Beyond Trending Topics: Real-World Event Identification on Twitter

Hila Becker
Columbia University
hila@cs.columbia.edu

Mor Naaman
Rutgers University
mor@rutgers.edu

Luis Gravano
Columbia University
gravano@cs.columbia.edu

Abstract

User-contributed messages on social media sites such as Twitter have emerged as powerful, real-time means of information sharing on the Web. These short messages tend to reflect a variety of events in real time, making Twitter particularly well suited as a source of real-time event content. In this paper, we explore approaches for analyzing the stream of Twitter messages to distinguish between messages about real-world events and non-event messages. Our approach relies on a rich family of aggregate statistics of topically similar message clusters. Large-scale experiments over millions of Twitter messages show the effectiveness of our approach for surfacing real-world event content on Twitter.

1 Introduction

Social media sites (e.g., Twitter, Facebook, and YouTube) have emerged as powerful means of communication for people looking to share and exchange information on a wide variety of real-world events. These events range from popular, widely known ones (e.g., a concert by a popular music band) to smaller scale, local events (e.g., a local social gathering, a protest, or an accident). Short messages posted on social media sites such as Twitter can typically reflect these events as they happen. For this reason, the content of such social media sites is particularly useful for real-time identification of real-world events and their associated user-contributed messages, which is the problem that we address in this paper.

Twitter messages reflect useful event information for a variety of events of different types and scale. These event messages can provide a set of unique perspectives, regardless of the event type (Diakopoulos, Naaman, and Kivran-Swaine 2010; Yardi and boyd 2010), reflecting the points of view of users who are interested or participate in an event. In particular, for unplanned events (e.g., the Iran election protests, earthquakes), Twitter users sometimes spread news prior to the traditional news media (Kwak et al. 2010; Sakaki, Okazaki, and Matsuo 2010). Even for planned events (e.g., the 2010 Apple Developers conference), Twitter users often post messages in anticipation of the event.

Identifying events in real time on Twitter is a challenging problem, due to the heterogeneity and immense scale of the data. Twitter users post messages with a variety of content types, including personal updates and various bits of information (Naaman, Boase, and Lai 2010). While much of the content on Twitter is not related to any particular real-world event, informative event messages nevertheless abound. As an additional challenge, Twitter messages, by design, contain little textual information, and often exhibit low quality (e.g., with typos and ungrammatical sentences).

Several research efforts have focused on identifying events in social media in general, and on Twitter in particular (Becker, Naaman, and Gravano 2010; Sakaki, Okazaki, and Matsuo 2010; Sankaranarayanan et al. 2009). Recent work on Twitter has started to process data as a stream, as it is produced, but has mainly focused on identifying events of a particular type (e.g., news events (Sankaranarayanan et al. 2009), earthquakes (Sakaki, Okazaki, and Matsuo 2010)). Other work identifies the first Twitter message associated with an event (Petrović, Osborne, and Lavrenko 2010).

Our focus in this work is on *online identification* of real-world event content. We identify each event—and its associated Twitter messages—using an online clustering technique that groups together topically similar tweets (Section 3.1). We then compute revealing features for each cluster to help determine which clusters correspond to events (Section 3.2). We use these features to train a classifier to distinguish between event and non-event clusters (Section 3.3). We validate the effectiveness of our techniques using a dataset of over 2.6 million Twitter messages (Section 4) and then discuss our findings and future work (Section 5).

2 Background and Problem Definition

In this section, we provide an overview of Twitter and then define the problem that we address in this paper.

2.1 Background: Twitter

Twitter is a popular social media site that allows users to post short textual messages, or *tweets*, which are up to 140 characters long. Twitter users can use a *hashtag* annotation format (e.g., #sb45) to indicate what their messages are about (e.g., "watching Superbowl 45 #sb45"). In addition, Twitter allows several ways for users to converse and interact by referencing each other in messages using the @ symbol. Twitter currently employs a proprietary algorithm to display *trending topics*, consisting of terms and phrases that exhibit "trending" behavior. While Twitter's trending topics sometimes reflect current events (e.g., "world cup"), they often include keywords for popular conversation topics (e.g., "#bieberfever," "getting ready"), with no discrimination between the different types of content.

2.2 Problem Definition

We now define the notion of real-world event in the context of a Twitter message stream, and provide a definition of the problem that we address in this paper.

