Location-based Event Detection Using Geotagged Semantic Graphs

Yifang Wei
Computer Science Department
Georgetown University
Washington D.C., the United States
yw255@georgetown.edu

Lisa Singh
Computer Science Department
Georgetown University
Washington D.C., the United States
lisa.singh@georgetown.edu

ABSTRACT

Event detection using Twitter has attracted a significant amount of research attention. While the emphasis of the related literature has been on detecting events without considering geography information, this work regards an event as something occurring at a particular location and time. We take a system perspective, focusing on the process of event detection using a framework that highlights different steps needed for identifying events in this noisy domain. We also propose an algorithm which leverages geotagged bursty term graphs to detect events from a tweet stream. Evaluating our approach on three large tweet streams from three different domains shows our approach significantly improves the detection precision and recall when compared to the state of the art approaches. In general, we find that simple modifications to existing algorithms improves location-based event detection across methods.

KEYWORDS

Text Event Detection, Location-based Graph Mining

1 INTRODUCTION

Twitter, a social media platform, has been intensively leveraged by people to report real life events. Event detection using Twitter has also been an active area of research [2] [13] [6] [5] [19] [10] [18] [12] [16] [17] [15] [11]. While most of the previous research focuses on detecting events without considering geography information, we consider an event as *something occurring at a particular location and time*. In this work, we propose an approach for detecting location-specific events from a tweet stream.

Our approach leverages a location-specific semantic graph. Graphs are well-suited for representing complex connections between related entities, and graph algorithms are designed to reason about these connections. In this work, we construct a geotagged bursty term graph to represent words and relationships that rapidly increase in frequency at a particular location. Each node in this graph represent *bursty* terms, and an edge represents a co-occurrence of two bursty terms within a tweet. For each geotagged bursty term graph, our approach identifies the top connected nodes (bursty terms) to represent an event, and uses the tweet most representative of the extracted bursty terms as the event synopsis.

To summarize, our contributions to the literature are as follows: (1) we propose a new event detection algorithm that takes advantage of geotagged bursty term graphs to identify events; (2) for each detected event, our approach specifies a locality as its primary location and a representative tweet as its synopsis, making the detected events more informative than that of the state of art; (3) we present a system framework that walks through the process of location-based event detection on Twitter; and (4) an empirical evaluation on three large real-world data sets shows that combining geography information with changes in bursting words significantly improves the event detection precision and recall when compared to the state of the art.

2 RELATED LITERATURE

The majority of literature related to unsupervised event detection on Twitter does not consider geography information when detecting events. Instead, it considers bursty textual segments (e.g., terms, phrases) to represent an event [19] [18] [12] [8] [7] [3]. One of particular relevance, proposed by Weng and colleagues, evaluates a term's burstiness using a wavelet transform [18]. The proposed method then calculates the pairwise correlation between pairs of busty terms in the wavelet domain, and uses a coherently connected group of bursty terms to signal an event's occurrence.

Another set of proposed algorithms leverages different graph structures when detecting events on Twitter [2] [13] [6] [5] [4] [14] [5]. Focusing on the ones that are most relevant to this research, Cataldi et al. [4] employ a Twitter following/follower network to quantify a user's significance in propagating information. Further, these values are used to modulate the *energy* of emergent keywords, and each strongly connected component of emergent keywords represents an event. Meladianos et al. [14] propose a K-degenerate graph, in which nodes represent terms in tweets, and edges represent co-occurrences of pairs of terms in a tweet. This approach uses terms pertaining to the highest *KCore* to represent the trending event in a time window of the tweet stream. Our approach differs from all the above mentioned works since our graph is location specific, i.e. we have different graphs for different locations, and we use bustiness to determine whether or not a word should appear in the graph, only words with large changes in frequency are added to the graphs.

Some previous work does consider location-based event detection [20] [9] [1] [15]. However, their models are not directly comparable to ours because of the assumptions made about location.

MLG'17, August 2017, Halifax, Nova Scotia, Canada

ACM ISBN 123-4567-24-567/08/06…$15.00
https://doi.org/10.475/123_4

Zhou and Chen [20] extend the classical LDA topic model to incorporate the location and time information, such that each topic is drawn from a distribution of words, locations, and time. In order to accomplish this, they extract user location information from user profiles and map tweets based on those locations. Our model does not assume the availability of user location information. Abdelhaq et al. [1] suggest that events might occur in a geographical space of substantial extent or within a small region. They construct a geographical grid, and represent the usage of a term as a distribution in the grid. Having a small usage entropy suggests the main usage of the term occurs in a geographical space of limited extent. They assume that the tweets are geo-tagged. Our approach does not make that assumption. Ramakrishnan et al. [15] build a system to predict civil unrests in Latin America. They use a list of names from different locations in Latin America to filter the tweet stream fed into their event detection system. Our approach uses a location ontology to allow for any major location to be investigated for events.

In general, what distinguishes our work from previous literature is the following: (1) using separate graphs for each location identified in the tweet stream, (2) using a location ontology to label the location associated with a tweet, (3) generating graphs that only contain bursty terms. We will show that these simple differences in data representation have a significant impact on the precision and recall of the detected events.

