B^L_{x_{low}})/\sqrt{L}$ and $U^L_y = (B^L_{y_{hi}} - B^L_{y_{low}})/\sqrt{L}$ respectively. The transformation function for location data $f(l)$ is then defined in Equation10:

$$f(l) = \lfloor (l_x - B^L_{x_{low}})/U^L_x \rfloor * \sqrt{L} + \lfloor (l_y - B^L_{y_{low}})/U^L_x \rfloor \tag{10}$$

Similar to the way that l is mapped to $\bar{l}$, a function that maps t into an index space $\bar{t}$ is also defined in equation 11. Here we treat t as a real valued scalar bounded in range from B^T_x to B^T_y. A unit length U^T is also calculated to be the unit length of each time cell in the lattice, which is $(B^T_{hi} - B^T_{low})/T$.

$$g(t) = \lfloor (l - B^T_{low})/U^L \rfloor \tag{11}$$

In our model we treat the timestamp of a document as a real-valued variable by dividing the UNIX time by the number of seconds in a month. By doing this we converted the information so that tweets are represented by a real-valued variable that defines the month and year in which they occur. This meets the requirement of the Gaussian distribution in which we used to model the temporal span of a particular event.

Generative Model

The graphical model we defined above can be used as a generative model that produces new tweets that have a geo coordinate, a time stamp and a set of words constituting the text of the message. The generative process is as follows:

- Pick an event $e \sim \mathrm{Mult}(\gamma)$.
- Pick a location $l \sim \mathrm{N}(\theta^{(L)}_e, \sigma^{(L)}_e)$
- Pick a time $t \sim \mathrm{N}(\theta^{(T)}_e, \sigma^{(T)}_e)$
- Pick a category distribution $\pi \sim \mathrm{Dir}(\alpha)$
- For each word w_i, first pick $z_i \sim \mathrm{Mult}(\pi)$ then pick $w_i \sim \Phi^{(*)}$

4 Model Inference

Given the number of hidden variables as well as the hierarchical structure of the model, exact inference is intractable. Instead, we use a Gibbs-EM algorithm (Andrieu et al. 2003; Wallach 2006) to infer the model parameters. Before we detail the inference procedure, we clarify three pieces of notation, O, Ω and Θ, that define the sets of variables we are concerned with during the inference procedure. The set $O = \{l, t, w\}$ defines the set of observed variables. The set $\Omega = \{e, z, \pi, \Phi^{(0)}, \Phi^{(L)}, \Phi^{(T)}, \Phi^{(E)}\}$ defines variables that will be solved during the E stage of the algorithm. Variables falling into this set are mainly those related to the language model. The variable $\Theta = \{\theta_L, \theta_T, \sigma_L, \sigma_T\}$ is a set of parameters that will be estimated during the M step. Note that we do not perform inference on the Bayesian hyper parameters $\{\alpha, \beta, \gamma\}$, treating them as static constants to be defined by the researcher. To avoid confusions, we have omitted all the Bayesian hyper parameters in our equations and we will follow this convention in the rest of the paper.

Table 1: Notations

Symbol	Size	Comments
D	1	number of documents
L	1	number of location plates
T	1	number of time plates
E	1	number of events
Z	1	number of topic categories
K	1	number of topics
V	1	number of vocabularies
W_d	1	number of words in document
l	D ×2	location lat and lon
t	D	timestamps
e	D	event index
w	W_d	word in a document
$\bar{l}$	D	location index of a document
$\bar{t}$	D	time index of a document
$\theta^{(L)}, \sigma^{(L)}$	E	mean and sd of event locations
$\theta^{(T)}, \sigma^{(T)}$	E	mean and sd of event time
z	W_d	topic category of word
π	D × Z	category distribution
Φ	K × V	word distribution for topics
α	Z	dirichlet prior for π
β	V	dirichlet prior for Φ
γ	E	multinomial prior for e
O	-	Observed variables
Ω	-	latent variables solved in E step
Θ	-	latent variables solved in M step

E Step

During the Expectation ("E") step, we assume that parameters in Θ are already known as the result of a previous Maximization ("M") step. We then use Gibbs sampling to generate samples for the parameters in Ω over a number of Gibbs iterations and use the average of these samples to approximate the expectation of the E step. Before we do this, however, we first integrate out $\Phi^{(*)}$ and π, resulting in a more efficient collapsed Gibbs sampling problem. Equation 12 gives the collapsed distribution we are interested in sampling from. Here Γ is the gamma function and $n^{z,k}_{d,r}$ denotes the number of times that a document d has a word r that falls into topic k of category z. If any of d, r, k or z are replaced by “*”, the value should be interpreted as one which takes the sum over this particular variable. Note again that in contrast to the standard LDA model, here we need to pay attention to both topic k and the category z.

