Event Identification and Tracking in Social Media Streaming Data

Andreas Weiler, Michael Grossniklaus, and Marc H. Scholl
University of Konstanz
Dept. of Computer & Information Science
Box D 188, 78457 Konstanz, Germany
firstname.lastname@uni-konstanz.de

ABSTRACT

In recent years, the growing popularity and active use of social media services on the web have resulted in massive amounts of user-generated data. With these data available, there is also an increasing interest in analyzing it and to extract information from it. Since social media analysis is concerned with investigating current events around the world, there is a strong emphasis on identifying these evens as quickly as possible, ideally in real-time. In order to scale with the rapidly increasing volume of social media data, we propose to explore very simple event identification mechanisms, rather than applying the more complex approaches that have been proposed in the literature. In this paper, we present a first investigation along this motivation. We discuss a simple sliding window model, which uses shifts in the inverse document frequency (IDF) to capture trending terms as well as to track the evolution and the context around events. Further, we present an initial experimental evaluation of the results that we obtained by analyzing real-world data streams from Twitter.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: Information Storage and Retrieval; H.4 [**Information Systems Applications**]: Miscellaneous

Keywords

event detection, stream processing, social media analytics

1. INTRODUCTION AND MOTIVATION

The continuous emergence of new web services, such as social media platforms and technologies for generating and receiving streamed data, imposes new challenges on the way such data volumes are processed and analyzed in real-time or near real-time. Since the users of information services are typically interested in current events and actual happenings of the world, it is necessary to retain the real-time characteristic of the streams and to perform the identification of real-world events as fast as possible.

In this paper, we focus on the use-case of Twitter, the most popular social microblogging site, which produces a large volume of data as a continuous stream of messages, so-called "tweets". Since its inception, the way people use Twitter has undergone a remarkable evolution. While Twitter was initially intended as a service to share short personal status messages, it quickly became a platform that people used to report on and stay informed about current events happening all around the world. This change in usage was also reflected in Twitter's user interface by changing the original prompt "*What are you doing?*" to the more general question "*What's happening?*"[1].

Another important characteristic of Twitter is its vibrant community with, as of 2013, 200 million daily active users from all around the world. Due to its lightweight approach to broadcasting, important news rapidly propagate through Twitter's densely interconnected social network. Although the resulting volume and variety of content in the information flow is a great opportunity for data analysis, it also gives rise to the challenge of detecting significant messages that can be used to identify events in the frenzy of tweets. As a consequence, several state of the art approaches exist that address the problem of accurate event detection. In the last six years, tweets have gone from 16 millions per year to 400 millions per day. Taking this rapid growth into consideration, we believe that the scalability of event identification is as important as its accuracy. Therefore, we propose to study simple approaches that, ideally, can be tuned to trade-off precision for lower computational complexity.

In this paper, we present a first exploration into this direction. We propose a simple event identification approach, which uses a sliding window model to extract events and the context of events in real-time from the live public data stream of Twitter. Our approach is based on monitoring shifts in the inverse document frequency (IDF) of terms and therefore suggests that it is possible to handle large amounts of data and get important insights by means of aggregation only. Since our approach is based on windowing, the window size is a natural parameter that can be used to control the precision/complexity trade-off. Apart from the approach itself, we also present a first evaluation based on a case study to obtain an indication of the results that can be expected from such an approach.

[1] `http://blog.twitter.com/2009/11/whats-happening.html`

2. EVENT ANALYSIS

Event detection is a classical problem in computer science and has been studied for many years in various research areas. A lot of research is dedicated to detecting anomalies or novelties in different data sources ([5, 11]), considering those phenomena to be an indication of an event. Further related research deals with the detection of changes or drifts in data streams ([1, 7, 9]).

Taking into account a vast number of Twitter messages generated each second, it becomes a crucial task to group messages by topics or events. Because there is no explicit knowledge about current or future events, the latter have to be identified and detected in an on-line fashion without limitation to any domain via predefined keywords. We propose that only by means of aggregation it is possible to handle large amount of data and gain important insights into it. Therefore, the detected high-level representations can be used to compress the tweets in a meaningful manner.