3 DEFINITIONS AND ASSUMPTIONS

Let $\mathbb{D} = \{d_i, d_2, ..., d_3\}$ denote a tweet stream, where d_i is the i^{th} tweet in the stream. Let t_i denote the time when a tweet is published. We are interested in detecting events E that occur in a specified time window. Let τ represent the set of time windows for tweets in $\mathbb{D}$. Each window contains a set of times, e.g. $\tau_1 = [t_0, t_j)$, where j is the j^{th} tweet in the first window, and $D(\tau_j)$ represents the tweets that occur in time window τ_j. A tweet is composed of a set of words, $W = w_i, w_2, ..., w_m$ and possibly a location l.

We make the following assumptions about tweets in the tweet stream $\mathbb{D}$:

(1) Given its 140 character length limit, a single tweet maps to at most one event [1].
(2) Not all the tweets in $\mathbb{D}$ contain a location. In these cases, we say that the tweet is not discussing a particular event.
(3) When a tweet has a specified location l, this location tends to be relevant to the event reported in the tweet, most likely mapping to the location where the event occurs.
(4) Two different events do not share the same set of keywords and the same location within the same time window τ_i.

Problem Statement: Given a tweet stream $\mathbb{D}$, the task of geotagged event detection is to identify events E from $\mathbb{D}$. The representation of an event e_m is a three-element tuple $\{\Delta, \Phi, \Sigma\}_m$, in which Δ represents the location where the event occurs, Φ represents the time when the event occurs, and Σ represents the synopsis of the event.

[1] While this is not always the case, we empirically find that it is a reasonable assumption that does not reduce the quality of the detected events

4 EVENT DETECTION

Algorithm 1 presents a high level view of our proposed approach. The input to our approach is a tweet stream $\mathbb{D}$, the number of time windows in the training phase p, and a locality ontology $\mathbb{O}$. The output is a set of detected events for each time window $D(\tau_i)$ in the detecting phase, and each detected event e is in the form of a three-element tuple. Here tuple $\{\Delta, \Phi, \Sigma\}$ is represented by tuple $(l, \tau_i, S(l, \tau_i))$, in which l represents the location where the event occurs, τ_i represents the time window when the event occurs, and $S(l, \tau_i)$ represents the synopsis of the event. Our approach begins by calculating the term frequency $w(\tau_i)$ of each term in each training window τ_i, where $w(\tau_i)$ denotes the term frequency of term w in $D(\tau_i)$ (Line 3). In the same way, our approach calculates the term frequency $w(\tau_i)$ of each term in each detecting window τ_i (Line 6). A term is considered *bursty* in time window τ_i if its term frequency compared to the previous p windows is significantly different. We denote a bursty term in time window τ_i as $b(\tau_i)$, and the set of bursty terms as $B(\tau_i) = \{b(\tau_i)\}$ (Line 7). Then, the predominant location l for each tweet is identified and tweets in $D(\tau_i)$ are divided into groups $\{D(l, \tau_i)\}$ according to their predominant locations (Line 8). A geotagged bursty term graph $G(l, \tau_i)$ is constructed using tweets having the same location (Line 10), in which a node is a bursty term $b(\tau_i)$ and an edge represents the co-occurrence of two bursty terms within a tweet. For each geotagged graph $G(l, \tau_i)$, an event e is extracted by identifying a semantically cohesive set of nodes (Line 11). Using the tweets in $D(l, \tau_i)$, the tweet most representative of the detected event e is selected as the event synopsis $S(l, \tau_i)$ (Line 12).

Algorithm 1: High level algorithm for online geotagged event detection

Input: *A tweet stream*: $\mathbb{D}$
The number of training windows: p
A location ontology: $\mathbb{O}$

Output: *Detected events*: E

1 /*******Training Phase*******/
2 **for** $i \leftarrow 1$ *to* p **do**
3     $w(\tau_i) = calculate\_term\_frequency(D(\tau_i))$

4 /*******Detecting Phase*******/
5 **for** $i \leftarrow p + 1$ *to* n **do**
6     $w(\tau_i) = calculate\_term\_frequency(D(\tau_i))$
7     $B(\tau_i) = bursty\_term\_extraction(w(\tau_i), \{w(\tau_i - p), \ldots, w(\tau_i - 1)\})$
8     $\{D(l, \tau_i)\} = location\_identification(D(\tau_i), \mathbb{O})$
9   **for** $D(l, \tau_i) \in \{D(l, \tau_i)\}$ **do**
10         $G(l, \tau_i) = create\_geotagged\_bursty\_term\_graph(D(l, \tau_i), B(\tau_i))$
11         $e = extract\_event(G(l, \tau_i))$
12         $S(l, \tau_i) = generate\_synopsis(e, D(l, \tau_i))$

The framework of our proposed approach has five components: location identification, bursty term identification, geotagged bursty term graph generation, geotagged event detection, and event synopsis generation (shown in Figure 1). We take a system perspective by describing a framework for generating location-based events and discussing our proposed algorithm in the context of this framework.

