An Unsupervised Framework of Exploring Events on Twitter: Filtering, Extraction and Categorization

Deyu Zhou and **Liangyu Chen**
School of Computer Science and Engineering
Southeast University, China
d.zhou@seu.edu.cn, cly1cn@126.com

Yulan He
School of Engineering and Applied Science
Aston University, UK
y.he@cantab.net

Abstract

Twitter, as a popular microblogging service, has become a new information channel for users to receive and exchange the most up-to-date information on current events. However, since there is no control on how users can publish messages on Twitter, finding newsworthy events from Twitter becomes a difficult task like "finding a needle in a haystack".

In this paper we propose a general unsupervised framework to explore events from tweets, which consists of a pipeline process of filtering, extraction and categorization. To filter out noisy tweets, the filtering step exploits a lexicon-based approach to separate tweets that are event-related from those that are not. Then, based on these event-related tweets, the structured representations of events are extracted and categorized automatically using an unsupervised Bayesian model without the use of any labelled data. Moreover, the categorized events are assigned with the event type labels without human intervention. The proposed framework has been evaluated on over 60 millions tweets which were collected for one month in December 2010. A precision of 70.49% is achieved in event extraction, outperforming a competitive baseline by nearly 6%. Events are also clustered into coherence groups with the automatically assigned event type label.

Introduction

With the increasing popularity of social media, social networking sites such as Twitter have become an important source of event information. As reported in (Petrovic et al. 2013), even 1% of the public stream of Twitter contains around 95% of all the events reported in the newswire. Twitter appears to cover nearly all newswire events, with some sports related events being only reported in Twitter because these events have value for a short period of time or to a very restricted audience. Therefore, it is crucial to extract events from the social streams such as tweets.

Previous research in event extraction has focused largely on news articles. Event extraction techniques typically rely on the detection of event "triggers" with their arguments for slot filling in event frames. Classical approaches to event extraction can be roughly categorized into three classes, pattern based (Tanev, Piskorski, and Atkinson 2008), machine learning based (Piskorski et al. 2008) and a hybrid combining the previous two categories (Grishman, Westbrook, and Meyers 2005). Recently there has been much interest in event extraction from Twitter. Ritter et al. (2012) presented a system called TwiCal to extract and categorize events from Twitter. They relied on a sequence labeler trained from annotated data to extract event phrases from Twitter. In (Abdelhaq, Sengstock, and Gertz 2013), a system called EvenTweet was constructed to extract localized events from a stream of tweets in real-time. The extracted events are described by start time, location and a number of related keywords.

Compared to newswire text, the social stream data such as tweets have the following characteristics:

- Fragmented and Noisy. Social media messages are often short, contain a large number of irregular and ill-formed words, and evolve rapidly over time. Comparing to formal text such news articles, it is more challenging to process fragmented and noisy messages. Also, most social media messages are not event-related.
- Wide Variety. Social media data are produced continuously by a large and uncontrolled number of users. As such, it is not possible to know the event types a priori and hence violates the use of existing event extraction approaches which either rely on manually-defined linguistic patterns representing expert knowledge to extract events or make use of corpora annotated with event-specific information such as actors, date, place, etc., to learn event extraction patterns.
- Redundancy. For most newsworthy events, there may be high volume of redundant messages referring to the same event.

The aforementioned characteristics of social stream data pose new challenges but also provide opportunities to explore unsupervised approaches for event extraction and categorisation based on the redundancy property of event-related tweets. In this paper, we propose a general unsupervised framework to explore events from tweets. It consists of two steps, filtering, extraction and categorization. In the filtering step, a keyword lexicon built from news articles published in the same period as tweets is used to filter out non-event-related tweets. Then, an unsupervised Bayesian model called Latent Event & Category Model (LECM) is employed to extract and categorize structured representation of events from the event-related tweets. The model extends the previously proposed Latent Event Model (LEM) (Zhou, Chen, and He

2014) by automatically grouping events into categories organized by event types. Furthermore, each event category is assigned with an event type label without manual intervention. While the previously proposed LEM model has only been tested on a small dataset of just over 2,000 tweets, we evaluate the LECM model on a much larger dataset of 60 millions tweets.

