A Simple Bayesian Modelling Approach to Event Extraction from Twitter

Deyu Zhou[†‡] **Liangyu Chen**[†] **Yulan He**[§]
[†] School of Computer Science and Engineering, Key Laboratory of Computer Network
and Information Integration, Ministry of Education, Southeast University, China
[‡] State Key Laboratory for Novel Software Technology, Nanjing University, China
[§] School of Engineering and Applied Science, Aston University, UK
d.zhou@seu.edu.cn, cly1cn@126.com, y.he@cantab.net

Abstract

With the proliferation of social media sites, social streams have proven to contain the most up-to-date information on current events. Therefore, it is crucial to extract events from the social streams such as tweets. However, it is not straightforward to adapt the existing event extraction systems since texts in social media are fragmented and noisy. In this paper we propose a simple and yet effective Bayesian model, called Latent Event Model (LEM), to extract structured representation of events from social media. LEM is fully unsupervised and does not require annotated data for training. We evaluate LEM on a Twitter corpus. Experimental results show that the proposed model achieves 83% in F-measure, and outperforms the state-of-the-art baseline by over 7%.

1 Introduction

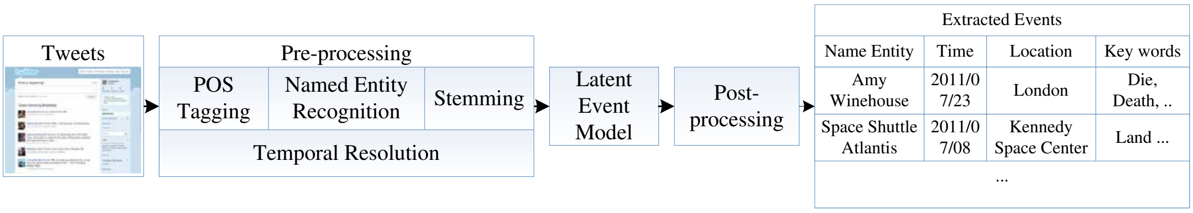
Event extraction is to automatically identify events from text with information about *what* happened, *when*, *where*, to *whom*, and *why*. Previous work in event extraction has focused largely on news articles, as the newswire texts have been the best source of information on current events (Hogenboom et al., 2011). Approaches for event extraction include knowledge-based (Piskorski et al., 2007; Tanev et al., 2008), data-driven (Piskorski et al., 2008) and a combination of the above two categories (Grishman et al., 2005). Knowledge-based approaches often rely on linguistic and lexicographic patterns which represent expert domain knowledge for particular event types. They lack the flexibility of porting to new domains since extraction patterns often need to be re-defined. Data-driven approaches require large annotated data to train statistical models that approximate linguistic phenomena. Nevertheless, it is expensive to obtain annotated data in practice.

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. Nevertheless, the social stream data such as Twitter data pose new challenges. Social media messages are often short and evolve rapidly over time. As such, it is not possible to know the event types a priori and hence violates the use of existing event extraction approaches.

Approaches to event extraction from Twitter make use of a graphical model to extract canonical entertainment events from tweets by aggregating information across multiple messages (Benson et al., 2011). In (Liu et al., 2012), social events involving two persons 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 which extracts an open-domain calendar of significant events represented by a 4-tuple set including a named entity, event phrase, calendar date, and event type from Twitter.

In our work here, we notice a very important property in social media data that the same event could be referenced by high volume messages. This property allows us resort to statistical models that can group similar events based on the co-occurrence patterns of their event elements. Here, event elements include named entities such as person, company, organization, date/time, location, and the relations among them. We can treat an event as a latent variable and model the generation of an event as a joint distribution of its individual event elements. We thus propose a Latent Event Model (LEM) which can automatically detect events from social media without the use of labeled data.

Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Short Papers), pages 700–705,
Baltimore, Maryland, USA, June 23-25 2014. ©2014 Association for Computational Linguistics



Name Entity	Time	Location	Key words
Amy Winehouse	2011/07/23	London	Die, Death, ..
Space Shuttle Atlantis	2011/07/08	Kennedy Space Center	Land ...
...			