$$
\begin{aligned}
P(z, e|\Theta, O) = & \int_{\phi^{(0)}} \int_{\phi^{(L)}} \int_{\phi^{(T)}} \int_{\phi^{(E)}} \int_{\pi} \\
& P(z, e, \pi, \Phi^{(0)}, \Phi^{(L)}, \Phi^{(T)}, \Phi^{(E)}|\Theta, O) \\
= & \prod_{d=1}^{D} \frac{\Gamma(\sum_{z=1}^{Z} \alpha_z)}{\prod_{z=1}^{Z} \Gamma(\alpha_z)} \frac{\prod_{z=1}^{Z} \Gamma(n_{d,*}^{z,*} + \alpha_z)}{\Gamma(\sum_{z=1}^{Z} n_{d,*}^{z,*} + \alpha_z)} \times \\
& \frac{\Gamma(\sum_{r=1}^{V} \beta_r)}{\sum_{r=1}^{V} \Gamma(\beta_r)} \frac{\sum_{r=1}^{V} \Gamma(n_{*,r}^{0,1} + \beta_r)}{\Gamma(\sum_{r=1}^{V} n_{*,r}^{0,1} + \beta_r)} \times \\
& \prod_{\bar{l}=1}^{L} \frac{\Gamma(\sum_{r=1}^{V} \beta_r)}{\sum_{r=1}^{V} \Gamma(\beta_r)} \frac{\sum_{r=1}^{V} \Gamma(n_{*,r}^{L,\bar{l}} + \beta_r)}{\Gamma(\sum_{r=1}^{V} n_{*,r}^{L,\bar{l}} + \beta_r)} \times \\
& \prod_{\bar{t}=1}^{T} \frac{\Gamma(\sum_{r=1}^{V} \beta_r)}{\sum_{r=1}^{V} \Gamma(\beta_r)} \frac{\sum_{r=1}^{V} \Gamma(n_{*,r}^{T,\bar{t}} + \beta_r)}{\Gamma(\sum_{r=1}^{V} n_{*,r}^{T,\bar{t}} + \beta_r)} \times \\
& \prod_{e=1}^{E} \frac{\Gamma(\sum_{r=1}^{V} \beta_r)}{\sum_{r=1}^{V} \Gamma(\beta_r)} \frac{\sum_{r=1}^{V} \Gamma(n_{*,r}^{E,e} + \beta_r)}{\Gamma(\sum_{r=1}^{V} n_{*,r}^{E,e} + \beta_r)}
\end{aligned} \tag{12}
$$

Word Category The word category variable z is sampled for each word in each document. The conditional probability of a specific category for word n in document d given all the other variables is proportional to the conditional probability given in Equation 13. While space constraints do not allow us to present the full derivation of the conditional probability, ideas utilized in the proofs of the original LDA algorithm in Griffiths and Steyvers (2004) can be directly applied to our efforts to derive the equation.

$$
\begin{aligned}
& P(z_{(d,n)} = z)|z_{\text{-}(d,n)}, \Theta, O) \\
& \quad \propto P(z_{(d,n)} = z), w_{\text{-}(d,n)}, w|\Theta, O) \\
& \quad \propto (n_{d,*}^{z,*-(d,n)} + \alpha_k) \frac{n_{*,r}^{z,*-(d,n)} + \beta_r}{\sum_{r=1}^{V} n_{*,r}^{z,*-(d,n)} + \beta_r}
\end{aligned} \tag{13}
$$

Category and Word Distribution After the category variable z is sampled for each word in each document in the data, we update all word distributions $\Phi^{(*)}$ as well as the category distribution π for each document according to Equation 14 and Equation 15. Again, while proofs are omitted, similar proofs can be found in Griffiths and Steyvers (2004). One thing worth noticing, however, is that $\pi_{d,z}$ is a bit different from its counterpart $\theta_{d,k}$ in the classic LDA model because of the second dimension k, which is a topic index in the classic LDA. In the present model, this value is changed to z, thus representing a draw from a category rather than a topic.

$$
\Phi_{k,v}^{(i)} = \frac{n_{*,v}^{i,k} + \beta_v}{\sum_{v=1}^{V} n_{*,v}^{i,k} + \beta_v} \tag{14}
$$

$$
\pi_{d,z} = \frac{n_{d,*}^{z,*} + \alpha_z}{\sum_{d=1}^{D} n_{d,*}^{z,*} + \alpha_z} \tag{15}
$$

