

Real-Time Detection, Tracking, and Monitoring of Automatically Discovered Events in Social Media

Miles Osborne*
Edinburgh

Sean Moran
Edinburgh

Richard McCreadie
Glasgow

Alexander Von Lunen
Loughborough

Martin Sykora
Loughborough

Elizabeth Cano
Aston

Neil Ireson
Sheffield

Craig Macdonald
Glasgow

Iadh Ounis
Glasgow

Yulan He
Aston

Tom Jackson
Loughborough

Fabio Ciravegna
Sheffield

Ann O'Brien
Loughborough

Abstract

We introduce **ReDites**, a system for real-time event detection, tracking, monitoring and visualisation. It is designed to assist Information Analysts in understanding and exploring complex events as they unfold in the world. Events are automatically detected from the Twitter stream. Then those that are categorised as being security-relevant are tracked, geolocated, summarised and visualised for the end-user. Furthermore, the system tracks changes in emotions over events, signalling possible flashpoints or abatement. We demonstrate the capabilities of **ReDites** using an extended use case from the September 2013 Westgate shooting incident. Through an evaluation of system latencies, we also show that enriched events are made available for users to explore within seconds of that event occurring.

1 Introduction and Challenges

Social Media (and especially Twitter) has become an extremely important source of real-time information about a massive number of topics, ranging from the mundane (what I had for breakfast) to the profound (the assassination of Osama Bin Laden).

Detecting events of interest, interpreting and monitoring them has clear economic, security and humanitarian importance.

The use of social media message streams for event detection poses a number of opportunities and challenges as these streams are: very high in volume, often contain duplicated, incomplete, imprecise and incorrect information, are written in informal style (i.e. short, unedited and conversational), generally concern the short-term zeitgeist; and finally relate to unbounded domains. These characteristics mean that while massive and timely information sources are available, domain-relevant information may be mentioned very infrequently. The scientific challenge is therefore the detection of the signal within that noise. This challenge is exacerbated by the typical requirement that documents must be processed in (near) real-time, such that events can be promptly acted upon.

The **ReDites** system meets these requirements and performs event detection, tracking, summarisation, categorisation and visualisation. To the best of our understanding, it is the first published, large-scale, (near) real-time Topic Detection and Tracking system that is tailored to the needs of information analysts in the security sector. Novel aspects of **ReDites** include the first large-scale treatment of spuriously discovered events and tailoring the event stream to the security domain.

*Corresponding author: miles@inf.ed.ac.uk

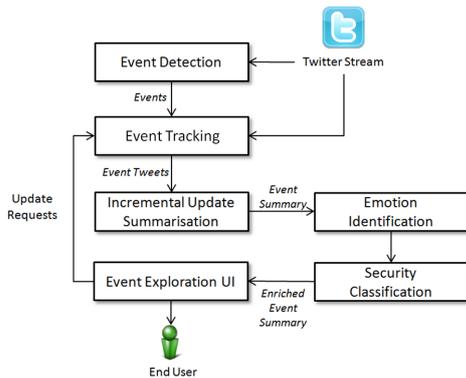


Figure 1: System Diagram

2 Related Work

A variety of event exploration systems have previously been proposed within the literature. For instance, *Trend Miner*¹ enables the plotting of term times series, drawn from Social Media (Preoțiu-Pietro and Cohn, 2013). It has a summarisation component and is also multilingual. In contrast, our system is focussed instead upon documents (Tweets) and is more strongly driven by real-time considerations. The *Social Sensor* (Aiello et al., 2013) system facilitates the tracking of pre-defined events using social streams.

In contrast, we track all automatically discovered events we find in the stream. The *Twitcident* (Abel et al., 2012) project deals with user-driven searching through Social Media with respect to crisis management. However, unlike **ReDites**, these crises are not automatically discovered. The *LRA Crisis Tracker*² has a similar purpose as **ReDites**. However, while LRA uses crowdsourcing, our **ReDites** system is fully automatic.

3 System Overview and Architecture

Figure 1 gives a high-level system description. The system itself is loosely coupled, with services from different development teams coordinating via a Thrift interface. An important aspect of this decoupled design is that it enables geographically dispersed teams to coordinate with each other. Event processing is comprised of the following main 4 steps:

1) New events are detected. An event is described by the first Tweet that we find discussing it and is defined as something that is captured within a single Tweet (Petrovic et al., 2010).

2) When an event is first discovered it may initially have little information associated with it. Furthermore, events evolve over time. Hence, the second step involves tracking the event – finding new posts relating to it as they appear and maintaining a concise updated summary of them.