4.1 Bursty Term Extraction

This component extracts terms that are bursty in the context of their term frequencies in the current time window τ_i compared to their values in the previous p time windows. Given all the terms in $D(\tau_i)$, the terms with a term frequency that is at least one standard deviation above their mean values are considered *bursty* in time window τ_i:

$$B(\tau_i) = \{w|w(\tau_i) > MEAN(\mathbf{w}(\tau_{i-p}, \tau_{i-1})) + STD(\mathbf{w}(\tau_{i-p}, \tau_{i-1}))\} \tag{1}$$

where $\mathbf{w}(\tau_{i-p}, \tau_{i-1})$ denotes $\{w(\tau_{i-p}), ..., w(\tau_{i-1})\}$. Term frequencies' mean values and standard deviations are calculated based on their values in the previous p windows, where p is a user-specified parameter. Terms go viral on Twitter because people are using these terms to discuss occurring event(s). When discussing the same event, people tend to use the same set of keywords. When a new event occurs, people will tend to use a new set of words that differs from the set of words they used to discuss previous events. Therefore, we should be able to represent events reported in tweets with bursty terms if these bursty terms are appropriately grouped.

4.2 Location Identification

The location identification component identifies the predominant location for each tweet. Using open-source data (described in Section 5), our approach constructs an ontology, in which each node represents a location. Such a location could be a country, a governorate, or a city. Using this ontology, we determine the predominant location for each tweet by counting the number of occurrences of each location in the tweet, and aggregating the number of occurrences of children locations to their parent locations iteratively. The location with the highest frequency count is considered the predominant location of the tweet. A tie indicates that a tweet does not have a predominant location and it is therefore removed from further analysis. Based on their determined predominant locations, our approach divides $D(\tau_i)$ into groups $\{D(l, \tau_i)\}$, where $D(l, \tau_i)$ denotes the set of tweets having l as their predominant location.

4.3 Geotagged Bursty Term Graph Generation

While there are many different representations of graphs, we choose to leverage a co-occurrence graph. For a set of tweets $D(l, \tau_i)$ having the same predominant location l, we construct a graph $G(l, \tau_i)$ in which a node represents a bursty term and an edge represents the co-occurrence of two terms within a tweet. Since all the nodes in the graph are tagged with the same location, we refer to the graph as a *geotagged graph*.

Our empirical analysis shows that in most cases two different events do not share the same set of keywords and the same location within the same time window (refer to our Assumption 3); in rare cases where two different events share the same set of keywords and the same location within the same time window, these two events tend to be closely related. Thus, we consider them as belonging to the same event. As an example, on November 13, 2015, three suicide bombers struck outside the national sports stadium of France during a football match. It was followed by several mass shootings and a suicide bombing at cafes and restaurants. Each of these attacks is considered a part of a large event *November 2015 Paris attacks*, as opposed to separate events [2].

While this distinction seems detailed, an important component of our approach is to generate separate graphs for bursty terms associated with different locations in each time window. So understanding location "boundaries" is necessary to accurately identify events. As we will show, doing so leads to improved accuracies.

4.4 Geotagged Event Extraction

The geotagged event extraction component aims to extract the terms most representative of all the nodes in the geotagged bursty term graph $G(l, \tau_i)$. This means that we want the selected terms to form a semantically cohesive group. In other words, we do not want the terms that have the highest frequency alone. Instead, we want the terms that are most frequent, but part of a connected component. Therefore, we begin by identifying the node/term with the highest frequency in the graph. We then select its neighbors in frequency order and iteratively continue the process until we have a set of k nodes. Once we have k nodes, we stop. The subgraph containing the k identified nodes is considered a semantically cohesive group since it is frequently occurring and has relationships among the nodes.

4.5 Synopsis Generation

Along with detecting events, it is important to determine a meaningful description of the event. While using keywords is one option, it is less informative than the tweets themselves. Therefore, this final component of our framework generates a synopsis by identifying the tweet that best describes the detected event, i.e., the tweet containing the most information about the event. To accomplish this, we identify the tweet that contains the highest overall term frequency of bursty terms associated with the detected event.

5 EVALUATION OF GEOTAGGED EVENT DETECTION

In this section we empirically evaluate our approach. We begin this section by describing the data sets, and then evaluate the different components of our framework.

5.1 Datasets

For our empirical analysis, we consider three tweet streams in three distinct domains: terrorism, migration, and politics.

Terrorism: The Islamic State of Iraq and Syria (ISIS) is a Sunni jihadist group with a violent ideology. The group is responsible for terrorist attacks worldwide in recent years, emerging as a top security concern for the United States and many other countries [3]. We work with an interdisciplinary team of researchers, students, and policymakers, some of whom have years of in-field research experience in the Middle East. With their help, we identified a set of hashtags that are related to ISIS. When collecting tweets using these hashtags with Twitter API, we find that #isis and #isil are the only two hashtags that each returns over 10 thousand tweets

[2]https://en.wikipedia.org/wiki/November_2015_Paris_attacks

[3]http://law.emory.edu/eilr/content/volume-30/issue-2/comment/isis-largest-threat-world-peace.html