Event Index In addition to sampling the category variables and distributions over the categories, we also must sample the event index e for each document d. The conditional probability for sampling the event index for a specific document based on all other variables is given in Equation 16. It is determined by three terms: a prior multinomial distribution on e, two Gaussian distributions, one each on location and time, and a term defining the joint likelihood of each word in the tweet. Observing that this expression can be further simplified and only those words w_i with $z_{w_i} = E$ are actually affecting the probability of sampling e, we are left with Equation 16

$$
\begin{aligned}
& P(e_d|\Omega \backslash e_d, \Theta, O) \\
& \quad \propto \prod_{i;z_i=e} P(w_i|z_i, \Phi^{(z_i)}) \cdot P(l|\theta_e^{(L)}, \sigma_e^{(L)}) \cdot \\
& \quad\quad P(t|\theta_e^{(T)}, \sigma_e^{(T)}) \cdot P(e_d|\gamma) \\
& \quad \propto \frac{1}{\sigma_e^{(L)} \sigma_e^{(T)}} \cdot \gamma(E = e) \cdot \prod_{i;z_i=e} \Phi^{(e)}(w = w_i) \cdot \\
& \quad\quad e^{-\frac{1}{2}[\frac{(L-\theta_e^{(L)})^T (L-\theta_e^{(L)})}{\sigma_e^{(L)2}} + \frac{(T-\theta_e^{(T)})^T (T-\theta_e^{(T)})}{\sigma_e^{(T)2}}]}
\end{aligned} \tag{16}
$$

M step

In the M step, we treat all the variables in Θ as parameters and estimate them by maximizing the likelihood function. Since we use Gibbs sampling in the E step, the likelihood function is an average over all samples drawn from the E step.

For each Gibbs step s we use a superscript to annotate the variables that are drawn from this specific step. The objective function of the M step $Q(\Theta)$ can be written in Equation 17. The goal of this M step is to find the latent variables in Θ that maximize this objective function. To achieve better optimization results, we add an L2 penalty term to the location and time deviations in our objective function in addition to the log likelihood. The penalty term has a factor $(1 + r_e)$, where r_e is the ratio of documents that belong to event e. If the ratio r_e for a specific event is high, it will receive a stronger penalty in the size of its spatial and temporal deviations, causing these variances to be restricted.

$$
\begin{aligned}
Q(\Theta) = & \frac{1}{S} \sum_{s=1}^{S} log(P(O, \Omega^{(s)}|\Theta^{(t)})) \\
& + \frac{1}{2} \lambda((||\sigma^{(L)}||_2^2 + ||\sigma^{(T)}||_2^2)(1 + r_e)) \\
\propto & \frac{1}{S} \sum_{s=1}^{S} \sum_{d=1}^{D} \Big[- (log(\sigma_{e_d^{(s)}}^{(L)}) + log(\sigma_{e_d^{(s)}}^{(T)})) \\
& - 0.5 (\frac{||l_d - \theta_{e_d^{(s)}}^{(L)}||}{\sigma_{e_d^{(s)}}^{(L)}{}^2} + \frac{||t_d - \theta_{e_d^{(s)}}^{(T)}||}{\sigma_{e_d^{(s)}}^{(T)}{}^2}) \Big] \\
& - \frac{1}{2} \lambda((||\sigma^{(L)}||_2^2 + ||\sigma^{(T)}||_2^2)(1 + r_e))
\end{aligned} \tag{17}
$$

Event Centers Event centers for both location and time can be estimated in a straightforward manner by maximizing the objective function.

$$\theta_e^{\hat{(L)}} = \frac{\sum_s \sum_{d;e_d^{(s)}=e} l_d}{\sum_s \sum_{d;e_d^{(s)}=e}} \tag{18}$$

Similarly, we can also acquire a MLE estimation for $\theta_e^{\hat{(T)}}$:

$$\theta_e^{\hat{(T)}} = \frac{\sum_s \sum_{d;e_d^{(s)}=e} t_d}{\sum_s \sum_{d;e_d^{(s)}=e}} \tag{19}$$

Event Variance In the estimation of the variance in space and time for each event, the penalty term we have introduced means that we can no longer use the MLE to find an optimal value for them. While this complicates inference, the penalty term is an important part of the model. It is introduced because in model development, we observed that as the number of EM steps increased, larger events tended to rapidly acquire more documents during training. This, in turn, increases the variance of these events to a value larger than we would expect to see for a spatially constraint event. This situation becomes worse over time and eventually these events come to dominate the analysis. The introduced L2 penalty restricts this from occurring.

To solve for the variances, we use a gradient descent approach to find the optimal value. In order to do so, we take the derivative of the EM objective function and acquire the gradient of the event deviations in Equation 20 and Equation 21. We then apply a standard gradient descent algorithm.

$$\frac{\partial Q(\Theta)}{\partial \sigma_e^{(L)}} = \frac{\sum_s \sum_{d:e_d=e} \frac{-1}{\sigma_e^{(L)}} + \frac{||l-\theta_e^{(L)}||}{\sigma_e^{(L)3}} - \lambda \sigma_e^{(L)}(1+r_e)}{S} \tag{20}$$

$$\frac{\partial Q(\Theta)}{\partial \sigma_e^{(T)}} = \frac{\sum_s \sum_{d:e_d=e} \frac{-1}{\sigma_e^{(T)}} + \frac{||t-\theta_e^{(T)}||}{\sigma_e^{(T)3}} - \lambda \sigma_e^{(T)}(1+r_e)}{S} \tag{21}$$

Initializations Several variables need to be properly initialized in order for the EM algorithm to converge to the correct distribution. The parameters z and e are initialized randomly within their domains. The variables $\theta^{(L)}$ and $\theta^{(T)}$ are initialized by learning a kernel density estimator from the data first and then drawing e samples from it. This initialization gives areas in space and time where tweets are concentrated a higher chance of becoming centers in location or time, respectively. Finally, the variables $\sigma^{(L)}$ and $\sigma^{(T)}$ are generated from a uniform distribution from 0 to 1.