3) Not all events are of interest to our intended audience, so we organise them. In particular, we determine whether an event is security-related (or otherwise), geolocate it, and detect how prominent emotions relating to that event evolve.

4) Finally, we visualise the produced stream of summarised, categorised and geolocated events for the analyst(s), enabling them to better make sense of the mass of raw information present within the original Twitter stream.

Section 6 further describes these four steps.

4 Data and Statistics

For the deployment of **ReDites**, we use the Twitter 1% streaming sample. This provides approximately four million Tweets per day, at a rate of about 50 Tweets a second. Table 1 gives some illustrative statistics on a sample of data from September 2013 to give a feel for the rate of data and generated events we produce. Table 2 gives timing information, corresponding with the major components of our system: time to process and time to transfer to the next component, which is usually a service on another machine on the internet. The latency of each step is measured in seconds over a 1000 event sample. 'Transfer' latencies is the time between one step completing and the output arriving at the next step to be processed (Thrift transfer time). Variance is the average deviation from the mean latency over the event sample.

When processing the live stream, we ingest data at an average rate of 50 Tweets per second and detect an event (having geolocated and filtered out non-English or spam Tweets) with a per-Tweet latency of 0.6 ± 0.55 seconds. Figure 2 gives latencies for the various major components of the system. All processing uses commodity machines.

5 The Westgate Shopping Mall Attack

As an illustrative example of a complex recent event, we considered a terrorist attack on the 21st of September, 2013.³ This event is used to demonstrate how our system can be used to understand it.

¹<http://www.trendminer-project.eu/>

²<http://www.lracrisistracker.com/>

³https://en.wikipedia.org/wiki/Westgate_shopping_mall_shooting

Measure	Event Detection	Tracking and Summ			Emotion Ident		Security Class	
	Detection	Transfer	Ranking	Summ	Transfer	Ident	Transfer	Class
Latency (sec.)	0.6226	0.7929	2.2892	0.0409	0.0519	0.2881	0.1032	0.1765
Variance (sec.)	0.5518	0.2987	1.3079	0.0114	0.0264	0.1593	0.0195	0.0610

Table 2: Event exploration timing and timing variance (seconds)

Data	Rate
Tweets	35 Million
Detected events	533k
Categorised (security-related) events	5795

Table 1: Data statistics, 1st September - 30th September 2013

In summary, a shopping Mall in Kenya was attacked from the 21st of September until the 24th of September. This event was covered by traditional newswire, by victims caught up in it as well as by terrorist sympathisers, often in real-time. As we later show, even though we only operate over 1% of the Twitter Stream, we are still able to find many (sub) events connected with this attack. There were 6657 mentions of Westgate in September 2013 in our 1% of sample Tweets.

6 Major Components

6.1 Event Detection

Building upon an established Topic Detection and Tracking (TDT) methodology, which assumes that each new event corresponds with seeing some novel document. the event detection component uses a hashing approach that finds novel events⁴ in constant time (Petrovic et al., 2010). To make it scale and process thousands of documents each second, it can optionally be distributed over a cluster of machines (via Storm⁵) (McCreadie et al., 2013). The system favours recall over precision and has been shown to have high recall, but a low precision (Petrovic et al., 2013). Given that we are presenting discovered events to a user and we do not want to overload them with false positives, we need to take steps to increase precision (ie present fewer false positives).

We use a *content categoriser* to determine whether a discovered event is worth reporting. Using more than 100k automatically discovered events from the Summer of 2011, we created a training set and manually labelled each Tweet:

⁴An event is defined as something happening at a given time and place. Operationally, this means something that can be described within a Tweet.

⁵<http://storm.incubator.apache.org/>

was it content bearing (what you might want to read about in traditional newswire) or irrelevant / not useful. With this labelled data, we used a Passive-Aggressive algorithm to build a content classifier. Features were simply unigrams in Tweets. This dramatically improves precision, to 70%, with a drop in recall to 25% (when tested on 73k unseen events, manually labelled by two annotators). We can change the precision-recall boundary as needed by adjusting the associated decision boundary. We do not consider non-English language Tweets in this work and they are filtered out (Lui and Baldwin, 2012).

Geolocation is important, as we are particularly interested in events that occur at a specific location. We therefore additionally geolocate any Tweets that were not originally geolocated. To geotag those Tweets that do not have any geo-location information we use the Tweet text and additional Tweet metadata (language, city/state/country name, user description etc), to learn a L_1 penalised least squares regressor (LASSO) to predict the latitude and longitude. The model is learnt on nearly 20 million geolocated Tweets collected from 2010-2014. Experiments on a held-out test dataset show we can localise Tweets to within a mean distance of 433 km of their true location. This performance is based on the prediction of *individual* tweet location and not, as in most previous work, on the location of a user who is represented by a set of tweets. Furthermore we are not restricted to a single, well-defined area (such as London) and we also evaluate over a very large set of unfiltered tweets.