Figure 1: The proposed framework for event extraction from tweets.

Our work is similar to TwiCal in the sense that we also focus on the extraction 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 LEM model. 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 7.7% in F-measure.

2 Methodology

Events extracted in our proposed framework are represented as a 4-tuple $\langle y, d, l, k \rangle$, where y stands for a non-location named entity, d for a date, l for a location, and k for an event-related keyword. 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 as their related information is not available in tweets. As illustrated in Figure 1, our proposed framework consists of three main steps, pre-processing, event extraction based on the LEM model and post-processing. The details of our proposed framework are described below.

2.1 Pre-processing

Tweets are pre-processed by time expression recognition, named entity recognition, POS tagging and stemming.

Time Expression Recognition. Twitter users might represent the same date in various forms. For example, “tomorrow”, “next Monday”, “ August 23th” in tweets might all refer to the same day, depending on the date that users wrote the tweets. To resolve the ambiguity of the time expressions, SUTime[1] (Chang and Manning, 2012) is employed, which takes text and a reference date as input and outputs a more accurate date which the time expression refers to.

Named Entity Recognition. 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. However, it is often observed that events mentioned in tweets are also reported in news articles in the same period (Petrovic et al., 2013). Therefore, named entities mentioned in tweets are likely to appear in news articles as well. We thus perform named entity recognition in the following way. First, a traditional NER tool such as the Stanford Named Entity Recognizer[2] is used to identify named entities from the news articles crawled from BBC and CNN during the same period that the tweets were published. The recognised named entities from news are then used to build a dictionary. Named entities from tweets are extracted by looking up the dictionary through fuzzy matching. We have also used a named entity tagger trained specifically on the Twitter data[3] (Ritter et al., 2011) to directly extract named entities from tweets. However, as will be shown in Section 3 that using our constructed dictionary for named entity extraction gives better results. We distinguish between location entities, denoted as l, and non-location entities such as person or organization, denoted as y.

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

[2] http://nlp.stanford.edu/software/CRF-NER.shtml

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

Finally, we use a POS tagger[4] trained on tweets (Gimpel et al., 2011) to perform POS tagging on the tweets data 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 words occurred less than 3 times are filtered.

After the pre-processing step, non-location entities y, locations l, dates d and candidate keywords of the tweets are collected as the input to the LEM model for event extraction.

2.2 Event Extraction using the Latent Event Model (LEM)

We propose an unsupervised latent variable model, called the Latent Event Model (LEM), to extract events from tweets. The graphical model of LEM is shown in Figure 2.

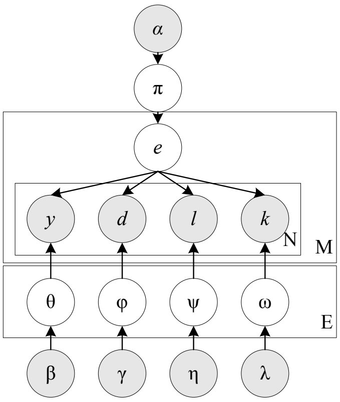


Figure 2: Laten Event Model (LEM).

In this model, we assume that 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 similar keyword to be assigned with the same event.

The generative process of LEM is shown below.

- Draw the event distribution $\boldsymbol{\pi}_e \sim$ Dirichlet(α)
- For each event $e \in \{1..E\}$, draw multinomial distributions $\boldsymbol{\theta}_e \sim$ Dirichlet(β), $\boldsymbol{\varphi}_e \sim$ Dirichlet(γ), $\boldsymbol{\psi}_e \sim$ Dirichlet(η), $\boldsymbol{\omega}_e \sim$ Dirichlet(λ).
- For each tweet $\boldsymbol{w}$
  - Choose an event $e \sim$ Multinomial($\boldsymbol{\pi}$),
  - For each named entity occur in tweet $\boldsymbol{w}$, choose a named entity $y \sim$ Multinomial($\boldsymbol{\theta}_e$),
  - For each date occur in tweet $\boldsymbol{w}$, choose a date $d \sim$ Multinomial($\boldsymbol{\varphi}_e$),
  - For each location occur in tweet $\boldsymbol{w}$, choose a location $l \sim$ Multinomial($\boldsymbol{\psi}_e$),
  - For other words in tweet $\boldsymbol{w}$, choose a word $k \sim$ Multinomial($\boldsymbol{\omega}_e$).