The main contributions of the paper are summarized below:

- We have proposed an end-to-end framework for event extraction and categorization from large-scale Twitter data without the use of labeled data.
- We have developed an unsupervised Bayesian modelling approach for jointly extracting and categorizing events without human intervention. The extracted event groups are further assigned with event type labels automatically.
- We have evaluated our proposed framework on a large dataset consisting of over 60 million tweets and observed a significant improvement of nearly 6% in precision compared to the start-of-the-art open event extraction approach.

Related Work

Event Detection on Tweets

Instead of extracting structured representations of events, event detection is to discover new or previously unidentified events where each event refers to something that happens at certain time and place. Event detection has long been addressed in the Topic Detection and Tracking program sponsored by the Defense Advanced Research Projects Agency. The concept of event in event detection in news (Allan 2002) is defined as real-world occurrence e with an associated time period T_e and a time-ordered stream of news messages M_e, of substantial volume, discussing the occurrence and published during time T_e. There has been some recent work on detecting events or tracking topics on Twitter. Sankaranarayanan et al. (2009) detected breaking news from tweets to build a news processing system, called TwitterStand. A naive Bayes classifier was employed to separate news from irrelevant information and an online clustering algorithm was used to group tweets into different clusters. Sakaki et al. (2010) trained a classifier based on features derived from individual tweets (e.g., the keywords in a tweet and the number of words it contains) to detect a particular type of event such as earthquakes and typhoons. They formulated event detection as a classification problem and trained a Support Vector Machine (SVM) on a manually labeled Twitter dataset comprising positive events (earthquakes and typhoons) and negative events (other events or non-events). In (Popescu, Pennacchiotti, and Paranjpe 2011), a pattern based approach was employed to automatically detect events involving known entities from Twitter. Becker et al. (2011) focused on online identification of real-world event content and its associated Twitter messages using an online clustering technique, which continuously clusters similar tweets and then classifies the clusters content into real-world events or non-events. Lee and Sumiya (2010) proposed a geo-social event detection system based on modeling and monitoring crowd behaviors via Twitter, to identify local festivals. A brief overview of event detection techniques applied to Twitter can be found in (Atefeh and Khreich 2013).

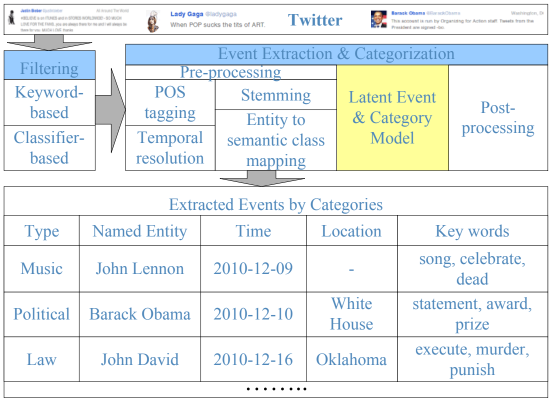
Event Extraction on Tweets

In recent years, there have been increasing interests in exploring event extraction from Twitter. Benson et al. (2011) proposed a graphical model to extract canonical entertainment events from tweets by aggregating information across multiple messages. In (Liu et al. 2012), social events are extracted from multiple similar tweets using a factor graph by harvesting the redundancy in tweets. Ritter et al. (2012) presented a system called TwiCal to extract and categorize events from Twitter. It requires some annotated tweets to train a sequence labeler based on Conditional Random Fields to extract event-related phrases from tweets. In (Abdelhaq, Sengstock, and Gertz 2013), localized events were extracted from a stream of tweets in real-time. The extracted events are described by a number of related keywords, start time and the location.

Our work is similar to TwiCal in the sense that we also focus on the extraction and categorization of structured representation of events from Twitter. However, TwiCal relies on a supervised sequence labeler trained on tweets annotated with event mentions for the identification of event-related phrases. We propose a simple Bayesian modelling approach which is able to directly extract event-related keywords from tweets without supervised learning. Also, TwiCal uses G^2 test to choose an entity y with the strongest association with a date d to form a binary tuple $\langle y, d \rangle$ to represent an event. On the contrary, the structured representation of events can be directly extracted from the output of our LECM model. Moreover, the extracted events are categorized and assigned with event type labels automatically in the proposed framework. We have conducted experiments on a Twitter corpus and the results show that our proposed approach outperforms TwiCal, the state-of-the-art open event extraction system, by nearly 6% in precision.