Prediction

One of the most important applications of the model proposed in the paper is to predict the location and time of tweets based on the words contained within them. To achieve this goal, we use another EM algorithm again to infer the hidden variables as well as the variable(s) we are interested in predicting. In the prediction setting, event specific parameters θ and σ and topic categories $\Phi^{(*)}$ are already trained and our goal is to infer z,e and either l, t or w given some or all of the other variables.

Category Variable and Event Index In our prediction EM algorithm, we estimate the category variable z and the event index e in the E step. This is almost the same process as the one in the training, as all other variables are again fixed. The only difference is that during the training stage, $n_{d,i}^{z,k}$ is initialized according to a randomly generated z and e while in the prediction stage these variables are the result of a trained model.

Predict Location and Time To predict location and time, we use the samples generated from the E step to make a point inference on one or both, depending on the task at hand. As opposed to the M step in the training stage, in our prediction task all event variables have already been learned and our goal is to estimate l and t instead. Equation 22 is the objective function for both l and t. Utilizing the fact that the addition of several Gaussian distributions is proportional to another Gaussian distribution, the summation term for the location and time distributions can each be absorbed into a single Gaussian distribution. The part of the likelihood function that contains the summation of word probabilities can also be simplified to consider only those words with topics related to either L or T. This results in an objective function that has a location component and a time component, each of which contains a Gaussian term and a grid density term.

$$\begin{aligned} Q'(l,t) \propto \frac{1}{S}\sum_{s=1}^{S}[&logP(l|\theta_{e^{(s)}}^{(L)}, \sigma_{e^{(s)}}^{(L)}) + logP(t|\theta_{e^{(s)}}^{(T)}, \sigma_{e^{(s)}}^{(T)}) \\ &+ \sum_{i}^{W} logP(w_i|\Phi^{(z_i)}, \bar{l}, \bar{t}, e^{(s)})] \\ \propto\ & logP(l|\theta_*^{(L)}, \sigma_*^{(L)}) + \frac{1}{S}\sum_{s=1}^{S}\sum_{i;z_i=L} log\Phi_{\bar{l}}^{(L)}(w_i) \\ &+ \underbrace{logP(t|\theta_*^{(T)}, \sigma_*^{(T)})}_{\text{Gaussian Term}} + \underbrace{\frac{1}{S}\sum_{s=1}^{S}\sum_{i;z_i=T} log\Phi_{\bar{t}}^{(T)}(w_i)}_{\text{Grid Density Term}} \\ \text{Where}\quad & \theta_*^{(L)} = \frac{\sum_s \frac{\theta_{e(s)}^{(L)}}{\sigma_{e(s)}^{(L)\ 2}}}{\sum_s \frac{1}{\sigma_{e(s)}^{(L)\ 2}}}, \quad \theta_*^{(T)} = \frac{\sum_s \frac{\theta_{e(s)}^{(T)}}{\sigma_{e(s)}^{(T)\ 2}}}{\sum_s \frac{1}{\sigma_{e(s)}^{(T)\ 2}}}, \\ \sigma_*^{(L)^2} &= \frac{S}{\sum_s \frac{1}{\sigma_{e(s)}^{(L)\ 2}}} \quad \text{and} \quad \sigma_*^{(T)^2} = \frac{S}{\sum_s \frac{1}{\sigma_{e(s)}^{(T)\ 2}}} \end{aligned} \tag{22}$$

Speeding Up the Optimization From Equation 22, we observe that the estimation of l and t can be done independently, as the objective functions of each entity are absolved of terms from the other. However, to infer either l or t based on the objective function is difficult using conventional optimization methods such as gradient descent since it involves optimizing an objective function that is not continuous. This occurs because the transformation from l and t to $\bar{l}$ and $\bar{t}$ makes the objective function no longer differentiable. Search based optimization techniques can still be applied but are exceedingly slow.