The definition of event has received attention across fields, from philosophy (Events 2002) to cognitive psychology (Zacks and Tversky 2001). In information retrieval, the concept of event has prominently been studied for event detection in news (Allan 2002). We borrow from this research to define an event in the context of our work. Specifically, we define an *event* as a real-world occurrence e with (1) an associated time period T_e and (2) a time-ordered stream of Twitter messages M_e, of substantial volume, discussing the occurrence and published during time T_e.

According to this definition, events on Twitter include widely known occurrences such as the presidential inauguration, and also local or community-specific events such as a high-school homecoming game or the ICWSM conference. Non-event content, of course, is prominent on Twitter and similar systems where people share various types of content such as personal updates, random thoughts and musings, opinions, and information (Naaman, Boase, and Lai 2010).

As a challenge, non-event content also includes forms of Twitter activity that trigger substantial message volume over specific time periods (Becker, Naaman, and Gravano 2011b), which is a common characteristic of event content. Examples of such non-event activity are Twitter-specific conversation topics or *memes* (e.g., using the hashtag #thingsparentssay). Our goal is to differentiate between messages about real-world events and non-event messages, where non-event messages include those for "trending" activities that are Twitter-centric but do not reflect any real-world occurrences. We now define our problem, as follows:

> *Consider a time-ordered stream of Twitter messages M. At any point in time t, our goal is to identify real-world events and their associated Twitter messages present in M and published before time t. Furthermore, we assume an online setting for our problem, where we only have access to messages posted before time t.*

3 Separating Event and Non-Event Content

We propose to address the event identification problem using an online clustering and filtering framework. We describe this framework in detail (Section 3.1), and then discuss the different types of features that we extract for clusters (Section 3.2), as well as the classification model that we use (Section 3.3) to separate event and non-event clusters.

3.1 Clustering and Classification Framework

We elected to use an incremental, online clustering algorithm in order to effectively cluster a stream of Twitter messages in real time. For such a task, we must choose a clustering algorithm that is scalable, and that does not require *a priori* knowledge of the number of clusters, since Twitter messages are constantly evolving and new events get added to the stream over time. Based on these observations, we propose using an incremental clustering algorithm with a threshold parameter that is tuned empirically during a training phase. Such a clustering algorithm considers each message in turn, and determines a suitable cluster assignment based on the message's similarity to existing clusters. (See (Becker, Naaman, and Gravano 2011a) for further details.)

To identify all *event* clusters in the stream, we compute a variety of revealing features using statistics of the cluster messages (Section 3.2). Since the clusters constantly evolve over time, we must periodically update the features for each cluster and compute features of newly formed clusters. We subsequently proceed to invoke a classification model (Section 3.3) that, given a cluster's feature representation, decides whether or not the cluster, and its associated messages, contains event information. With the appropriate choice of classification model, we can also select the top events in the stream at any point in time, according to the clusters' probability of belonging to the event class.

3.2 Cluster-Level Event Features

We compute features of Twitter message clusters in order to reveal characteristics that may help detect clusters that are associated with events. We examine several broad categories of features that describe different aspects of the clusters we wish to model. Specifically, we consider temporal, social, topical, and Twitter-centric features. We summarize these features below. (See (Becker, Naaman, and Gravano 2011a) for further details.)

Temporal Features: The volume of messages for an event e during the event's associated time T_e exhibits unique characteristics (see the definition of event in Section 2.2). To effectively identify events in our framework, a key challenge is to capture this temporal behavior with a set of descriptive features for our classifier. We design a set of temporal features to characterize the volume of frequent cluster terms (i.e., terms that appear frequently in the set of messages associated with a cluster) over time. These features capture any deviation from expected message volume for any frequent cluster term or a set of frequent cluster terms. Additionally, we also compute the quality of fit of an exponential function to the term's hourly binned message histogram.

Social Features: We designed social features to capture the interaction of users in a cluster's messages. These interactions might be different between events, Twitter-centric activities, and other non-event messages (Becker, Naaman, and Gravano 2011b). User interactions on Twitter include retweets (forwarding, indicated by RT @username), replies (conversation, indicated by @username in the beginning of the tweet), and mentions (indicated by @username anywhere except the beginning of the tweet). Our social features include the percentage of messages containing each of these types of user interaction out of all messages in a cluster.