Our approach of using sliding windows to identify events in streaming data considers the timestamp information included within the tweets as a basis for the window sizes. In the following, we describe the identification of events by using textual analysis of the content of tweets and statistical analysis of the frequency of terms. Once an event has been detected, we keep track of the most co-occurring terms with the event term, which mainly describe the context around an event. This can be helpful to provide a better overview and more insights into the event's evolution to analysts and other users.

2.1 Event Identification

The content of the tweet messages provides a high variety of information and as such can be considered as the most important dimension in the data set. In the following, we describe the process of event identification by analyzing the content of tweets.

The first step of the event term extraction process is the tokenization and part-of-speech tagging of the tokens by using a especially for Twitter tailored tokenizer and Part-Of-Speech Tagger [8]. Since pure tokenization of texts results in an abundance of terms, but the main subject of an event is typically reflected in nouns we filter the resulting set of tokens to nouns and proper nouns for further treatment. Additionally to single nouns, we also take bigrams (nouns with preceding adjectives, verbs, or nouns) into account. We further reduce the token set by discarding all tokens, which are shorter than four characters, as well as tokens contained in a standard English stopword list, or containing any non-alphabetic characters. Because the spelling of words in social media can be very diverse and the amount of terms would increase a lot, we also discard tokens with more than three successive repetitions of the same character (e.g., "hellooooo", "gooooaaalll"). Once the preprocessing is done, each tweet message is represented by a set of terms $T = (t_1, t_2, t_3, ...)$. To avoid wrong identification of event candidates by continuous repetition of the same term, these sets are kept duplicate-free.

The evolution of relevant terms is evaluated by an ongoing process using the following rules. The incoming stream $TWS = (tw_1, tw_2, tw_3, ...)$ is partitioned into fixed sized windows $(w_1, w_2, w_3, ...)$. For each extracted term (we call them event candidates ec) in the stream we continuously calculate an *IDF* [16] ($idf(ec)$) value for each of the windows. In addition, we calculate the percentage of the shift of the IDF value from one window to another ($sidf(ec)$), which is only possible if the ec occurs in two successive windows. For further evaluation, we also calculate the average IDF value ($avg(idf(ec))$) of all terms in the window and the average value of all shifts $avg(sidf(ec))$ (possible for all ec occurring in the last two successive windows) between two succeeding windows.

If the ec occurs in more then n successive windows, we check the IDF value of all n windows against the average value $avg(idf(ec))$ of the corresponding window. If all values are lower then the average value the ec is further evaluated. After the first check, we check the IDF value shift $sidf(ec)$ for the windows (w_{n-3}, w_{n-2}), (w_{n-2}, w_{n-1}) and (w_{n-1}, w_n) against the corresponding average values. If all shift values are higher than the corresponding average value and the added up shift value is over a certain threshold, we identify this ec as an event term. In this way, both fast and slower increasing event terms can be identified, and the identification of event terms adapts to dynamically changing boundaries, which are the currently existing average values.

Since the amount of event candidates in the term set increases continuously we discard all terms, which are missing in a window. Figure 3 shows that there is an almost equivalent number of terms in the windows over time. Event candidates, which are identified as event terms are passed on to the event tracking phase, which is described in the next section.

2.2 Event Tracking

Once the event identification phase has identified an event candidate as event term, it is also interesting for an analyst to keep track of the event, to get an ongoing overview and insight of the happenings related to the event, or to evaluate the importance of an event. Therefore, the event tracking phase of our analysis is initiated after an event term is identified.

To support this process, we extract all co-occurring terms of the event term, which includes verbs, nouns, and adjectives and use the term cleaning methods, which are mentioned before. Afterwards we calculate the percentage of the co-occurrences of the term with the event term in the corresponding window and order them by the percentage value. To summarize the context around a identified event term the top n co-occurrence terms are extracted continuously.

3. EVALUATION

In this section, we describe the evaluation of our approach in terms of the experimental setup and the experiments that we conducted. In both experiments, we applied an implementation of our approach to the identification of events in the Twitter social data stream and subsequent tracking of the evolution of these identified events.