We thus develop a method particular to our specific issue that can estimate l and t rapidly. To see how we can speed up the optimization, observe that the grid density term in Equation 22 is fixed when variables fall within a single grid cell. For example for all l such that $\bar{l}$ are the same, these l will fall into the same cell. For all variables falling into the same cell, it is up to the Gaussian term to determine the optimal value. For each grid cell, if the Gaussian center falls outside of it, the optimal point within the cell is the point along the cell boundary that is closest to the Gaussian center. If the Gaussian center falls inside of the grid cell, the optimal point will be the Gaussian center. Using the fact, we can effectively reduce the complexity of the optimization to a linear time algorithm in the number of squares in the location lattice, L when evaluating l or linear to the number of elements in the temporal lattice, T when evaluating t.

5 Experimental Results

In order to show the value of our approach in analyzing real-world data, we ran our model on a Twitter data set collected within the geographical boundary of Egypt from October 2009 to November 2013. We are particularly interested in this data set because social movements were frequent in Egypt at this time(Anderson 2011) and Twitter has been considered by many to have played at least some role in both planning and promoting of these demonstrations and gatherings (Lotan et al. 2011; Comunello and Anzera). We examine two aspects of the model in our experiment. First, we provide a qualitative interpretation of several events uncovered from a trained model to illustrate our ability to discover major events that can match reports from newspaper and online sources. Second, we provide a quantitative analysis of the prediction accuracies of location and time in a held out testing data set. In all cases, experiments are run with 400 Gibbs sampling steps, by fixing $L = 100$ and $T = 100$ and varying the number of events E unless otherwise noted. We set hyperparameters to be the following values: $\alpha = 0.05$, $\beta = 0.05$, $\gamma = 1.0$.

Data Set

We pre-processed the data so that only tweets written in Arabic remained, having observed that nearly all tweets utilizing the English character set were use a non-standard language that is phonetically similar to Arabic but was largely uninterpretable. For example, while with the help of a native speaker we were able to discern that ”tab3an” means ”of course”, large portions of these tweets were not interpretable. We filter out all tweets that are composed of less than 95% of Arabic characters[2]. After these preprocessing steps, we are left with roughly 1.4 million tweeets over with a vocabulary size of approximately 180K words. The geo-boundary we use is defined by the latitude/longitude point (21.89, 24.84) in the lower right corner and the point (32.16, 37.70) in the upper right corner. This covers the entirety of the area of Egypt. Table 2 is a summary of basic statistics in our data set.

Table 2: Basic Statistics of the Data Set

Geo Boundary	(21.89,24.84),(32.16,37.70)
Time Covered	from Oct,2009 to Nov,2013
Num.Tweets	1,436,186
Num.Words	183,478

Table 3: Spatial and temporal parameters of each event

E	Geo Center	G SD	Start Time	End Time
E1	30.86,29.87	0.43	2011-01-30	2011-03-21
E2	31.23,30.93	0.24	2013-09-10	2013-09-26
E3	31.77,30.84	0.32	2012-01-29	2012-03-22
E4	29.98,31.05	0.37	2012-10-15	2012-11-22
E5	31.20,29.57	0.37	2013-09-09	2013-10-13

Qualitative Analysis of Events

We believed that looking for real life interpretations of the events we have detected was an intuitive first step for model validation. To do so, we selected five events from the output of our trained model that spanned different geographical regions and time periods. The events discovered by the algorithm are summarized in Table 3. Please note that all event geo-centers are in the format of (Lat,lon) pair and the start date and end date are determined by $\theta_e^{(T)} - \sigma_e^{(T)}$ and $\theta_e^{(T)} + \sigma_e^{(T)}$. The spatial distribution of the five events is illustrated in Figure 2, where each point represents a tweet and a particular event being ascribed to by the color and shape. The figure displays up to 20,000 randomly sampled tweets that the model associated with these five events. Figure 2 also overlays a contour graph for all points in the graph. The contour plot is constructed using a mixture Gaussian distribution. To construct such a mixture Gaussian distribution, we use γ to serve as the mixture weight and use the event geographical centers and deviations for each Gaussian component. The result is a single distribution on a two-dimensional space that represents latitude and longitude. Curved circles in the contour plot represent the probability density of the distribution. Regions with multiple such curves are the ones that have steep change in their mixture Gaussian distributions. The contour plot shows three clear geographical clusters that correspond to three large cities in Egypt: Alexandria (left), Cairo (bottom right) and El-Mahalla El-Kubra (top right). As is also clear, certain events are located within the same cities. Without the temporal and lexical dimensions of the model, it would thus be difficult to discern differences between these events. However, exploring these distributions makes it relatively easy to observe the very different focus of each of these sets of tweets.