Topical Features: Topical features describe the topical coherence of a cluster, based on a hypothesis that event clusters tend to revolve around a central topic, whereas non-event clusters do not. Rather, non-event clusters often center around a few terms (e.g., "sleep," "work") that do not reflect a single theme (e.g., with some messages about sleep, others about work, and a few about sleeping at work). Messages in event clusters are likely to share more terms, as they identify key aspects of the events they describe (e.g., "Couric," "Obama," and "interview" are common among messages describing Katie Couric's interview of President Obama).

Twitter-Centric Features: While the goal of our classifier is to distinguish between event and non-event data, we highlight the differences between non-event clusters that correspond to Twitter-centric activities, which are a specific class of non-event messages (Section 2.2), and the real-world event clusters that we wish to identify. As discussed above, Twitter-centric activities often exhibit characteristics that resemble real-world events, especially as captured by temporal features, which generally offer a strong signal for the presence of event content. To address this challenge, we design a set of features that target commonly occurring patterns in non-event clusters with Twitter-centric behavior, including tag usage, and presence of multi-word hashtags.

3.3 Event Classification

Using the above features, we train an event classifier by applying standard machine learning techniques (see Section 4). This classifier predicts which clusters correspond to events at any point in time (i.e., at any point in the stream; see Section 2.2). Specifically, to identify event clusters at the end of hour h, we first compute the features of all clusters with respect to h, and then use the classification model with each cluster's feature representation to predict the probability that the cluster contains event information.

Due to the large volume of data on Twitter, it is possible that at any point in time our classifier may label many clusters as events. In an event browsing scenario, where users look for information on current events, it is essential to display a select subset of these identified event clusters. To that end, we are interested in the ability of our classifier to select the top events according to their probability of belonging to the event class. We compare the results of our classifier against several baseline approaches next.

4 Experiments

We evaluated our event identification strategies on a large dataset of Twitter data. We describe this dataset and report the experimental settings (Section 4.1), and then turn to the results of our experiments (Section 4.2).

4.1 Experimental Settings

Data: Our dataset consists of over 2,600,000 Twitter messages posted during February 2010 (Becker, Naaman, and Gravano 2011a). Since we are interested in identifying events both with local and with broad geographical interest, we collected these messages from users who identified their location as New York City. We cluster our dataset in an online fashion as described in Section 3.1. We use the data from the first two weeks in February for training and report our results on test data from the last two weeks in February.

Annotations: We use human annotators to label clusters for both the training and testing phases of our experiments. For complete details of our annotation guidelines, methodology, and annotator agreement measures, please refer to (Becker, Naaman, and Gravano 2011a).

For the training set, we annotated 504 clusters, randomly selected from the top-20 fastest-growing clusters according to hourly message volume at the end of each hour in the second week of February 2010. After removing 34 ambiguous clusters and dropping 96 clusters on which the annotators disagreed, we were left with 374 clusters. For the test set, we used 300 clusters collected at the end of five different hours in the third and fourth weeks of February 2010. At the end of each hour we select the 20 fastest-growing clusters according to hourly volume, the top-20 clusters according to our classifier (Section 3.3), and 20 random clusters, for a total of 100 clusters per method over the five hours.

Training Classifiers: We train a classifier to distinguish between real-world event and non-event clusters (*RW-Event*). We extracted cluster-level features for each cluster in the training set (Section 3.2) and used the Weka toolkit (Witten and Frank 2005) to train our classification model. We first applied a resampling filter to balance the class distribution, which was skewed towards the non-event class, and then we trained and evaluated the classifier using 10-fold cross validation. We explored a variety of classifier types and selected support vector machines (specifically, Weka's sequential minimal optimization implementation) for *RW-Event*, as it yielded the best overall performance in exploratory tests over the training set. We also fit logistic regression models to the output of the support vector machine, to obtain probability estimates of the class assignment.

As a baseline, we use a strong text classification approach based on the textual content of the messages in the cluster. Specifically, we trained a Naïve Bayes classifier (*NB-Text*) that treats all messages in a cluster as a single document, and uses the *tf-idf* weights of textual terms as features. This classifier, distinguishing between events and non-events, is similar to the one used by Sankaranarayanan et al. (2009) for identifying news in Twitter messages.

Evaluation: To evaluate the performance of each classifier, we use the macro-averaged F_1 metric (Manning, Raghavan, and Schütze 2008). This evaluation metric is widely used and is effective for evaluating classification results where it is desirable to assign an equal weight to the classifier's performance on each class.