We use Collapsed Gibbs Sampling (Griffiths and Steyvers, 2004) to infer the parameters of the model and the latent class assignments for events, 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 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:

$$P(e_m = t|\boldsymbol{e}_{-m}, \boldsymbol{y}, \boldsymbol{d}, \boldsymbol{l}, \boldsymbol{z}, \Lambda) \propto \frac{n_t^{-m} + \alpha}{M + E\alpha} \times$$

$$\prod_{y=1}^{Y} \frac{\prod_{b=1}^{n_{t,y}^{(m)}} (n_{t,y} - b + \beta)}{\prod_{b=1}^{n_t^{(m)}} (n_t - b + Y\beta)} \times \prod_{d=1}^{D} \frac{\prod_{b=1}^{n_{t,d}^{(m)}} (n_{t,d} - b + \gamma)}{\prod_{b=1}^{n_t^{(m)}} (n_t - b + D\gamma)}$$

$$\times \prod_{l=1}^{L} \frac{\prod_{b=1}^{n_{t,l}^{(m)}} (n_{t,l} - b + \eta)}{\prod_{b=1}^{n_t^{(m)}} (n_t - b + L\eta)} \times \prod_{k=1}^{V} \frac{\prod_{b=1}^{n_{t,k}^{(m)}} (n_{t,k} - b + \lambda)}{\prod_{b=1}^{n_t^{(m)}} (n_t - b + V\lambda)}$$

where n_t is the number of tweets that have been assigned to the event t; M is the total number of tweets, $n_{t,y}$ is the number of times named entity y has been associated with event t; $n_{t,d}$ is the number of times dates d has been associated with event t; $n_{t,l}$ is the number of times locations l has been assigned with event t; $n_{t,k}$ is the number of times keyword k has associated with event t, 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.

Once the class assignments for all events are known, we can easily estimate the model parameters $\{\boldsymbol{\pi}, \boldsymbol{\theta}, \boldsymbol{\varphi}, \boldsymbol{\psi}, \boldsymbol{\omega}\}$. We set the hyperparameters $\alpha = \beta = \gamma = \eta = \lambda = 0.5$ and run Gibbs

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

sampler for 10,000 iterations and stop the iteration once the log-likelihood of the training data converges under the learned model. Finally we select an entity, a date, a location, and the top 2 keywords of the highest probability of every event to form a 4-tuple as the representation of that event.

2.3 Post-processing

To improve the precision of event extraction, we remove the least confident event element from the 4-tuples using the following rule. If $P(\text{element})$ is less than $\frac{1}{\xi}P(S)$, where $P(S)$ is the sum of probabilities of the other three elements and ξ is a threshold value and is set to 5 empirically, the element will be removed from the extracted results.

3 Experiments

In this section, we first describe the Twitter corpus used in our experiments and then present how we build a baseline based on the previously proposed TwiCal system (Ritter et al., 2012), the state-of-the-art open event extraction system on tweets. Finally, we present our experimental results.

3.1 Dataset

We use the First Story Detection (FSD) dataset (Petrovic et al., 2013) in our experiment. It consists of 2,499 tweets which are manually annotated with the corresponding event instances resulting in a total of 27 events. The tweets were published between 7th July and 12th September 2011. These events cover a range of categories, from celebrity news to accidents, and from natural disasters to science discoveries. It should be noted here that some event elements such as location is not always available in the tweets. Automatically inferring geolocation of the tweets is a challenging task and will be considered in our future work. For the tweets without time expressions, we used the tweets' publication dates as a default. The number of tweets for each event ranges from 2 to around 1000. We believe that in reality, events which are mentioned in very few tweets are less likely to be significant. Therefore, the dataset was filtered by removing the events which are mentioned in less than 10 tweets. This results in a final dataset containing 2468 tweets annotated with 21 events.