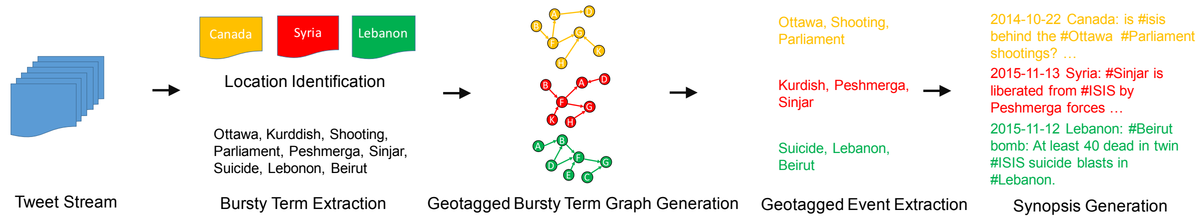


Figure 1: The framework of our proposed approach

Table 1: Datasets

	Hashtags	Start Date	End Data	#Tweets
Terrorism	#isis, #isil	2014-09-10	2016-03-10	15,835,184
Migration	#flee, #refugees	2014-09-10	2016-03-10	5,511,455
Politics	#hillary, #trump	2016-07-03	2016-09-13	18,368,438
	#votehillary, #votetrump	2016-09-27	2016-11-13	2,210,202

per day on average. This evaluation consists of tweets containing one of these two hashtags published between September 2014 and March 2016.

Migration: More than one million migrants crossed into Europe in 2015 [4], triggering a crisis which many European countries have been struggling to deal with. We worked with the Institute for the Study of International Migration (ISIM) at Georgetown University to identify a set of hashtags related to migration. Two popular ones are #flee and #refugees. In this evaluation, we use tweets containing one of these hashtags between September 2014 and March 2016.

Politics: Our research team collected a set of hashtags related to the 2016 US presidential election: #hillary, #trump, #votehillary, and #votetrump. This evaluation uses tweets containing any of these four hashtags from July 2016 to November 2016, which covers the most crucial part of the campaign season. Table 1 shows some statistics about these three datasets.

Noise is pervasive in tweets. An analysis of 2,000 English tweets originating from the United States conducted by San Antonio-based market-research firm Pear Analytics shows only 4% of the 2,000 tweets are reporting real world events [5]. Therefore, a well designed preprocessing step is imperative for a data mining task on a tweet stream. Our preprocessing includes (1) removing all the retweets, since many retweets are automatically generated by retweet bots and are thus irrelevant to real world events [6], (2) removing any tweet containing over two urls, or over two Twitter handles, (3) filtering the tweet stream using the cybozu library [7] to remove any tweet considered non-English, (4) tokenizing and stemming tweets, and (5) removing stopwords (the hashtags employed to collect tweets are also considered stopwords) and non-alphanumerical characters.

[4]https://en.wikipedia.org/wiki/European_migrant_crisis
[5]https://en.wikipedia.org/wiki/Twitter
[6]https://en.wikipedia.org/wiki/Twitterbot
[7]http://developer.cybozu.co.jp/archives/oss/2010/10/language-detect.html

Table 2: Three exemplar tweets whose determined predominant locations are relevant to the events reported in the tweets, but not the locations where the events occur

	Tweet	Determined Locality	Actual Locality
Terrorism	raqqa the airstrikes today is by russian warplanes and they don t hit isis hq most of the places are civilian they destroy bridge	Russia	Syria
Migration	the three year old was killed as his family made a desperate bid to flee syria and mirror columnist carole malone says we must learn	Syria	Turkey
Politics	breaking donald trump jr just pledged 89 delegates from new york officially placing donaldtrump over 1237 as the gop nominee	New York	Ohio

5.2 Location Identification

We build our location ontology using Wikipedia and Statoids [8]. Wikipedia has a set of pages listing all the major cities around the world by country, and Statiods lists governorates and the major cities in governorates and their populations for each country. Leveraging these two sources, we construct a three-level ontology, including countries, governorates, and cities. The raw ontology has 37,379 nodes with 55,816 locality names. After applying the population-based duplicate name removal, the ontology is left with 46,560 distinct locality names.

To evaluate our location identification approach, we obtain a random sample of 300 tweets from the tweets having predominant locations for each of the three datasets, and manually check whether the determined location of a tweet maps to the location where the event reported in the tweet actually occurs. We find that for some tweets the determined predominant locations are the locations where the events reported in the tweets occur (Type A), whereas for some other tweets the determined predominant locations are relevant to the events reported in the tweets, but not the locations where the events occur (Type B). As an example, the first row in Table 2 gives a tweet reporting *Russian airstrikes in Syria on Nov 3, 2015.* The determined location of the tweet is Russia, but the airstrikes actually occurred in Syria. We consider this a relevant location, but it is not the predominant one. This type of location labeling is a mistake in certain contexts, but reasonable in other contexts. Therefore, we keep track of how often this occurs.