Methodologies

Given a raw stream of Twitter, irrelevant or noisy tweets are filtered out firstly. Only tweets which are more likely describing events are kept. Afterwards, a Bayesian model called Latent Event & Category model (LECM) is employed to extract events and group them in different categories. Here, an event is represented as a tuple $\langle y, d, l, k \rangle$ where y stands for non-location named entities, d for a date, l for a location, and k for event-related keywords. Each event mentioned in tweets can be closely depicted by this representation. It should be noted that for some events, one or more elements in their corresponding tuples might be absent since the information relating to certain event elements might not be available in tweets. As illustrated in Figure 1, our proposed framework consists of two main steps, filtering and event extraction and categorization based on the LECM model. The details of our proposed framework are described below.


Lady Gaga @ladygaga
When POP sucks the tits of ART.
Twitter
Barack Obama @BarackObama
Filtering
Keyword-based
Classifier-based
Event Extraction & Categorization
Pre-processing
POS tagging
Temporal resolution
Stemming
Entity to semantic class mapping
Latent Event & Category Model
Post-processing

Extracted Events by Categories

Type	Named Entity	Time	Location	Key words
Music	John Lennon	2010-12-09	-	song, celebrate, dead
Political	Barack Obama	2010-12-10	White House	statement, award, prize
Law	John David	2010-12-16	Oklahoma	execute, murder, punish
....				

Figure 1: The proposed framework for exploring event from Twitter.

Tweet Filtering

Two approaches have been explored for filtering tweets. The first approach is through lexicon matching. We first build a lexicon which contains keywords extracted from news articles published around the same period as tweets. We then only keep the tweets containing words that can be found in the lexicon.

Apart from the keyword-based approach, we have investigated another approach which casts tweet filtering as a binary classification problem. Given a set of tweets $M = (m_1, ..., m_k)$, the classifier outputs a class label $C \in \{\text{event, non-event}\}$. The non-event tweets are removed at this stage. To build a good classifier, it is crucial to design a proper feature set. Considering that the number of event-related tweets is significantly less than non-event-related tweets, we propose to construct a feature set in the following way.

- *Binary word features*. We select words occurred more frequently in event-related tweets but rarely in non-event tweets as highly class-indicative features to build our feature set. The importance score of a word is defined as TFP/TFN, where TFP is the term frequency in the event-related tweets while TFN is the term frequency in non-event tweets. We sort the words by their importance scores and only select the top n words to construct binary features (presence of the word or not). In the experiments discussed in the paper, n is set to 100 empirically.
- *Other event-related features*. We notice that tweets containing information related to authoritative news agencies such as CNN or BBC and some phrases such as "breaking news" most likely describe real-world events. As such, we also include binary features indicating the presence of news agencies and some manually selected indicative phrases. Furthermore, we add other binary features (Sriram et al. 2010) which consist of time-related phrases, opinionated words, currency and percentage signs, URLs, reply to other users such as "@username", etc.
- *Event elements*. As an event is described as "something that happens at a given place and time", the presence of named entity, location, and time information could be potentially useful to determine the occurrence of an event in text. Hence, they are also used as features to train a binary classifier.

Event Extraction and Categorization

Events in the framework are represented as a 4-tuple $\langle y, d, l, k\rangle$, where y stands for non-location named entities, d for a date, l for a location, and k for event-related keywords. The event extraction and categorization component follows three steps as illustrated in Figure 1: pre-processing, event extraction and categorization using the proposed LECM model, post-processing. The details of each step are presented below.

Pre-processing Tweets are pre-processed by time expression resolution, named entity recognition, part-of-speech (POS) tagging and stemming, and finally the mapping of named entities to semantic concepts.