Turning to the Westgate example, the first mention of it in our data was at 10:02 UTC. There were 57 mentions of Westgate in discovered events, of which 42 mentioned Kenya and 44 mentioned Nairobi. The first mention itself in Twitter was at 09:38 UTC. We declared it an event (having seen enough evidence and post-processing it) less than one second later:

Westgate under siege. Armed thugs. Gunshots reported. Called the managers, phones are off/busy. Are cops on the way?

We also captured numerous informative sub-

events covering different aspects and sides of the central Westgate siege event, four of these are illustrated below:

Post Time	Tweet
10:05am	RT @ItsMainaKageni: My friend Ruhila Adatia passed away together with her unborn child. Please keep her family and new husband in your thou
10:13am	RT howden_africa: Kenya police firing tear gas and warning shots at Kenyan onlookers. Crowd getting angry #westgate
10:10am	RT @BreakingNews: Live video: Local news coverage of explosions, gunfire as smoke billows from Nairobi, Kenya, mall - @KTNKenya
10:10am	"Purportedly official Twitter account for al-Shabaab Tweeting on the Kenyan massacre HSM_Press (http://t.co/Xn Cz9 BulGj)

6.2 Tracking and Summarisation

The second component of the event exploration system is *Tracking and Summarisation (TaS)*. The aim of this component is to use the underlying Tweet stream to produce an overview for each event produced by the event detection stage, updating this overview as the event evolves. Tracking events is important when dealing with live, ongoing disasters, since new information can rapidly emerge over time.

TaS takes as input a Tweet representing an event and emits a list of Tweets summarising that event in more detail. TaS is comprised of two distinct sub-components, namely: real-time tracking; and event summarisation. The real-time tracking component maintains a sliding window of Tweets from the underlying Tweet stream. As an event arrives, the most informative terms contained⁶ form a search query that is used to retrieve new Tweets about the event. For example, taking the Tweet about the Westgate terrorist attack used in the previous section as input on September 21st 2013 at 10:15am, the real-time tracking sub-component retrieved the following related Tweets from the Twitter Spritzer (1%) stream⁷ (only 5/100 are shown):

ID	Post Time	Tweet	Score
1	10:05am	Westgate under siege. Armed thugs. Gunshots reported. Called the managers, phones are off/busy. Are cops on the way?	123.7
2	10:13am	DO NOT go to Westgate Mall. Gunshots and mayhem, keep away until further notice.	22.9
3	10:13am	RT DO NOT go to Westgate Mall. Gunshots and mayhem, keep away until further notice.	22.9
4	10:10am	Good people please avoid Westgate Mall. @PoliceKE @IGkimaiyo please act ASAP, reports of armed attack at #WestgateMall	22.2
5	10:07am	RT @steve_enzo: @kirimiachieng these thugs won't let us be	11.5

⁶Nouns, adjectives, verbs and cardinal numbers

⁷<https://dev.twitter.com/docs/streaming-apis/streams/public>

The second TaS sub-component is event summarisation. This sub-component takes as input the Tweet ranking produced above and performs extractive summarisation (Nenkova and McKeown, 2012) upon it, i.e. it selects a subset of the ranked Tweets to form a summary of the event. The goals of event summarisation are two-fold. First, to remove any Tweets from the above ranking that are not relevant to the event (e.g. Tweet 5 in the example above). Indeed when an event is first detected, there may be few relevant Tweets yet posted. The second goal is to remove redundancy from within the selected Tweets, such as Tweets 2 and 3 in the above example, thereby focussing the produced summary on novelty. To tackle the first of these goals, we leverage the score distribution of Tweets within the ranking to identify those Tweets that are likely background noise. When an event is first detected, few relevant Tweets will be retrieved, hence the mean score over the Tweets is indicative of non-relevant Tweets. Tweets within the ranking whose scores diverge from the mean score in the positive direction are likely to be on-topic. We therefore, make an include/exclude decision for each Tweet t in the ranking R :

$$include(t, R) = \begin{cases} 1 & \text{if } score(t) - SD(R) > 0 \\ & \text{and } |SD(R) - score(t)| > \\ & \theta \cdot \frac{1}{|R|} \sum_{t' \in R} |SD(R) - score(t')| \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $SD(R)$ is the standard deviation of scores in R , $score(t)$ is the retrieval score for Tweet t and θ is a threshold parameter that describes the magnitude of the divergence from the mean score that a Tweet must have before it is included within the summary. Then, to tackle the issue of redundancy, we select Tweets in a greedy time-ordered manner (earliest first). A similarity (cosine) threshold between the current Tweet and each Tweet previously selected is used to remove those that are textually similar, resulting in the following extractive summary:

ID	Post Time	Tweet	Score
1	10:05am	Westgate under siege. Armed thugs. Gunshots reported. Called the managers, phones are off/busy. Are cops on the way?	123.7
2	10:13am	DO NOT go to Westgate Mall. Gunshots and mayhem, keep away until further notice.	22.9
4	10:10am	Good people please avoid Westgate Mall. @PoliceKE @IGkimaiyo please act ASAP, reports of armed attack at #WestgateMall	22.2

Finally, the TaS component can be used to track

events over time. In this case, instead of taking a new event as input from the event detection component, a previously summarised event can be used as a surrogate. For instance, a user might identify an event that they want to track. The real-time search sub-component retrieves new Tweets about the event posted since that event was last summarised. The event summarisation sub-component then removes non-relevant and redundant Tweets with respect to those contained within the previous summary, producing a new updated summary.

6.3 Organising Discovered Events

The events we discover are not targeted at information analysts. For example, they contain sports updates, business acquisitions as well as those that are genuinely relevant and can bear various opinions and degrees of emotional expression. We therefore take steps to filter and organise them for our intended audience: we predict whether they have a specific security-focus and finally predict an emotional label for events (which can be useful when judging changing viewpoints on events and highlighting extreme emotions that could possibly motivate further incidents).

6.3.1 Security-Related Event Detection

We are particularly interested in security-related events such as violent events, natural disasters, or emergency situations. Given a lack of in-domain labelled data, we resort to a weakly supervised Bayesian modelling approach based on the previously proposed Violence Detection Model (VDM) (Cano et al., 2013) for identifying security events.

In order to differentiate between security and non-security related events, we extract words relating to security events from existing knowledge sources such as DBpedia and incorporate them as priors into the VDM model learning. It should be noted that such a word lexicon only provides initial prior knowledge into the model. The model will automatically discover new words describing security-related events.

We trained the VDM model on a randomly sampled 10,581 Tweets from the TREC Microblog 2011 corpus (McCreadie et al., 2012) and tested the model on 1,759 manually labelled Tweets which consist of roughly the same number of security-related and non-security related Tweets. Our results show that the VDM model achieved 85.8% in F-measure for the identification

of security-related Tweets, which is not far from the F-measure of 90% obtained using the supervised Naive Bayes classifier despite using no labelled data in the model.

Here, we derived word priors from a total of 32,174 documents from DBpedia and extracted 1,612 security-related words and 1,345 non-security-related words based on the measurement of relative word entropy. We then trained the VDM model by setting the topic number to 50 and using 7,613 event Tweets extracted from the Tweets collected during July-August 2011 and September 2013 in addition to 10,581 Tweets from the TREC Microblog 2011 corpus. In the aforementioned Westgate example, we classify 24% of Tweets as security-related out of a total of 7,772 summary Tweets extracted by the TaS component. Some of the security-related Tweets are listed below⁸:

ID	Post Time	Tweet
1	9:46am	Like Bin Laden kind of explosion? @The_realBIGmeat: There is an explosion at westgate!"
2	10:08am	RT @SmritiVidarthi: DO NOT go to Westgate Mall. Gunshots and mayhem, keep away till further notice.
3	10:10am	RT @juliegichuru: Good people please avoid Westgate. @PoliceKE @IGkimaiyo please act ASAP, reports of armed attack at #WestgateMall.
4	10:13am	there has bn shooting @ Westgate which is suspected to b of gangs.....there is tension rt nw....

6.3.2 Emotion

Security-related events can be fraught, with emotionally charged posts possibly evolving over time, reflecting ongoing changes in underlying events. Eight basic emotions, as identified in the psychology literature (see (Sykora et al., 2013a) for a detailed review of this literature) are covered, specifically; anger, confusion, disgust, fear, happiness, sadness, shame and surprise. Extreme values –as well as their evolution– can be useful to an analyst (Sykora et al., 2013b). We detect emotions in Tweets and support faceted browsing. The emotion component assigns labels to Tweets representing these emotions. It is based upon a manually constructed ontology, which captures the relationships between these emotions and terms (Sykora et al., 2013a).

We sampled the summarised Tweets of the Westgate attack, starting from the event detection and following the messages over a course of seven days. In the relevant Tweets, we detected that

⁸Note some Tweets happen on following days.