3.1 Setup

The Twitter platform provides direct access to the public live stream of Twitter messages via a set of developer APIs. The Twitter API [17] enables application developers to receive a large portion of the total number of daily produced tweets. By using the Twitter Streaming API with the so-called "gardenhose" access level, we are able to collect 10% of the total public live stream. Additionally, we merge our data set with a geo-filtered stream to increase the number

of geo-tagged tweets. We can assume that about 10% of the incoming tweets have geographic information available. This information is set either automatically by the mobile device, or manually by the author of the tweet, or by both of them. Figure 1 displays statistics about the amount of incoming tweets for a representative sample of days. From these numbers we can conclude that we can receive an average of over one million tweets per hour with the average of 20,000 tweets per minute. Further we can see that there is a certain decrease in the number of tweets between the hours 11 and 16 each day.

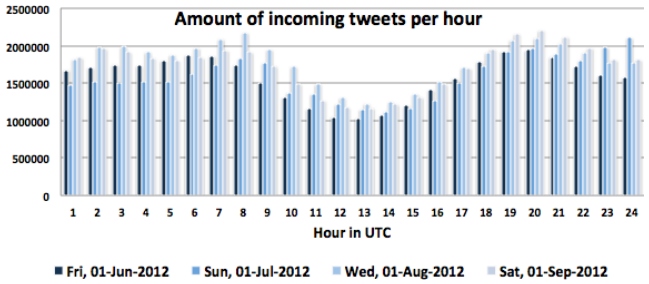

Figure 1: Number of incoming tweets per hour for the first day of the months June, July, August, and September 2012.

In our implementation, we rely on the native XML database system *BaseX*[2] for data management and processing. All designated incoming data is stored in a standardized format to support fast and easy data access. The data in the Twitter streams is in the JavaScript Object Notation (*JSON*) format, which is efficiently converted to XML on-the-fly using the JSON conversion functionality of BaseX. Since this solution converts the JSON object directly to XML, there is an automatic adaptation to all potential changes in the format of the streamed Twitter data.

For this work, we simulate a live-stream behavior of Twitter by pushing stored tweets from the database in the same sequence, in which they were gathered from the stream. The client can register a filter query—formulated in *XQuery*—on the stream to receive only a requested type of tweets from the database system. For example, it is possible to filter for tweets with valid geographic information or for tweets containing a specified keyword. By simulating a live streaming environment we ensure that the analysis can also be directly applied to the on-line stream by connecting to the live Twitter streaming data running through the database instead of connecting to the simulated stream.

3.2 Experiments

For our experiments, we simulate a live stream of real-life Twitter data and analyze the streamed tweets. Since we only need the text of the tweets, we run a continuous filter query for the text field on the incoming stream and discard all other unused data fields. This helps us to minimize the amount of data to process during the analysis. The window size for this evaluation is set to one minute, which allows us to identify events within four minutes after the first triggering appearance of the corresponding event term. The minimum limit of the added up IDF shift value for a term to become an event term is set to 20% for single terms and 12% for bigrams. These values can be easily changed and dynamically adjusted. To track the evolution of identified events, we also take four minute windows to extract the top 10 most co-occurring terms of the event term. By manually evaluating the event terms and the most frequent co-occurrence terms of the event terms we can derive that the event terms are a mixture of non-english terms which are not filtered out by our analysis, names of famous people (e.g., *Justin Bieber*, *Chris Brown*), and real-world events.

The first experiment deals with the hours from *07:00 - 09:00 AM UTC on Wednesday, April 11th 2012*. In this time frame our analysis identifies a total of 50 event terms for single terms and 21 event terms for bigrams within a total of 916,948 tweets. To further explain the usefulness of our approach, we choose the following two event terms. In Table 1, we can see how the event term "earthquake" in minute 8:45 with an overall shift of 22.13% evolved to an event term. Table 2 shows the evolution of the event term "tsunami" five minutes later in minute 8:50 with an overall shift of 27.67%. The evolution of the IDF value of the event terms in difference to the evolution of the IDF value of non-event terms can be seen in Figure 2. We can see that the IDF value of the event term "earthquake" increases significantly and therefore there is a high shift in the value. Furthermore, the event term "tsunami" shows almost the same behavior just five minutes later. In contrast to these terms, the two non-event terms "twitter" and "love" have almost no change in the IDF value over time.