As Twitter users might represent the same date in various forms, SUTime[1] (Chang and Manning 2012) is employed to resolve the ambiguity of time expressions. For example, temporal expressions such as "tomorrow" and "last Friday" are mapped to a specific date based on the tweet's publish date. Named entity recognition (NER) is a crucial step since the results would directly impact the final extracted 4-tuple $\langle y, d, l, k\rangle$. It is not easy to accurately identify named entities in the Twitter data since tweets contain a lot of misspellings and abbreviations. A named entity tagger trained specifically on the Twitter data[2] (Ritter et al. 2011) is used to directly extract named entities from tweets. A POS tagger[3] trained on tweets (Gimpel et al. 2011) is used to perform POS tagging on the tweets and apart from the previously recognised named entities, only words tagged with nouns, verbs or adjectives are kept. These remaining words are subsequently stemmed and word occurred less than 3 times are filtered. We use the API provided by Freebase[4] to map named entities to semantic classes. This is to provide a certain level of abstraction of named entities. For example, "Celine Dion" and "Justin Bieber" could be mapped to the "music" class. For named entities with more than one semantic class, we simply chose the one with the highest relevance score.

Latent Event & Category Model To extract events in tweets and group events into categories, an unsupervised latent variable model, called Latent Event & Category Model (LECM), is proposed to extract and cluster event instances. It is assumed that in the model, each tweet message $m \in \{1..M\}$ is assigned to one event instance e, while e is modeled as a joint distribution over the named entities y, the date/time d when the event occurred, the location l where the event occurred and the event-related keywords k. This assumption essentially encourages events that involve the same named entities, occur at the same time and in the same location and have the same keywords to be assigned with the

[1] http://nlp.stanford.edu/software/sutime.shtml

[2] http://github.com/aritter/twitter-nlp

[3] http://www.ark.cs.cmu.edu/TweetNLP

[4] http://www.freebase.com/

same event. As the event distribution is shared across social media posts with the same named entities, dates, locations and keywords, it essentially preserve the ambiguity that for example, events comprising the same date and location may or may not belong to the same event. It is also assumed that each event e is assigned to one event type t, while t is modeled as a joint distribution over the semantic classes y' to which the named entities are mapped and the event-related keywords k. This assumption essentially encourages events that involve the same semantic class and have similar keyword to be categorized into the same event type. The graphical model of LECM is shown in Figure 2.

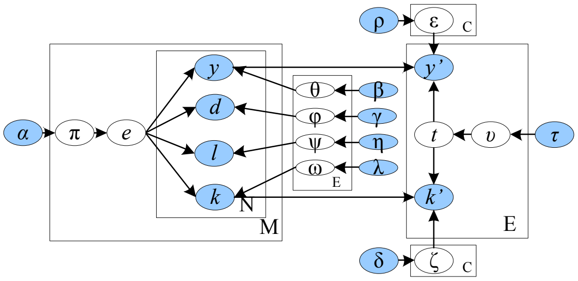


Figure 2: LECM: a latent variable model for event extraction and categorization.

The generative process of LECM is shown below.

- Draw the event distribution $\boldsymbol{\pi}_e \sim$ Dirichlet(α).
- Draw the event type distribution $\boldsymbol{\upsilon} \sim$ Dirichlet(τ).
- For each event $e \in \{1..E\}$, draw multinomial distributions $\boldsymbol{\theta}_e \sim$ Dirichlet$(\beta), \boldsymbol{\varphi}_e \sim$ Dirichlet$(\gamma), \boldsymbol{\psi}_e \sim$ Dirichlet$(\eta), \boldsymbol{\omega}_e \sim$ Dirichlet(λ).
- For each event type $t \in \{1..C\}$, draw multinomial distributions $\boldsymbol{\epsilon}_t \sim$ Dirichlet$(\rho), \boldsymbol{\zeta}_t \sim$ Dirichlet(δ).
- For each tweet $\mathbf{w}$
  - Choose an event $e \sim$ Multinomial($\boldsymbol{\pi}$),
  - For each named entity occur in tweet m, choose a named entity $y \sim$ Multinomial($\boldsymbol{\theta}_e$),
  - For each date occur in tweet m, choose a date $d \sim$ Multinomial($\boldsymbol{\varphi}_e$),
  - For each location occur in tweet m, choose a location $l \sim$ Multinomial($\boldsymbol{\psi}_e$),
  - For other word positions, choose a word $k \sim$ Multinomial($\boldsymbol{\omega}_e$).
- For each event $\mathbf{e}$
  - Choose an event type $t \sim$ Multinomial($\boldsymbol{\upsilon}$),
  - For each named entity occur in event e, choose a semantic class $y' \sim$ Multinomial($\boldsymbol{\epsilon}_t$),
  - For each keyword in event e, choose a keyword $k' \sim$ Multinomial($\boldsymbol{\zeta}_t$).