8.6% had emotive terms in them, which is in line with the aforementioned literature. Some example expressions of emotion include:

Time	Tweet	Emotions
03:34	-) Ya so were those gunshots outside of gables?! I'm terrified ?	Fear
06:27	-) I'm so impressed @ d way. Kenyans r handling d siege.	Surprise
14:32	-) All you xenophobic idiots spewing anti-Muslim bullshit need to -get in one of these donation lines and see how wrong you ?	Fear Disgust

For Westgate, the emotions of sadness, fear and surprise dominated. Very early on the emotions of fear and sadness were expressed, as Twitter users were terrified by the event and saddened by the loss of lives. Sadness and fear were – over time – the emotions that were stated most frequently and constantly, with expressions of surprise, as users were shocked about what was going on, and some happiness relating to when people managed to escape or were rescued from the mall. Generally speaking, factual statements in the Tweets were more prominent than emotive ones. This coincides with the emotive Tweets that represented fear and surprise in the beginning, as it was not clear what had happened and Twitter users were upset and tried to get factual information about the event.

6.4 Visual Analytics

The visualisation component is designed to facilitate the understanding and exploration of detected events. It offers faceted browsing and multiple visualisation tools to allow an information analyst to gain a rapid understanding of ongoing events. An analyst can constrain the detected events using information both from the original Tweets (e.g. hashtags, locations, user details) and from the updated summaries derived by **ReDites**. The analyst can also view events using facet values, locations/keywords in topic maps and time/keywords in multiple timelines. By combining information dimensions, the analyst can determine patterns across dimensions to determine if an event should be acted upon – e.g the analyst can choose to view Tweets, which summarise highly emotive events, concerning middle eastern countries.

7 Discussion

We have presented **ReDites**, the first published system that carries out large-scale event detection, tracking summarisation and visualisation for the security sector. Events are automatically identified and those that are relevant to information analysts

are quickly made available for ongoing monitoring. We showed how the system could be used to help understand a complex, large-scale security event. Although our system is initially specialised to the security sector, it is easy to repurpose it to other domains, such as natural disasters or smart cities. Key aspects of our approach include scalability and a rapid response to incoming data.

Acknowledgements

This work was funded by EPSRC grant EP/L010690/1. MO also acknowledges support from grant ERC Advanced Fellowship 249520 GRAMPLUS.

References

- F. Abel, C. Hauff, G.-J. Houben, R. Stronkman, and K. T. Semantics + filtering + search = twitcident. exploring information in social web streams. In *Proc. of HT*, 2012.
- L. M. Aiello et al. L. Aiello, G. Petkos, C. Martin, D. Corney, S. Papadopoulos, R. Skraba, A. Goker, Y. Kompatsiaris, A. Jaimes Sensing trending topics in Twitter. *Transactions on Multimedia Journal*, 2012.
- A.E. Cano, Y. He, K. Liu, and J. Zhao. A weakly supervised bayesian model for violence detection in social media. In *Proc. of IJCNLP*, 2013.
- M. Lui and T. Baldwin. Langid.py: An off-the-shelf language identification tool. In *Proc. of ACL*, 2012.
- R. McCreadie, C. Macdonald, I. Ounis, M. Osborne, and S. Petrovic. Scalable distributed event detection for twitter. In *Proc. of Big Data*, 2013.
- R. McCreadie, I. Soboroff, J. Lin, C. Macdonald, I. Ounis and D. McCullough. On building a reusable Twitter corpus. In *Proc. of SIGIR*, 2012.
- A. Nenkova and K. McKeown. A survey of text summarization techniques. In *Mining Text Data Journal*, 2012.
- S. Petrovic, M. Osborne, and V. Lavrenko. Streaming first story detection with application to Twitter. In *Proc. of NAACL*, 2010.
- S. Petrovic, M. Osborne, R. McCreadie, C. Macdonald, I. Ounis, and L. Shrimpton. Can Twitter replace newswire for breaking news? In *Proc. of WSM*, 2012.
- D. Preoțiuc-Pietro and T. Cohn. A temporal model of text periodicities using gaussian processes. In *Proc. of EMNLP*, 2012.
- M. D. Sykora, T. W. Jackson, A. O'Brien, and S. Elayan. Emotive ontology: Extracting fine-grained emotions from terse, informal messages. *Computer Science and Information Systems Journal*, 2013.
- M. D. Sykora, T. W. Jackson, A. O'Brien, and S. Elayan. National security and social media monitoring. In *Proc. of EISIC*, 2013.