3.2 Baseline construction

The baseline we chose is TwiCal (Ritter et al., 2012). The events extracted in the baseline are represented as a 3-tuple $\langle y, d, k\rangle$[5], where y stands for a non-location named entity, d for a date and k for an event phrase. We re-implemented the system and evaluate the performance of the baseline on the correctness of the exacted three elements excluding the location element. In the baseline approach, the tuple $\langle y, d, k\rangle$ are extracted in the following ways. Firstly, a named entity recognizer (Ritter et al., 2011) is employed to identify named entities. The TempEx (Mani and Wilson, 2000) is used to resolve temporal expressions. For each date, the baseline approach chose the entity y with the strongest association with the date and form the binary tuple $\langle y, d\rangle$ to represent an event. An event phrase extractor trained on annotated tweets is required to extract event-related phrases. Due to the difficulties of re-implementing the sequence labeler without knowing the actual features set and the annotated training data, we assume all the event-related phrases are identified correctly and simply use the event trigger words annotated in the FSD corpus as k to form the event 3-tuples. It is worth noting that the F-measure reported for the event phrase extraction is only 64% in the baseline approach (Ritter et al., 2012).

3.3 Evaluation Metric

To evaluate the performance of the propose approach, we use $precison$, $recall$, and $F-measure$ as in general information extraction systems (Makhoul et al., 1999). For the 4-tuple $\langle y, d, l, k\rangle$, the precision 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?

If the extracted representation does not contain keywords, its precision is calculated by checking the criteria 1. If the extracted representation contains keywords, its precision is calculated by checking both criteria 1 and 2.

3.4 Experimental Results

The number of events, E, in the LEM model is set to 25. The performance of the proposed

[5] TwiCal also groups event instances into event types such as "Sport" or "Politics" using LinkLDA which is not considered here.

Method	Tuple Evaluated	Precision	Recall	F-measure
Baseline	$\langle y, d, k\rangle$	75%	76.19%	75.59%
Proposed	$\langle y, d, l\rangle$	96%	80.95%	87.83%
Proposed	$\langle y, d, l, k\rangle$	92%	76.19%	83.35%

Table 1: Comparison of the performance of event extraction on the FSD dataset.

Method	Tuple Evaluated	Precision	Recall	F-measure
TW-NER	$\langle y, d, l\rangle$	88%	76.19%	80.35%
TW-NER	$\langle y, d, l, k\rangle$	84%	76.19%	79.90%
NW-NER	$\langle y, d, l\rangle$	96%	80.95%	87.83%
NW-NER	$\langle y, d, l, k\rangle$	92%	76.19%	83.35%

Table 2: Comparison of the performance of event extraction using different NER method.

framework is presented in Table 1. The baseline re-implemented here can only output 3-tuples $\langle y, d, k\rangle$ and we simply use the gold standard event trigger words to assign to k. Still, we observe that compared to the baseline approach, the performance of our proposed framework evaluated on the 4-tuple achieves nearly 17% improvement on precision. The overall improvement on F-measure is around 7.76%.

3.5 Impact of Named Entity Recognition

We experimented with two approaches for named entity recognition (NER) in preprocessing. One is to use the NER tool trained specifically on the Twitter data (Ritter et al., 2011), denoted as "TW-NER" in Table 2. The other uses the traditional Stanford NER to extract named entities from news articles published in the same period and then perform fuzzy matching to identify named entities from tweets. The latter method is denoted as "NW-NER" in Table 2. It can be observed from Table 2 that by using NW-NER, the performance of event extraction system is improved significantly by 7.5% and 3% respectively on F-measure when evaluated on 3-tuples (without keywords) or 4-tuples (with keywords).