Parameter Estimation We use Collapsed Gibbs Sampling (Griffiths and Steyvers 2004) to infer the parameters of the model and the latent class assignments for events and categories, given observed data $\mathcal{D}$ and the total likelihood. Gibbs sampling allows us repeatedly sample from a Markov chain whose stationary distribution is the posterior of e_m, t_e from the distribution over that variable given the current values of all other variables and the data. Such samples can be used to empirically estimate the target distribution.

Letting the subscript $-m$ denote a quantity that excludes data from mth tweet, the conditional posterior for e_m is:

$$P(e_m = e|\mathbf{e}_{-m}, \mathbf{y}, \mathbf{d}, \mathbf{l}, \mathbf{k}, \Lambda) \propto \frac{n_e^{-m}+\alpha}{M+E\alpha}\times$$
$$\prod_{y=1}^{Y}\frac{\prod_{b=1}^{n_{e,y}^{(m)}}(n_{e,y}-b+\beta)}{\prod_{b=1}^{n_e^{(m)}}(n_e-b+Y\beta)} \times \prod_{d=1}^{D}\frac{\prod_{b=1}^{n_{e,d}^{(m)}}(n_{e,d}-b+\gamma)}{\prod_{b=1}^{n_e^{(m)}}(n_e-b+D\gamma)}$$
$$\times \prod_{l=1}^{L}\frac{\prod_{b=1}^{n_{e,l}^{(m)}}(n_{e,l}-b+\eta)}{\prod_{b=1}^{n_e^{(m)}}(n_e-b+L\eta)} \times \prod_{k=1}^{V}\frac{\prod_{b=1}^{n_{e,k}^{(m)}}(n_{e,k}-b+\lambda)}{\prod_{b=1}^{n_e^{(m)}}(n_e-b+V\lambda)}$$

where n_e is the number of tweets that have been assigned to the event e; M is the total number of tweets, $n_{e,y}$ is the number of times named entity y has been associated with event e; $n_{e,d}$ is the number of times dates d has been associated with event e; $n_{e,l}$ is the number of times locations l has been assigned with event e; $n_{e,k}$ is the number of times keyword k has associated with event e, counts with (m) notation denote the counts relating to tweet m only. Y, D, L, V are the total numbers of distinct named entities, dates, locations, and words appeared in the whole Twitter corpus respectively. E is the total number of events which needs to be set.

Letting the subscript $-e$ denote a quantity that excludes data from eth event , the conditional posterior for t_e is:

$$P(t_e = t|\mathbf{t}_{-e}, \mathbf{y}', \mathbf{k}', \Lambda) \propto \frac{\tau + n_t^{-e}}{E + C\tau}$$
$$\times \prod_{\tilde{y}\in Y_e}\frac{\rho + n_{t,\tilde{y}}^{-e}}{\sum_{y'=1}^{S} n_{t,y'}^{-e} + S\rho} \times \prod_{\tilde{k}\in K_e}\frac{\delta + n_{t,\tilde{k}}^{-e}}{\sum_{k'=1}^{V} n_{t,k'}^{-e} + V\delta}$$

where C is the number of the event types, Y_e is the set of y' belonging to e, $n_{t,y'}$ is the times of non-location entity' semantic class y' being assigned with event type t, K_e is the set of k' belonging to e, $n_{t,k'}$ is the times of the keyword k' being assigned with event type t and S is the total number of distinct non-location named entities' semantic classes appeared in the whole Twitter corpus respectively.

Post-processing To improve the precision of event extraction, we remove the least confident event element from the 4-tuple using the following rules.

- If N(element) $< n_1$, the element will be removed from the extracted results. Here, N(element) is the number of occurrence of the element in the tweets with event e.
- If N(element) $> m/n_2$, the element will be kept. Here, m is the number of tweets with event e.

Here, n_1, n_2 are the thresholds to be set to 7 and 5 empirically.

Our model automatically groups events into different event clusters. For each event cluster, the most prominent semantic class obtained based on the event entities in the cluster is employed as the event type of the event cluster.

Table 1: Statistics of the datasets used in the experiments.