[8]http://www.statoids.com

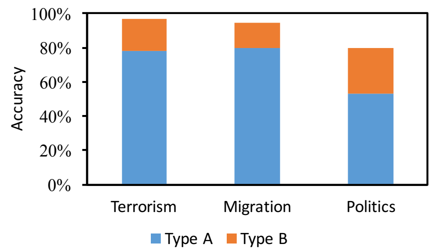


Figure 2: Location identification accuracy. Type A: the determined locations are the locations where the events reported in the tweets occur; Type B: the determined locations are relevant to the events reported in the tweets, but not the locations where the events occur

If we consider both types of answers as correct answers, our location identification approach achieves accuracies of 97.0% and 94.7% for the terrorism dataset and migration dataset, respectively (shown in Figure 2). The accuracy is 79.7% for the politics dataset. This lower accuracy occurs because some names of politicians also map to location names, e.g. Lincoln.

5.3 Geotagged Graph Generation

Since an exhaustive and authoritative list of ground truth events is not available, our team manually created a list of ground truth events. We focused in on a 30 day consecutive stream for each of the three datasets. More specifically, we choose Nov 1, 2015 to Nov 30, 2015 for the terrorism dataset, September 1, 2015 to September 30, 2015 for the migration dataset, and July 13, 2016 to August 12, 2016 for the politics dataset. These time periods were chosen because of their comparatively higher volume of data during these time periods (refer to Figure 1).

Applying location identification and geotagged graph generation to a time window of a tweet stream yields a list of geotagged graphs. In the rest of this evaluation, we only consider the top 5 geotagged graphs generated for each time window, which correspond to 150 geotagged graphs throughout the 30 day tweet stream. Note that for the terrorism dataset and the migration dataset, we generate geotagged graphs on the country level. For the politics dataset, we generate geotagged graphs on the state level. Figure 4 shows the geotagged graph distribution over countries or states. For the terrorism dataset, the top 2 countries are Syria and Russia. It confirms our intuition: Syria is the country where most of ISIS's territory is, and Russia is deeply involved in the conflict. For the migration dataset, the top 2 countries are Syria and Germany: Syria is the country where most of the refugees left from to go to Europe, and Germany has accepted a large number of refugees. With respect to the politics dataset, the top states include DC, New York, Ohio, Texas, Florida, Colorado, etc. DC is the US capital; New York is the home state of both candidates; Texas has the second largest number of electors in the electoral college; Ohio, Florida, and Colorado are three key swing states, where both candidates had a number of events.

5.4 Geotagged Event Extraction

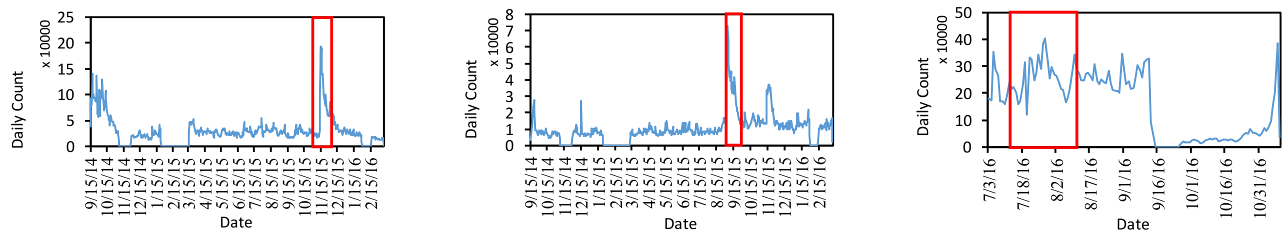
In detecting events using our geotagged event extraction component, we set the length of the time window to a day. We set the number of time windows in the training phase p to 10, and the number of top terms representing a detected event k to 10. These numbers were chosen based on an empirical sensitivity analysis. Note that the 10 training windows are not included in the selected 30 day tweet stream. They correspond to 10 days prior to the 30 day detecting phase. We evaluate the accuracy of the events detected by our approach, and compare our approach (Geo) to three state-of-art event approaches described in Section 2: [14] (KCore), [4] (Emerge), and [18] (Wavelet). For each of the four approaches, we consider the top 5 events (if available) detected for each day, and manually check whether each of them maps to any real world event by exploring Wikipedia event pages and using their key terms as search queries for Google search. In this way, we build a list of ground truth events reported in the tweet stream. Then we manually check whether a detected event maps to any event on the ground truth event list. Table 3 shows the number of events labeled as ground truth events for the three datasets.