	earthquake	**average**
IDF Value Minute 8:42	6.54	7.69
IDF Value Minute 8:43	6.70	8.09
IDF Value Minute 8:44	5.85	7.93
IDF Value Minute 8:45	5.15	7.85
IDF Shift Minute 8:42-8:43	-2.50%	-3.86%
IDF Shift Minute 8:43-8:44	12.72%	1.45%
IDF Shift Minute 8:44-8:45	11.91%	0.39%
Total Shift	22.13%	

Table 1: Detection of event term "earthquake" in minute 8:45.

	tsunami	**average**
IDF Value Minute 8:47	5.24	8.18
IDF Value Minute 8:48	4.76	8.29
IDF Value Minute 8:49	4.35	8.43
IDF Value Minute 8:50	3.92	8.39
IDF Shift Minute 8:47-8:48	9.24%	-0.99%
IDF Shift Minute 8:48-8:49	8.64%	-1.22%
IDF Shift Minute 8:49-8:50	9.79%	0.19%
Total Shift	27.67%	

Table 2: Detection of event term "tsunami" in minute 8:50.

The result of our event identification analysis indicates that an "earthquake" and a "tsunami" happened in the corresponding time frame. With the event tracking analysis we are able to extract more useful information about the identified events. After the event is detected, we analyze the new incoming tweets corresponding to that event and extract the ten most frequent co-occurrence terms for the time windows of four minutes. The ten most frequent common terms and their percentage frequency for the minutes after the event "earthquake" are the following:

[2]`http://www.basex.org`

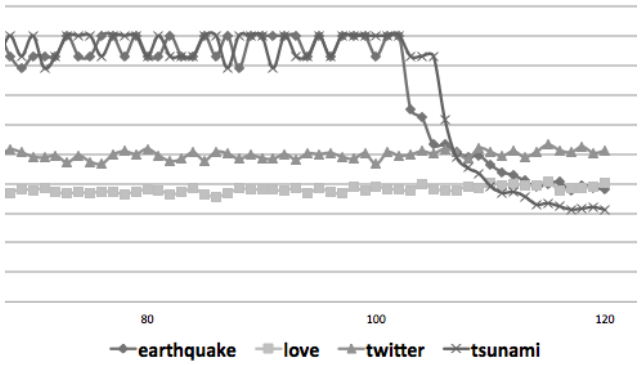

Figure 2: Sample IDF value evolution of the event terms "earthquake" and "tsunami" and the non-event terms "love" and "twitter" in the first experiment.

- *Minute 8:46 to Minute 8:49:* epicenter (7.39), aceh (6.82), banda (5.68), tsunami (5.11), chennai (5.11), office (4.55), depth (4.55), warning (4.55), malaysia (3.98), northern (3.41)
- *Minute 8:50 to Minute 8:54:* aceh (37.02), tsunami (25.39), warning (24.22), agency (23.26), issues (23.06), epicenter (17.83), sumatra (17.44), indonesia (17.25), coast (15.50), west (15.31)
- *Minute 8:55 to Minute 8:59:* aceh (42.13), tsunami (33.43), warning (30.62), agency (24.72), issues (24.72), indonesia (21.35), scale (17.42), magnitude (17.13), sumatra (16.85), richter (16.85)

We can see that in the first window the term "tsunami" is in only 5% of the tweets with "earthquake". In the second window, however, there is a rapidly growing frequency of the term "tsunami", which allows us to conclude that the topic drifts from discussions and news about the earthquake to messages about an expected or ongoing tsunami. Also further information like "aceh" (a city in Indonesia), "indonesia", and "sumatra" is extracted by the event tracking phase. The extraction of the ten most frequent common terms identifies also terms like "aceh" or "warning". With this information we are able to combine the two detected events ("earthquake" and "tsunami") into one top event. The terms for the minutes after the "tsunami" event happened can be seen in the following:

- *Minute 8:51 to Minute 8:55:* gempa (64.96), peringatan (50.76), bengkulu (50.67), dini (50.49), lampung (50.49), sumut (50.13), sumbar (49.96), aceh (25.88), warning (13.84), earthquake (12.76)
- *Minute 8:56 to Minute 9:00:* gempa (52.53), peringatan (40.08), dini (39.45), bengkulu (39.03), lampung (38.71), sumbar (38.71), sumut (38.50), aceh (29.11), warning (18.78), earthquake (13.71)