Dataset	Property	Value
I	Source	Manually annotated
	#Event-related tweets	2,891
	#Non-event-related tweets	26,000
II	Source	Twitter Streaming API
	#Tweets	60,000,000
	Time-Range	2010-12-01→ 2010-12-31

Experiments

In this section, we firstly describe the datasets used in our experiments. We then present the steps taken to evaluate our system and introduce the baseline system for comparison. Finally, we present the experimental results.

Setup

We built two datasets from tweets collected in the month of December in 2010. Dataset I contains manually annotated event-related or not related tweets for the training of a binary classifier in the filtering step. Tweets are annotated as event-related if relevant news articles can be found in the one-week window before and after the tweets' publication dates. We argue that this is a reasonable choice since newsworthy events would be more interesting than others. Dataset II [5] contains 60 millions unlabelled tweets which are used to evaluate the proposed framework. Table 1 reports the statistics of these two datasets.

The evaluation is conducted in three aspects: filtering, extraction and categorization. For tweet filtering, as most tweets in Dataset I and II are not event-related, we only report the performance of classifying event-related tweets. Precision is defined as the proportion of the correctly identified event-related tweets out of the system returned event-related tweets. Recall is defined as the proportion of correctly identified true event-related tweets.

For the evaluation of event extraction results, since it is almost impossible to know exactly how many true events in Dataset II due to the large volume of tweets it contains, we only report the precision of our event extraction results. For the 4-tuple $\langle y, d, l, k \rangle$, the precision value is calculated based on the following criteria:

1. Do the entity y, location l and date d that we have extracted refer to the same event?
2. Are the keywords k in accord with the event that other extracted elements y, l, d refer to and are they informative enough to tell us what happened?

The baseline we chose is TwiCal (Ritter et al. 2012), the state-of-the-art open event extraction system on tweets. The events extracted in the baseline are represented as a 3-tuple $\langle y, d, k \rangle$, where y stands for a non-location named entity, d for a date and k for an event phrase. We re-implemented the whole system and evaluate the performance of the baseline on the correctness of the exacted three elements only excluding the location element.

[5] http://cse.seu.edu.cn/people/zhoudeyu/AAAI2015-data.zip

Table 2: Tweet filtering results of classifying event-related tweets on Dataset I.

Method	Precision (%)	Recall (%)	F-measure (%)
Keyword-based	73.03	25.73	38.05
SVM-based	81.65	11.22	19.73

Table 3: Comparison of the performance of event extraction on Dataset II.

Method	Precision
Without Filtering	28.33%
With Filtering	70.49%
TwiCal	64.28%
the Proposed Framework	70.49%

The performance of event categorization is difficult to evaluate due to the significantly large number of tweets involved in our experiments. As such, we only present some event categorization examples to illustrate the feasibility of our proposed framework.

Experimental Results

Tweet Filtering As previously discussed, we have explored both keyword-based and classifier-based approaches for tweet filtering. For classifier-based approach, we use Weka (Hall et al. 2009) to train an SVM with default parameters on Dataset I and perform 3-fold cross validation. The results are shown in Table 2.

Since most tweets in Dataset I are not event-related, it makes sense to only report the results on the event-related class. The performance obtained here is comparable to the state-of-the-art results on tweet classification (Sriram et al. 2010). It can be observed that the SVM-based approach achieves higher precision but with much lower recall rate. It might be attributed to the highly imbalanced training data in Dataset I where only about 10% tweets are event-related. We also tested both keyword-based and SVM-based approaches on Dataset II. Due to the large size of Dataset II, it is impossible to find out the actual performance of both approaches. We instead randomly selected 1,000 tweets identified as event-related by each approach and manually checked the accuracy. We found that the keyword-based approach gives higher precision compared to the SVM-based approach. As such, we chose to use the keyword-based approach for tweet filtering in all the subsequent experiments.

Table 4: Examples of the extracted events using the framework with or without filtering.

Entity	Keywords
Extracted events without filtering	
Harry Potter	like, watch, movie
God	thank, wish, love
Lady Gaga	star, nightlife, blog
Justin Bieber	club, music, photo
Extracted events with filtering	
Windows Phone	os, release, mango
Philadelphia Eagles, Ubalo Jimenez	sign, championship, sense
Amy winehouse	death, RIP, sad

Table 5: Examples of event categorization results. The event type labels are automatic assigned using the most frequent semantic class for each event type.