Table 3: The number of ground truth events for the three datasets

Dataset	Number of Events
Terrorism	100
Migration	98
Politics	107

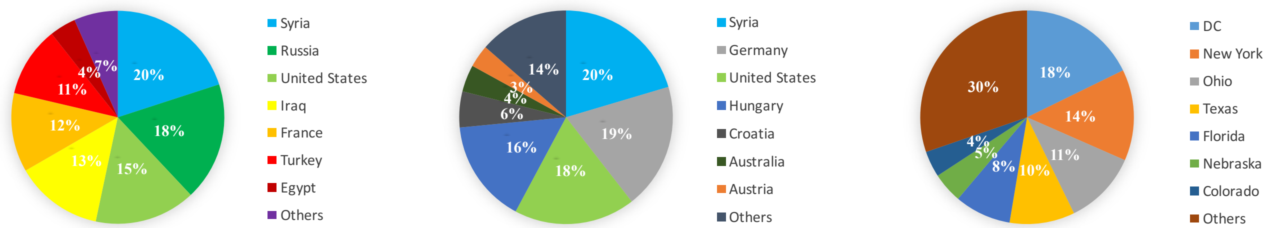
We use two metrics in evaluating event detection accuracy: precision and recall. Precision is the percentage of detected events mapping to ground truth events, and recall is the percentage of ground truth events mapped by detected events. Figure 5 shows the precision and recall of our proposed approach and the three baseline approaches. We can see that our approach achieves over 55% precision and over 65% recall on all the three datasets, significantly outperforming the baselines in terms of both precision and recall. The success of our approach can be attributed to two factors: (1) it provides a tweet as the synopsis for a detected event, which greatly facilitates mapping it to a ground truth event. In contrast, both the Emerge approach and the Wavelet approach provide a set of terms as the synopsis of a detected event, which is much less interpretable; (2) our approach detects events only based on the tweets with predominant locations. This helps filter Twitter memes since memes usually do not contain location information. When looking at the results produced by the state of the art methods, we observe that most of the events detected by the KCore approach are Twitter memes. Finally, we pause to mention that there could be events that none of the methods detected that we are missing. This is a limitation of our evaluation approach resulting from evaluation data sets not being available.

Since our event detection approach also specifies a location for each detected event, we are interested in understanding the how the location maps to the ground truth locations of the events: what is the percentage of the determined locations that map to where the events actually occur (Type A), and what is the the percentage of the determined locations that are relevant to the detected events, but not the locations where the events occur (Type B). The results are shown in Figure 6. We can see that by combining both types of answers together, our approach achieves over 90% accuracy for all the three datasets. Again, the terrorism and migration location accuracy are higher than the location accuracy for the politics dataset.



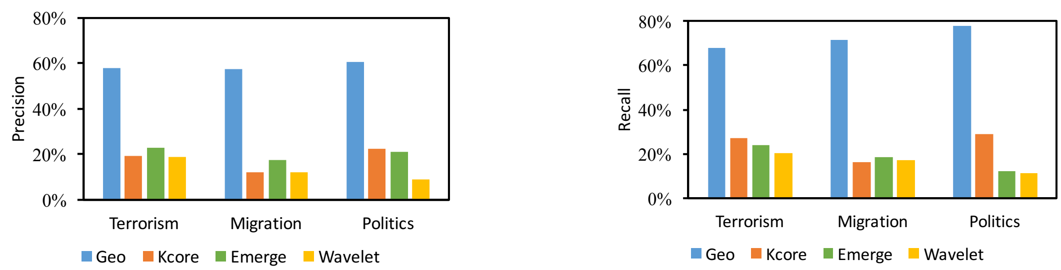
(a) Terrorism (b) Migration (c) Politics

Figure 3: Daily tweet count of the three datasets (Red boxes confine the time periods chosen for detecting events; There are three gaps for the terrorism dataset and the migration dataset, and one gap for the politics dataset in our data collection due to infrastructure unavailability)



(a) Terrorism (b) Migration (c) Politics

Figure 4: Geotagged graphs distribution over countries and states



(a) Precision (b) Recall

Figure 5: Event detection accuracy of different approaches

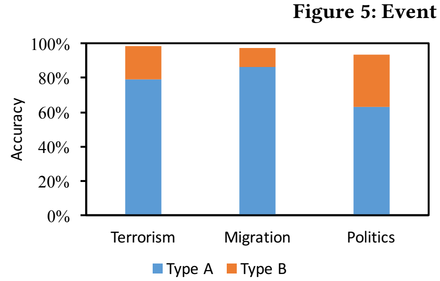


Figure 6: Event location accuracy. Type A: the determined locations are the locations where the events occur; Type B: the determined locations are relevant to the events, but not the locations where the events occur

5.5 Event Synopsis

We now show the event synopses generated by different approaches. Table 4 provides the synopses of the top 3 events (if available) detected by different approaches on a single day for the terrorism dataset. Table 5 and Table 6 provide the synopses of the top 3 events detected by our approach on a single day for the migration dataset and the politics dataset (events detected by baseline approaches are not shown given the space limitation). In these tables, the *GT* rows list the corresponding ground truth events, and the *Map GT* column shows which ground truth event a detected event maps to. We can see that the event synopses generated by our approach and the KCore are much more informative than the Emerge approach and the Wavelet approach, since both our approach and the KCore approach give a tweet as event synopsis, whereas the Emerge approach and the Wavelet approach give a set of terms as event synopsis. On the other hand, our approach achieves a much higher event detection accuracy than the KCore approach, which can be seen in Table 4, Table 5, and Table 6. Combining the event accuracy and event synopsis informativeness, our approach clearly outperforms all the three baseline approaches.