Figure 3 displays the temporal distributions of the five events of interest. Though we have analyzed each event independently in validating the model, we focus here on the most relevant event, labeled Event 1(E1). This event's tweets were heavily centered in Cairo and took place during the

[2] This percentage excludes English punctuations and Twitter mentions which usually fall into the English character sets. For more details on the data as part of a larger set, we refer the reader to (Carley, Wei, and Joseph)

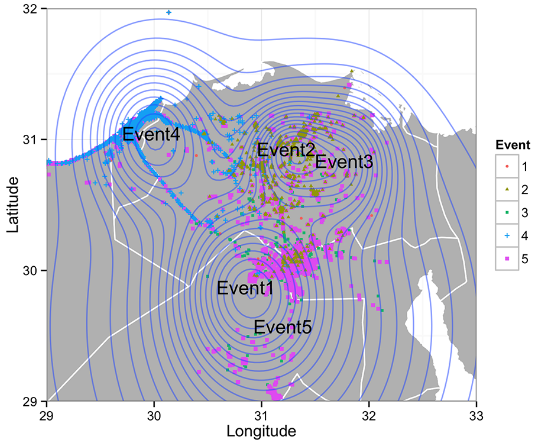

Figure 2: Geographical visualizations of the events and tweets belong to these events

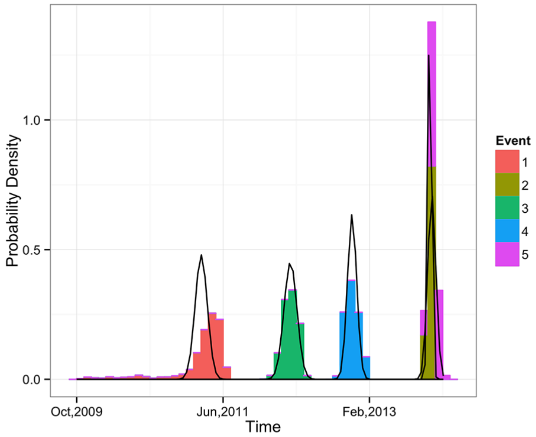

Figure 3: Temporal visualizations of the events

earlier portion of 2011. Without considering the topical focus of the event, these clues suggest that it corresponds to the initial protests that spurred the rapid spread of the social movement generally referred to as the Arab Spring (Anderson 2011). The protests were held largely in Tahrir Square, located within Cairo. Additionally, the central date associated with the protests was January 25 and start from January 28 the government started to force the protestors to leave. Nevertheless, the main protest lasted for approximately three weeks with continuous demonstrations continued after that. The model's inferred start date for Event 1 was January 30th, extending to an end date of March 21st.

The topic for Event 1 in the event category in Table 4 supports the idea that Event 1 uncovers the protests in Tahrir Square. Here we see words such "burn", "arrested", "honor", "injustice", "tortured", all of which match what we would expect to have seen and have expected to be protested during the demonstrations. Indeed, the focal date of the protests occurred on January 25 and we correspondingly observe that the popular term "jan25" appear frequently in our data set. The most representative words in Event 1's topic also include the name "ghonim", referring to the activist Wael Ghonim who played a central role in the protests.

Table 4: Top words for each event

E1	jan25	arrested	Egypt	Ghonim
	burn	injustice	Libya	tortured
E2	guilt	minimum	death	hurts
	Arif	home	pulse	lord of
E3	scar	pharmacist	disease	immediately
	eye	urticaria	evil	transplantation
E4	live	promise	tireless	condensed
	need	granulate	thanks	traipse
E5	end	voice	winter	lord, thou
	god	I want	lord	to god

While we focus here on Event 1, we note that the other events in our dataset do appear to have a qualitative realization in the real world. For example, Event 3 describes a (comparatively) minor event related to an outbreak of hand and foot disease in Egypt around February of 2012 [3].

Quantitative Analysis

While our qualitative analysis shows the real-world relevance of model output, it does not provide an illustration of how well the model fits the data, nor how it performs in a predictive setting. In this section, we compare three variants of the model and use each for three different prediction tasks given varying amounts of information about the test data. We train each model on a training data set composed of a randomly selected set of 90% of the data, leaving 10% of the data for testing. We explain the models used, the prediction tasks and the level of information we use from the test data in turn below.

Model variants The first model variant we consider is the full model proposed in Figure 1, marked as **M=L+T**. Second, we use a model with only the location component, ignoring information on time and thus ignoring $\bar{t}$, and $\Phi^{(T)}$. We denote this as **M=L**. Finally, we use a model that does not utilize location information, eliminating the location variables l, $\bar{l}$ and $Phi^{(L)}$. This is denoted as **M=T**.

Prediction tasks In the first task, we use each model and the information given to us in the test data to predict the words in each tweet. We evaluate this by using perplexity. Second, we use each model to predict the time of each tweet in the test data. Finally, we use each model to predict the location of each tweet in the test data.