Event Type	Event			
	Entity	Location	Date	Keywords
Goverment	Barack Obama	White House	2010-12-09	interview, economy, focus
	Senate	-	2010-12-01	block, legislation, repeal
	Obama	Oslo	2010-12-10	statement, award, prize
	Dmitry Medvedev, President Obama	Russia	2010-12-23	congratulate,laud, treaty
Music	Justin Bieber	-	2010-12-23	song, leak, new
	Lady Gaga	Germany	2010-12-03	song, steal, investigate
	John Lennon	-	2010-12-09	song, celebrate, dead
Sports	Adrian Gonzalez	-	2010-12-04	trade, talk, espn
	LeBron James	-	2010-12-08	score, point, win
	Mike Brown, ESPN	-	2010-12-24	coach,join,analyst
	Diana Taurasi, Phoenix Mercury	-	2010-12-24	positive,ban,test
Business	Microsoft	Washington	2010-12-05	co-founder, give, million
	Microsoft	Germany	2010-12-28	deny, kinect, quadruple
	Microsoft, Internet Explorer	-	2010-12-24	warn ,vulnerability, blast
	Google	-	2010-12-24	track, santa
	Google	-	2010-12-07	give, nexus, s
Law	High Court	-	2010-12-10	verdict, gay, change
	Supreme Court	-	2010-12-07	law, case, hear
	Supreme Court	-	2010-12-06	hear, law, punish
	John David	Oklahoma	2010-12-16	execute, murder, punish
TV	Ryan Reynolds, Scarlett Johansson	LA	2010-12-24	divorce, file, official
	Kathy Griffin, Chelsea Handler	-	2010-12-08	fire bully accusation
	Shelley Malil	-	2010-12-16	sentence, life, prison

Event Extraction After the filtering step, we are left with less than 250,000 tweets in Dataset II. These tweets are fed into LECM for event extraction and categorization. The event extraction precisions on Dataset II are presented in Table 3. In our experiments, the number of events is set to 400 which was chosen using the perplexity measure on the 10% held-out set from Dataset II. It can be observed that the filtering step is really crucial to event extraction. By filtering out non-event-related tweets, the precision of our event extraction component increases dramatically from 28.33% to 70.49%. When compared against the baseline approach, TwiCal, it can be observed from Table 3 that our proposed framework significantly outperforms the baseline with nearly 6% improvement on precision. One possible reason is that in a large scale Twitter data such as Dataset II, tweets with temporal keywords are rare and many event-related tweets have no date information. As such, TwiCal which relies on the association between named entities and dates for event extraction fails to handle tweets with no date information. On the contrary, our proposed model is flexible and allows date information to be missed in event tweets. Also, TwiCal assumes that one event has only one named entity, which is not true in some cases. For example, in the tweet "Russian President Dmitry Medvedev on Thursday congratulated President Barack Obama on the Senate's approval of a new nuclear arms control treaty between the countries", both "Dmitry Medvedev" and "Barack Obama" are involved. Our proposed approach does not impose such a constraint.

To further understand the effect of our filtering step, examples of the events extracted using our proposed framework with and without filtering are presented in Table 4. It can be observed that without filtering, some extracted events are not really newsworthy events although they also contain named entities and meaningful keywords. For example, there are many tweets talking about watching the movie "Harry Potter". But these tweets are not relating to newsworthy events.

Event Categorization The event extraction and categorization component automatically clusters events into different event types. We empirically set the number of event types to 25 in the LECM model. Some example event categorization results are presented in Table 5. It can be observed from the example results that our event categorization component does group similar events together. Moreover, the event type label assigned to each cluster is quite meaningful.

Conclusions and Future Work

In this paper, we have proposed an unsupervised framework to explore events from tweets. A pipeline process consists of filtering, extraction and categorization is introduced. All the steps here are fully unsupervised, which makes our proposed framework specifically plausible for analyzing events in the large-scale social stream data. The proposed framework has been evaluated on a large Twitter data consisting of 60 million tweets and has achieved a precision of 70.49%, comfortably outperforming a baseline by 6%. A possible future direction is to build a unified framework for filtering and event extraction and categorization simultaneously in order to reduce the error propagated in the pipeline process.