The second experiment deals with the hours from *05:00 - 09:00 AM UTC on Friday, July 20th 2012*. In this time frame our analysis identifies a total of 73 event terms for single terms and a total of 54 event terms for bigrams. Figure 3 shows the frequencies of the tweets and terms per minute, the overall total of all tweets for the four hours is 4,602,574 tweets. The frequency overview shows that we have an almost constantly number of terms and tweets in the windows over time. In Table 3, we can see how the event term "knight rises" in minute 7:20 with an overall shift of 14.21% evolved to an event term. Table 4 shows the evolution of the event term "aurora" almost an hour later in minute 8:19 with an overall shift of 20.64%.

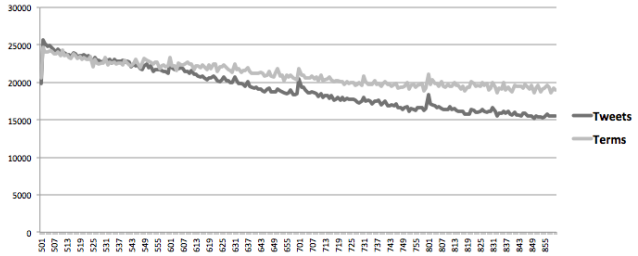

Figure 3: Frequency of the amount of tweets and terms per minute in the four hours of the second experiment.

	knight rises	average
IDF Value Minute 5:51	8.61	9.13
IDF Value Minute 5:52	7.81	9.13
IDF Value Minute 5:53	7.78	9.10
IDF Value Minute 5:54	7.59	9.11
IDF Shift Minute 5:51-5:52	9.32%	-0.19%
IDF Shift Minute 5:52-5:53	0.42%	0.10%
IDF Shift Minute 5:53-5:54	2.46%	-0.23%
Total Shift	12.20%	

Table 3: Detection of event term "knight rises" in minute 5:54.

	aurora	average
IDF Value Minute 8:16	8.59	8.84
IDF Value Minute 8:17	7.29	8.84
IDF Value Minute 8:18	6.96	8.82
IDF Value Minute 8:19	6.89	8.82
IDF Shift Minute 8:16-8:17	15.15%	0.00%
IDF Shift Minute 8:17-8:18	4.53%	0.02%
IDF Shift Minute 8:18-8:19	0.96%	-0.11%
Total Shift	20.64%	

Table 4: Detection of event term "aurora" in minute 8:19.

The result of our event identification analysis shows that a couple of events happened in the corresponding hour. In our case we are interested in two events. The first one is called "knight rises" and the second is "aurora". Since we have no knowledge as to what these events are about, we use the results of the event tracking analysis to extract more useful information about the events. The ten most frequent co-occurrence terms and their percentage frequency for the minutes after the event "knight rises" are the following:

- *Minute 5:56 to Minute 6:00:* dark (96.43), movie (10.71), experience (7.14), century (7.14), line (7.14), cinemas (7.14), center (7.14), imax (7.14), theatre (7.14), river (3.57)

- *Minute 6:00 to Minute 6:04:* dark (100.00), cinemark (8.11), century (8.11), sinners (5.41), children (5.41), masked (5.41), imax (5.41), theaters (5.41), mamba (5.41), people (5.41)
- *Minute 6:04 to Minute 6:08:* dark (100.00), movie (9.52), people (7.14), batman (7.14), midnight (7.14), spiderman (4.76), regal (4.76), rumor (4.76), alert (4.76), spoiler (4.76)

By looking at the co-occurrence terms, we are able to figure out that most of the tweets discuss the "premiere" of a movie in an IMAX theatre at midnight. Since there is an ongoing premiere of the new movie "The Dark Knight Rises" on this day and time, we can conclude that the identified event is about the corresponding real-world event.