Utilization of test data For all of the three prediction tasks, we vary the level of information we use from the test data in order to make the specified prediction. When analyzing perplexity, we vary whether or not we provide the model with time information, location information, neither or both. Giving the full model temporal or location information should naturally improve its ability to predict the words

[3] http://www.fao.org/news/story/en/item/129919/icode/

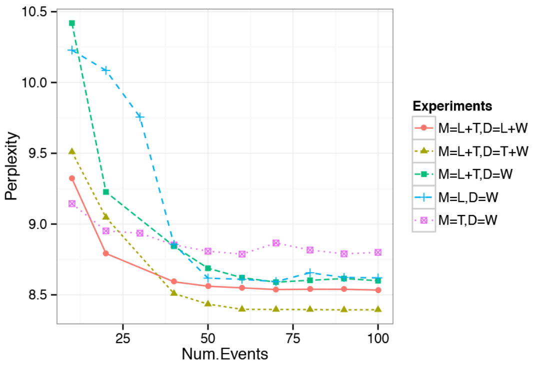

Figure 4: Perplexity over the number of events

used in the tweet. Note that when we give the model neither time nor location, the full model reduces to an LDA-like one. For predicting location, we vary whether or not the full model is given time, while for predicting time we vary whether or not the full model is given location. In both cases, all models are given the words in each document in the test data.

Perplexity analysis

We define the log perplexity of a document D_{test} in Equation 23. The value is equal to the negative sum of the log probability of all words appearing in our test data set. The higher the probability of each word in the model, the lower the perplexity.

$$log(PPX(D_{test})) = -\frac{1}{N_W}\sum_{d\in D}\sum_{w\in W_{d,*}} log(p(W_{d,w})) \tag{23}$$

Experimental results for perplexity are illustrated in Figure 4, where each colored line represents a different model/test data combination. For example, the line marked with ”M=L+T,D=L+W” represents the results with Model M=L+T trained on a data set where both location and text information are given for training while ”M=L+T,D=W” represents the same model where only text is given during training. On the x-axis we vary the number of events the model is trained with. Two important observations can be made about the plot. First, the figure shows that up to a point, model performance improves with an increasing number of events regardless of the model and test data used. When the number of events becomes large enough (e.g. 50) the decrease in perplexity is not as substantial as before, suggesting that the number of events is large enough to capture the major event information in our data set. Second, and more importantly, Figure 4 shows that the full model performs significantly better than all other models when given temporal and text information about the test data and when trained with a large enough number of events.

Prediction of location and time

The prediction of location and time shows similar pattern to perplexity, indicating that with certain number of events approaches, the full model performs better than the alternative models. And the more data we provide in training, the better prediction results we will achieve. This is illustrated in Figure 5 and Figure 6. Results thus indicate that the model is able to make good use of the provided information and improves on models that do not take into account location or time.

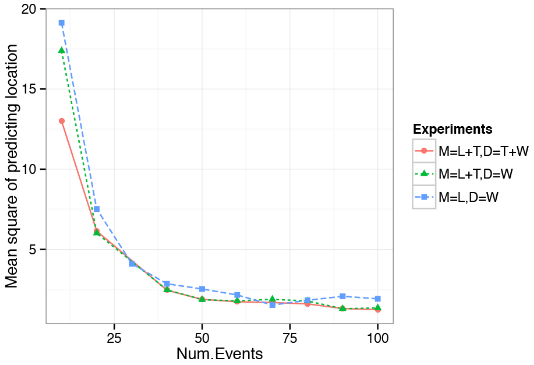

Figure 5: Mean square error (MSE) of predicting location over the number of events

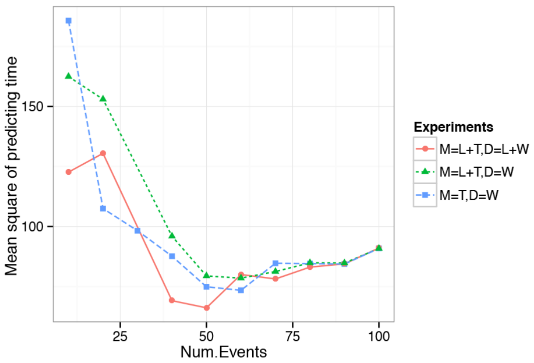

Figure 6: Mean square error (MSE) of predicting time over the number of events

6 Conclusions and Future Work

In this paper we proposed a probabilistic graphical model to discover latent events that are clustered in the spatial, temporal and lexical dimensions. Both the qualitative analysis and quantitative analysis we present justified our model on a large Twitter data set. Results show that our model improved over baseline approaches on a variety of prediction tasks. These qualitative efforts show that our work can be used in a variety of application areas where event extraction and location/time prediction of social media data is of interest, like in the detection of protests and demonstrations as shown here but also in detecting, for example, important local sporting events that may be relevant to different users.

One important component of the model is the Gaussian assumptions on the distributions of both the geo-spatial coordinates and the time stamps of the events. These assumptions ensure the existence of event location and time centers which are represented by the density mass in the Gaussian distribution. They also enable the model to discover events ranging from geo-spatially/temporally constrained to those that are more universal. The assumptions of using Gaussian to model location and time are also validated in prior work such as Hong et al. (2012) and Sakaki, Okazaki, and Matsuo (2010). Still, it may be interesting to explore other options for the structure of the geospatial and temporal distribution of events in the future.