Acknowledgments

The authors thank the anonymous reviewers for their insightful comments. This work was funded by the National Natural Science Foundation of China (61103077), Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry, the Fundamental Research Funds for the Central Universities (the Cultivation Program for Young Faculties of Southeast University), and the start-up grant from Aston University, UK.

References

Abdelhaq, H.; Sengstock, C.; and Gertz, M. 2013. Eventweet: Online localized event detection from twitter. *Proceedings of the VLDB Endowment* 1326–1329.

Allan, J., ed. 2002. *Topic Detection and Tracking: Event-based Information Organization.* Norwell, MA, USA: Kluwer Academic Publishers.

Atefeh, F., and Khreich, W. 2013. A survey of techniques for event detection in twitter. *Computational Intelligence*.

Becker, H.; Naaman, M.; and Gravano, L. 2011. Beyond trending topics: Real-world event identification on twitter. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*.

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. Stroudsburg, PA, USA: Association for Computational Linguistics.

Chang, A. X., and Manning, C. D. 2012. Sutime: A library for recognizing and normalizing time expressions. In *8th International Conference on Language Resources and Evaluation (LREC 2012)*.

Gimpel, K.; Schneider, N.; O'Connor, B.; Das, D.; Mills, D.; Eisenstein, J.; Heilman, M.; Yogatama, D.; Flanigan, J.; and Smith, N. A. 2011. Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*, HLT '11, 42–47. Stroudsburg, PA, USA: Association for Computational Linguistics.

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):5228C5235.

Grishman, R.; Westbrook, D.; and Meyers, A. 2005. Nyu's english ace 2005 system description. In *ACE 05 Evaluation Workshop*.

Hall, M.; Frank, E.; Holmes, G.; Pfahringer, B.; Reutemann, P.; and Witten, I. H. 2009. The weka data mining software: An update. *SIGKDD Explorations* 11(1).

Lee, R., and Sumiya, K. 2010. Measuring geographical regularities of crowd behaviors for twitter-based geo-social event detection. In *Proceedings of the 2nd ACM SIGSPATIAL International Workshop on Location Based Social Networks*, LBSN '10, 1–10. New York, NY, USA: ACM.

Liu, X.; Zhou, X.; Fu, Z.; Wei, F.; and Zhou, M. 2012. Exacting social events for tweets using a factor graph. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, 1692–1698.

Petrovic, S.; Osborne, M.; McCreadie, R.; Macdonald, C.; Ounis, I.; and Shrimpton, L. 2013. Can twitter replace newswire for breaking news? In *Proceedings of the 7th International AAAI Conference on Weblogs and Social Media*.

Piskorski, J.; Tanev, H.; Atkinson, M.; and Van Der Goot, E. 2008. Cluster-centric approach to news event extraction. In *International Conference on New Trends in Multimedia and Network Information Systems*, 276–290.

Popescu, A.-M.; Pennacchiotti, M.; and Paranjpe, D. 2011. Extracting events and event descriptions from twitter. In *Proceedings of the 20th international conference companion on World Wide Web (WWW)*, 105–106.

Ritter, A.; Clark, S.; Etzioni, O.; et al. 2011. Named entity recognition in tweets: an experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 1524–1534. Association for Computational Linguistics.

Ritter, A.; Mausam; 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*, KDD '12, 1104–1112. New York, NY, USA: 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.

Sankaranarayanan, J.; Samet, H.; Teitler, B. E.; Lieberman, M. D.; and Sperling, J. 2009. Twitterstand: News in tweets. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '09, 42–51. New York, NY, USA: ACM.

Sriram, B.; Fuhry, D.; Demir, E.; Ferhatosmanoglu, H.; and Demirbas, M. 2010. Short text classification in twitter to improve information filtering. In *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '10, 841–842. New York, NY, USA: ACM.

Tanev, H.; Piskorski, J.; and Atkinson, M. 2008. Real-time news event extraction for global crisis monitoring. In *13th International Conference on Applications of Natural Language to Information Systems (NLDB)*, 207–218.

Zhou, D.; Chen, L.; and He, Y. 2014. A simple bayesian modelling approach to event extraction from twitter. In *Proceedings of the The 51st Annual Meeting of the Association for Computational Linguistics (ACL)*, 700–705.