Since the analysis also detected an event "aurora" in minute 8:19, there could be a correlation between these two events. The extraction of the ten most frequent co-occurring terms for the event term "aurora" identifies the following terms:

- *Minute 8:20 to Minute 8:24:* shooting (78.69), colorado (75.41), film (67.21), premiere (67.21), people (65.57), dark (63.93), knight (63.93), local (62.30), media (60.66), rises (60.66), update (59.02)
- *Minute 8:24 to Minute 8:28:* colorado (70.45), shooting (61.36), dark (52.27), knight (52.27), film (45.45), premiere (45.45), people (43.18), rises (43.18), media (36.36), local (36.36), update (36.36)

We can derive that the newly detected event is somehow related to the earlier detected event "knight rises". Since the first event describes the premiere of a new movie and the new event "aurora" describes a mass shooting happening during the movie premiere of "The Dark Knight Rises" in "Colorado", we can conclude that there is a dependency between these two events.

4. RELATED WORK

The extreme popularity of Twitter and access to its public social data stream have resulted in an increasing amount of Twitter-related scientific, industrial, and governmental research initiatives. In this section, we summarize the most related work.

Bontcheva *et al.* [4] present an overview of sense making in social media data, which also includes current event detection methods in social media streams. They classified detection methods into three categories: clustering-based, model-based, and those based on signal processing.

An event detection system dedicated to earthquakes is presented by Sakaki *et al.* [14]. In contrast to our approach, they use the keyword search feature provided by the Twitter API to gather data in specified time intervals. Schühmacher *et al.* [15] propose another domain-specific event detection method on microblogs to support forensic analysis. They train a linear classifier to detect suspicious posts. Weng *et al.* [18] use wavelet analysis on frequency-based raw signals of terms from tweets for detecting events. They use a keyword-filtered dataset to show their practical usage for identifying events during the Singapore General Election in 2011. Marcus *et al.* [10] demonstrate an application called "TwitInfo", which identifies and labels event peaks for given search queries related to the event. In contrast to our proposed idea, which uses an unfiltered data stream, all of the above mentioned systems are somehow restricted.

More recently, Ritter *et al.* [13] presented the first approach for open domain event extraction from Twitter. Their approach is based on latent variable models and proceeds by first discovering event types, which match the data and then using these results to classify aggregate events. However, no discussion about applying this approach directly to the streaming data is included. Alvanaki *et al.* [3] proposed a system "enBlogue", which analyzes statistics about tags and tag pairs for identifying unusual shifts in correlations. Further recent work proposed by Nishida *et al.* [12] shows a classification model of tweet streams for identifying changes in statistical properties on word basis, which is used for topic classification.

General research on on-line event detection has a long track record. In 1998, Yang *et al.* [19] published a study about retrospective and on-line event detection. They used text retrieval and clustering techniques for detecting events in a temporally ordered stream of news stories. In the same year, Allan *et al.* [2] focused on a strict on-line setting by using a modified single-pass clustering approach for event detection and information filtering for event tracking. However these two approaches used clean and well-formed news stories as sources for detecting events.

5. CONCLUSIONS

In this paper, we presented a method for identifying events in the real-world social media data streams of Twitter. We have shown that by means of aggregation it is possible to handle large volumes of data and gain important insights into it. We believe that under ideal conditions the data streamed by Twitter can support faster detection of events than by using reports of news agencies. Although we obtained the data through the Twitter API that only provides 10% of the total data stream, which might introduce a skew in the tweets we analyze, the total stream can be assumed to contain more complete information an event.

Our evaluation shows that we are able to identify events as well as to track the progress of the event and the context around it in a simulated environment. However, the identification also detects a certain amount of non-event terms as events. This is an indication that the identification phase needs to be improved by including more information, like geographical data or other features extracted from the metadata. The tracking of the events shows that the context around an event can be described properly and it is also possible to identify relationships and dependencies between events. For example, in both experiments we were able to draw the conclusion that the second identified event is a follow-up or related event of the first one. With the continuous removal of event candidates from the term set, we are able to scale to the amount and the speed of tweets and terms in the streaming data.

6. FUTURE WORK

A first extension of our approach will be the integration of further information into the event identification phase. This goal can be achieved by using information from the metadata fields or by extracting more information from the textual content of the tweets. In addition to the actual content of the tweet messages, Twitter provides 60 metadata fields describing the tweet (e.g., count of retweets, geographic location) and the user's profile (e.g., count of followers). This

additional knowledge can be used to extract further characteristics of the identified events. For example, if a majority of tweets related to an event have similar geographical information (such as the same city or country), one can assume that the event possibly originated at that location. Furthermore, it will be an interesting task to implement a categorization and ranking (e.g., globally important) analysis for the detected events. To support the ranking of events, we can also integrate the metadata (e.g., number of retweets vs. number of independent tweets) in our analysis.