There are several ways in which the present work can be further extended. First, both location and time are converted into an index through an evenly distributed selection function. There may be better approaches in cases where geo-temporal distributions are uneven, as is frequently the case in real-world data. Second, a control on granularity of the event should be added so that when tweaking the granularity of the variables, one can generate (or discover) events that are more localized or globalized. Finally, the assumption that a spatial and temporal related topic is allocated on an evenly spaced grid requires further investigation. One immediate solution is to use techniques such as k-d tree to generate topics on regions of different sizes.

7 Acknowledgments

This work was supported in part by the Office of Naval Research (ONR) through a MURI N000140811186 on adversarial reasoning, DTRA HDTRA11010102, MINERVA N000141310835 and by Center for Computational Analysis of Social and Organization Systems (CASOS). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Office of Naval Research, Defense Threat Reduction Agency (DTRA) or the U.S. government.

References

Anderson, L. 2011. Demystifying the arab spring: parsing the differences between tunisia, egypt, and libya. *Foreign Aff.* 90:2.

Andrieu, C.; De Freitas, N.; Doucet, A.; and Jordan, M. I. 2003. An introduction to mcmc for machine learning. *Machine learning* 50(1-2):5–43.

Benson, E.; Haghighi, A.; and Barzilay, R. 2011. Event discovery in social media feeds. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, 389–398. Association for Computational Linguistics.

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent dirichlet allocation. *the Journal of machine Learning research* 3:993–1022.

Brants, T.; Chen, F.; and Farahat, A. 2003. A system for new event detection. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, 330–337. ACM.

Carley, K. M.; Wei, W.; and Joseph, K. High dimensional network analysis. In *Big Data Over Networks, Robert Cui (Eds)*. Cambridge University Press.

Casella, G., and George, E. I. 1992. Explaining the gibbs sampler. *The American Statistician* 46(3):167–174.

Comunello, F., and Anzera, G. Will the revolution be tweeted? a conceptual framework for understanding the social media and the arab spring. 23(4):453–470.

Griffiths, T. L., and Steyvers, M. 2004. Finding scientific topics. *Proceedings of the National academy of Sciences of the United States of America* 101(Suppl 1):5228–5235.

Hong, L.; Ahmed, A.; Gurumurthy, S.; Smola, A. J.; and Tsioutsiouliklis, K. 2012. Discovering geographical topics in the twitter stream. In *Proceedings of the 21st international conference on World Wide Web*, 769–778. ACM.

Jordan, M. I. 1998. *Learning in Graphical Models:[proceedings of the NATO Advanced Study Institute...: Ettore Mairona Center, Erice, Italy, September 27-October 7, 1996]*, volume 89. Springer.

Kumaran, G., and Allan, J. 2004. Text classification and named entities for new event detection. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, 297–304. ACM.

Lotan, G.; Graeff, E.; Ananny, M.; Gaffney, D.; Pearce, I.; and Boyd, D. 2011. The revolutions were tweeted: Information flows during the 2011 tunisian and egyptian revolutions. *International Journal of Communication* 5:1375–1405.

Matuszka, T.; Vinceller, Z.; and Laki, S. 2013. On a keyword-lifecycle model for real-time event detection in social network data. In *Cognitive Infocommunications (CogInfoCom), 2013 IEEE 4th International Conference on*, 453–458. IEEE.

Pak, A., and Paroubek, P. 2010. Twitter as a corpus for sentiment analysis and opinion mining. In *LREC*.

Panisson, A.; Gauvin, L.; Quaggiotto, M.; and Cattuto, C. 2014. Mining concurrent topical activity in microblog streams. *arXiv preprint arXiv:1403.1403*.

Ritter, A.; Etzioni, O.; and Clark, S. 2012. Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, 1104–1112. ACM.

Sakaki, T.; Okazaki, M.; and Matsuo, Y. 2010. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*, 851–860. ACM.

Štajner, T., and Grobelnik, M. 2009. Story link detection with entity resolution. In *WWW 2009 Workshop on Semantic Search*.

Thom, D.; Bosch, H.; Koch, S.; Worner, M.; and Ertl, T. 2012. Spatiotemporal anomaly detection through visual analysis of geolocated twitter messages. In *Pacific Visualization Symposium (PacificVis), 2012 IEEE*, 41–48. IEEE.

Wallach, H. M. 2006. Topic modeling: beyond bag-of-words. In *Proceedings of the 23rd international conference on Machine learning*, 977–984. ACM.

Zubiaga, A.; Spina, D.; Amigó, E.; and Gonzalo, J. 2012. Towards real-time summarization of scheduled events from twitter streams. In *Proceedings of the 23rd ACM conference on Hypertext and social media*, 319–320. ACM.