3.6 Impact of the Number of Events E

We need to set the number of events E in the LEM model. Figure 3 shows the performance of event extraction versus different value of E. It can be observed that the performance of the proposed framework improves with the increase of the value of E until it reaches 25, which is close to the actual number of events in our data. If further increasing E, we notice more balanced precision/recall values and a relatively stable F-measure. This shows that our LEM model is less sensitive to the number of events E so long as E is set to a relatively larger value.

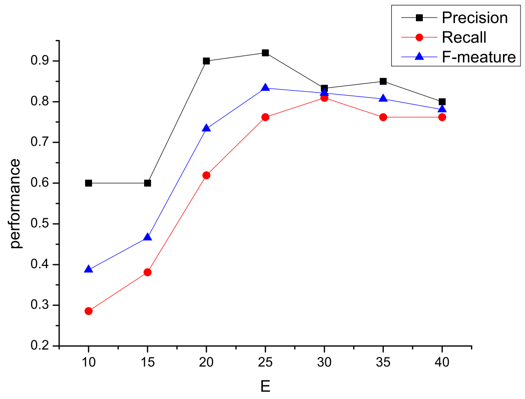


Figure 3: The performance of the proposed framework with different number of events E.

4 Conclusions and Future Work

In this paper we have proposed an unsupervised Bayesian model, called the Latent Event Model (LEM), to extract the structured representation of events from social media data. Instead of employing labeled corpora for training, the proposed model only requires the identification of named entities, locations and time expressions. After that, the model can automatically extract events which involving a named entity at certain time, location, and with event-related keywords based on the co-occurrence patterns of the event elements. Our proposed model has been evaluated on the FSD corpus. Experimental results show our proposed framework outperforms the state-of-the-art baseline by over 7% in F-measure. In future work, we plan to investigate inferring geolocations automatically from tweets. We also intend to study a better method to infer date more accurately from tweets and explore efficient ranking strategies to rank evens extracted for a better presentation of results.

Acknowledgments

This work was funded by the National Natural Science Foundation of China (61103077), Ph.D. Programs Foundation of Ministry of Education of China for Young Faculties (20100092120031), Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry, the Fundamental Research Funds for the Central Universities, and the UK's EPSRC grant EP/L010690/1.

References

Edward Benson, Aria Haghighi, and Regina Barzilay. 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*, HLT '11, pages 389–398, Stroudsburg, PA, USA. Association for Computational Linguistics.

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

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. 2011. Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of ACL*.

Thomas L. Griffiths and Mark Steyvers. 2004. Finding scientific topics. In *Proceedings of the National Academy of Sciences 101 (Suppl. 1)*, page 5228C5235.

Ralph Grishman, David Westbrook, and Adam Meyers. 2005. Nyu's english ace 2005 system description. In *ACE 05 Evaluation Workshop*.

Frederik Hogenboom, Flavius Frasincar, Uzay Kaymak, and Franciska de Jong. 2011. An overview of event extraction from text. In *Workshop on Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2011) at Tenth International Semantic Web Conference (ISWC2011)*, pages 48–57.

Xiaohua Liu, Xiangyang Zhou, Zhongyang Fu, Furu Wei, and Ming Zhou. 2012. Exacting social events for tweets using a factor graph. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, pages 1692–1698.

John Makhoul, Francis Kubala, Richard Schwartz, and Ralph Weischedel. 1999. Performance measures for information extraction. In *Proceedings of DARPA Broadcast News Workshop*.

Inderjeet Mani and George Wilson. 2000. Robust temporal processing of news. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, ACL '00, pages 69–76, Stroudsburg, PA, USA. Association for Computational Linguistics.

Sasa Petrovic, Miles Osborne, Richard McCreadie, Craig Macdonald, Iadh Ounis, and Luke Shrimpton. 2013. Can twitter replace newswire for breaking news? In *Proceedings of ICWSM'13*.

J. Piskorski, H. Tanev, and P. Oezden Wennerberg. 2007. Extracting violent events from on-line news for ontology population. In *Business Information Systems*, pages 287–300.

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

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

Alan Ritter, Mausam, Oren Etzioni, and Sam Clark. 2012. Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, pages 1104–1112, New York, NY, USA. ACM.

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