5.6 Case Study: Geotagged vs. Non-Geotagged

As stated previously, one of the reasons our approach performs better than the baseline approaches is that it leverages geography information during graph construction. Although the baseline approaches do not consider geography information in detecting events, we hypothesize that leveraging geography information could also improve their performance. For this analysis, we investigate the impact of incorporating location information into the

Table 4: Synopses of events detected by different approaches on Nov 13, 2015 for the terrorism dataset

	No.	Map GT	Event Synopsis
GT	1	GT 1	ISIS carries out a series of coordinated attacks in Paris, killing 130 people.
	2	GT 2	Pentagon confirms Jihadi John was killed in an airstrike carried out by US.
	3	GT 3	Suicide bomb and road blast kill 26 in Baghdad.
Geo	1	GT 1	FRANCE: good job boys thanks for paris attack france paris attack parisattack jihadist mujahedeen islam terror alqaeda isis is isil
	2	GT 2	SYRIA: us military says reasonably certain that raqqa airstrike killed daesh terrorist jihadijohn reuters is isis syria iraq
	3	GT 3	IRAQ: 150 killed wounded in a triple isis suicide attacks in east baghdad in sadr in one day iraq
KCore	1	GT 1	good job boys thanks for paris attack france paris attack parisattack jihadist mujahedeen islam terror alqaeda isis is isil
	2	GT 1	fuck terrorism fuck isis fuck al queda fuck u and fuck everything you fucking stand for you fucking fuckwit fuckers isis parisattacks
	3	GT 1	parisattacks is a wake up call for every single country in the world except the countries that support isis parisattacks parisshooting
Emerge	1	GT 1	isis, the, to, of, in, ccot, rt, is, on, paris
Wavelet	1	None	cold, usual, carlyfiorina, nuclear, journalneo, experi, thr, constitut, weather, ecentauri
	2	None	ilnewsflash, billion, tag, price, oct, record, victim, star, hunt, shock
	3	None	destruct, true, communiti, thereaperteam, hellfir, oust, journal, boom, worth, european

Table 5: Synopses of events detected by our approach on Sept 5, 2015 for the migration dataset

	No.	Map GT	Event Synopsis
GT	1	GT 1	Alan Kurdi, a three-years-old Syrian child, drown in the Mediterranean Sea.
	2	GT 2	Refugees are stranded at Budapest train station.
	3	GT 3	Refugees applauded in Germany by locals.
Geo	1	GT 1	SYRIA: the three year old was killed as his family made a desperate bid to flee syria mirror columnist carole malone says we must learn
	2	GT 2	HUNGRAY: hungarian families arrive at the railway station with aid for refugees a new wave waiting to leave for their destination budapest
	3	GT 3	GERMANY: check out germany being super cool as ever applauding refugees arriving in munich way to go deutschland

baseline approaches. More specifically, we divide a tweet stream $\mathbb{D}$ into multiple tweet streams $\{\mathbb{D}(l)\}$ by using the predominant location of each tweet in $\mathbb{D}$. These geotagged tweet streams are then fed into a baseline approach.

Table 6: Synopses of events detected by our approach on August 11, 2016 for the politics dataset

	No.	Map GT	Event Synopsis
GT	1	GT 1	Hillary Clinton makes a speech in Detroit.
	2	GT 2	Donald Trump holds rally in Kissimmee, Florida.
	3	GT 3	Utah Governor Gary Herbert says he'll be voting for Donald Trump.
Geo	1	GT 1	MICHIGAN: hillary you bill and obama destroyed michigan economy now you have the nerve to speak about creat jobs in detroit
	2	GT 2	FLORIDA: and whichever florida pastors applauded to trump claim about obama founding isis should resign immediately
	3	GT 3	UTAH: utah govgaryherbert is voting for trump so i must not vote for herbert in the fall anyonebuttrump anyonebutherbert

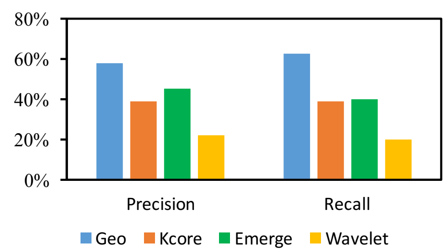


Figure 7: Event detection accuracy of the geotagged baseline approaches and our approach on the terrorism dataset

Given a geotagged tweet stream $\mathbb{D}(l)$, each geotagged baseline approach returns one event at location l during each time window τ_i. The events detected from multiple geotagged tweet streams $\{\mathbb{D}(l)\}$ within the same time window τ_i are sorted according to the number of tweets in the tweet window $|D(l, \tau_i)|$ in a descending order. Similar to Section 5.4, we select the top 5 events per time window for each of the three geotagged baseline approaches and our proposed approach, and manually check whether a detected event maps to any real world event.

We apply the three geotagged baseline approaches and our proposed approach to the terrorism dataset. Figure 7 shows the precision and recall of the four approaches. Combining Figure 5 (event detection accuracy of non-geotagged baselines) and Figure 7 (event detection accuracy of geotagged baselines), we can see that leveraging the location information improves both the detection precision and recall by over 20% for the KCore approach and the Emerge approach; with respect to the Wavelet approach, the improvement is not significant, but still observable. We conclude that including geography information is beneficial to event detection in general. On the other hand, we can see that our proposed approach still outperforms the geotagged baseline approaches. It suggests that other components of our approach, including bursty term extraction, geotagged bursty term graph generation, and event extraction, are important for detecting events accurately.