A further extension is the integration of additional data sources. Stock exchange markets, weather forecasts, data from news agencies, RSS feeds, and further social media services offer contents that can be retrieved in different ways as streams and could also enrich our event identification and tracking analysis. For example, even the social media photo sharing platform Flickr was recently used as data source for event detection [6].

For evaluation purposes it would be interesting to evaluate our approach against more complex state of the art approaches, such as the one presented by Weng *et al.* [18]. This line of future work would enable us to better understand how much complexity is needed to differentiate between event terms and standard terms.

7. REFERENCES

[1] I. Adä and M. R. Berthold. Unifying Change – Towards a Framework for Detecting the Unexpected. In *Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops*, ICDMW '11, pages 555–559, Washington, DC, USA, 2011. IEEE Computer Society.

[2] J. Allan, R. Papka, and V. Lavrenko. On-line new event detection and tracking. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '98, pages 37–45. ACM, 1998.

[3] F. Alvanaki, S. Michel, K. Ramamritham, and G. Weikum. See what's enblogue: real-time emergent topic identification in social media. In *Proceedings of the 15th International Conference on Extending Database Technology*, EDBT '12, pages 336–347, New York, NY, USA, 2012. ACM.

[4] K. Bontcheva and D. Rout. Making Sense of Social Media Streams through Semantics: a Survey. *Semantic Web*, 2012.

[5] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Comput. Surv.*, 41(3), 2009.

[6] L. Chen and A. Roy. Event detection from Flickr data through wavelet-based spatial analysis. In *Proceedings of the 2009 ACM CIKM International Conference on Information and Knowledge Management (CIKM '09)*, 2009.

[7] A. Dries and U. Rückert. Adaptive concept drift detection. *Stat. Anal. Data Min.*, 2:311–327, 2009.

[8] K. Gimpel, N. Schneider, B. O'Connor, D. Das, D. Mills, J. Eisenstein, M. Heilman, D. Yogatama, J. Flanigan, and N. A. Smith. Part-of-Speech Tagging for Twitter: Annotation, Features, and Experiments. In *ACL (Short Papers)*, pages 42–47, 2011.

[9] D. Kifer, S. Ben-David, and J. Gehrke. Detecting change in data streams. In *Proceedings of the Thirtieth international conference on Very large data bases - Volume 30*, VLDB '04, pages 180–191. VLDB Endowment, 2004.

[10] A. Marcus, M. S. Bernstein, O. Badar, D. R. Karger, S. Madden, and R. C. Miller. Twitinfo: aggregating and visualizing microblogs for event exploration. In *Proceedings of the 2011 annual conference on Human factors in computing systems*, CHI '11, pages 227–236. ACM, 2011.

[11] M. Markou and S. Singh. Novelty detection: A review - part 1: Statistical approaches. *Signal Processing*, 83:2003, 2003.

[12] K. Nishida, T. Hoshide, and K. Fujimura. Improving tweet stream classification by detecting changes in word probability. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '12, pages 971–980, New York, NY, USA, 2012. ACM.

[13] A. Ritter, Mausam, O. Etzioni, and S. Clark. 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, 2012. ACM.

[14] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes Twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*, WWW '10, pages 851–860. ACM, 2010.

[15] J. Schühmacher and C. Koster. Signalling events in text streams. In P. Daras and O. Mayora-Ibarra, editors, *UCMedia*, volume 40 of *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pages 335–339. Springer, 2009.

[16] K. Sparck Jones. *A statistical interpretation of term specificity and its application in retrieval*, pages 132–142. Taylor Graham Publishing, 1988.

[17] Twitter Team. Developing for @twitterapi (Techcrunch Disrupt Hackathon), 2012, https://dev.twitter.com/docs/intro-twitterapi.

[18] J. Weng, Y. Yao, E. Leonardi, and F. Lee. Event Detection in Twitter. Technical report, HP Labs, 2011.

[19] Y. Yang, T. Pierce, and J. Carbonell. A study of retrospective and on-line event detection. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '98, pages 28–36. ACM, 1998.