6 CONCLUSIONS

In this paper, we proposed an approach for detecting events from a tweet stream that leverages geotagged bursty term graphs. An empirical evaluation on three large real-world data sets shows that our approach significantly improves the event detection precision and recall when compared to the state of the art, and that adding geography information to the other approaches does improve their precision and recall. We also find that events are easier to detect because of our simple approach to labeling them using tweets containing bursty words. Future directions include understanding the relationship between event detection and levels of noise on Twitter, and considering varying time windows and periods for event detection.

Acknowledgments We thank subject matter experts for labeling data and their general subject matter expertise provided throughout the process. This work was supported in part by the National Science Foundation (NSF) Grant SMA-1338507 and the Georgetown University Mass Data Institute (MDI).

REFERENCES

[1] Hamed Abdelhaq, Christian Sengstock, and Michael Gertz. 2013. Eventweet: Online localized event detection from twitter. *VLDB Endowment* 6, 12 (2013), 1326–1329.
[2] Charu C Aggarwal and Karthik Subbian. 2012. Event Detection in Social Streams.. In *SDM*, Vol. 12. SIAM, 624–635.
[3] Hila Becker, Mor Naaman, and Luis Gravano. 2011. Beyond Trending Topics: Real-World Event Identification on Twitter. *ICWSM* 11, 2011 (2011), 438–441.
[4] Mario Cataldi, Luigi Di Caro, and Claudio Schifanella. 2010. Emerging topic detection on twitter based on temporal and social terms evaluation. In *Proceedings of the Tenth International Workshop on Multimedia Data Mining*. ACM, 4.
[5] Feng Chen and Daniel B Neill. 2014. Non-parametric scan statistics for event detection and forecasting in heterogeneous social media graphs. In *KDD*. ACM, 1166–1175.
[6] Jure Ferlez, Christos Faloutsos, Jure Leskovec, Dunja Mladenic, and Marko Grobelnik. 2008. Monitoring network evolution using MDL. In *ICDE*. IEEE, 1328–1330.
[7] Georgiana Ifrim, Bichen Shi, and Igor Brigadir. 2014. Event detection in twitter using aggressive filtering and hierarchical tweet clustering. In *Second Workshop on Social News on the Web (SNOW), Seoul, Korea, 8 April 2014*. ACM.
[8] Theodoros Lappas, Benjamin Arai, Manolis Platakis, Dimitrios Kotsakos, and Dimitrios Gunopulos. 2009. On burstiness-aware search for document sequences. In *KDD*. ACM, 477–486.
[9] Theodoros Lappas, Marcos R Vieira, Dimitrios Gunopulos, and Vassilis J Tsotras. 2012. On the spatiotemporal burstiness of terms. In *VLDB*. 836–847.
[10] KH Lei, R Khadiwala, and KC-C TEDAS Chang. 2012. A Twitter-based Event Detection and Analysis System. In *ICDE*.
[11] Jure Leskovec, Lars Backstrom, and Jon Kleinberg. 2009. Meme-tracking and the dynamics of the news cycle. In *KDD*. ACM, 497–506.
[12] Chenliang Li, Aixin Sun, and Anwitaman Datta. 2012. Twevent: segment-based event detection from tweets. In *CIKM*. ACM, 155–164.
[13] Cindy Xide Lin, Bo Zhao, Qiaozhu Mei, and Jiawei Han. 2010. PET: a statistical model for popular events tracking in social communities. In *KDD*. ACM, 929–938.
[14] Polykarpos Meladianos, Giannis Nikolentzos, François Rousseau, Yannis Stavrakas, and Michalis Vazirgiannis. 2015. Degeneracy-Based Real-Time Sub-Event Detection in Twitter Stream.. In *ICWSM*. 248–257.
[15] Naren Ramakrishnan, Patrick Butler, Sathappan Muthiah, Nathan Self, Rupinder Khandpur, Parang Saraf, Wei Wang, Jose Cadena, Anil Vullikanti, Gizem Korkmaz, et al. 2014. 'Beating the news' with EMBERS: Forecasting Civil Unrest using Open Source Indicators. In *KDD*. ACM, 1799–1808.
[16] Alan Ritter, Oren Etzioni, Sam Clark, et al. 2012. Open domain event extraction from twitter. In *KDD*. ACM, 1104–1112.
[17] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. 2010. Earthquake shakes Twitter users: real-time event detection by social sensors. In *WWW*. ACM, 851–860.
[18] Jianshu Weng and Bu-Sung Lee. 2011. Event Detection in Twitter. *ICWSM* 11 (2011), 401–408.
[19] Wei Xie, Feida Zhu, Jing Jiang, Ee-Peng Lim, and Ke Wang. 2013. Topicsketch: Real-time bursty topic detection from twitter. In *ICDM*. IEEE, 837–846.
[20] Xiangmin Zhou and Lei Chen. 2014. Event detection over twitter social media streams. *The VLDB journal* 23, 3 (2014), 381–400.