We also evaluate our classifiers' ability to identify events among a set of top clusters, ordered by their probability of belonging to the event class at the end of each hour. As a baseline for this "event surfacing" task, we consider the event thread selection approach presented by Petrović et al. (2010), which selects the fastest-growing threads in a stream of Twitter messages (*Fastest*). In addition, we compare our approach against a technique that selects clusters randomly (*Random*).

To evaluate the event surfacing task, we use *Precision*@K, which captures the quality of ranked lists with focus on the top results. Precision@K reports the fraction of correctly identified events out of the top-K selected clusters, averaged over all hours.

4.2 Experimental Results

We begin by examining the performance of our *RW-Event* classifier against the *NB-Text* baseline classifier. The performance on the training set reflects the accuracy of each clas-

Classifier	Training	Test
NB-Text	0.785	0.702
RW-Event	**0.849**	**0.837**

Table 1: F_1 score of our classifiers on training and test sets.

Description	Terms
Westminster Dog Show	westminster, dog, show, club
Obama & Dalai Lama meet	lama, dalai, meet, obama, china
NYC Toy Fair	toyfairny, starwars, hasbro, lego
Marc Jacobs Fashion Show	jacobs, marc, nyfw, show, fashion

Table 2: Sample events identified by the *RW-Event* classifier.

sifier computed using 10-fold cross-validation. The performance on the test set measures how well each classification model predicts on the test set of 100 randomly selected clusters. Table 1 shows the F_1 scores of the classifiers on both the training and test sets. As we can see, *RW-Event* outperformed *NB-Text* over both training and test sets, showing that it is overall more effective in predicting whether or not our clusters contain real-world event information. A deeper examination of our results revealed that *NB-Text* was especially weak at classifying event clusters, accurately predicting only 25% of event clusters on the test set. A sample of event clusters identified by *RW-Event*, and their most frequent terms, are presented in Table 2.

The next set of results describes how well *RW-Event* performs for the "event surfacing" task. Recall that the goal of this task is to identify the top events in the stream per hour. We report Precision@K (Figure 1) scores for varying K, averaged over the five hours selected for the test set. We compared the results of *RW-Event* to two baselines: *Fastest* and *Random* (Section 4.1). Not surprisingly, the proportion of events identified by *Random* is very low, as most data on Twitter does not contain event information. The proportion of events identified by *Fastest* was higher than that of *Random*. *RW-Event* performed well across the board, better than both baselines according to precision.

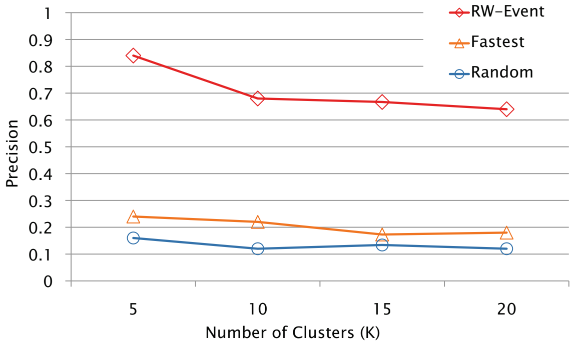

Figure 1: Precision @ K for our classifier and baselines.

5 Conclusions

We presented an end-to-end approach for identifying real-world event content on Twitter. This work provides the first step in a series of tools that improve on the generic analysis of "trending topics." In future work, we aim to reason even more finely about different types of events that are reflected in Twitter data. Given a robust classification of events, extending the work described here, we can improve prioritization, ranking, and filtering of extracted content on Twitter and similar systems, as well as provide more targeted and specialized content visualization.

6 Acknowledgments

This material is based on work supported by NSF Grants IIS-0811038, IIS-1017845, and IIS-1017389, and by two Google Research Awards. In accordance with Columbia Univ. reporting requirements, Prof. Gravano acknowledges ownership of Google stock as of the writing of this paper.

References

Allan, J., ed. 2002. *Topic Detection and Tracking: Event-based Information Organization*. Kluwer Academic Publisher.

Becker, H.; Naaman, M.; and Gravano, L. 2010. Learning similarity metrics for event identification in social media. In *WSDM'10*.

Becker, H.; Naaman, M.; and Gravano, L. 2011a. Beyond trending topics: Real-world event identification on Twitter. Technical Report cucs-012-11, Columbia University.

Becker, H.; Naaman, M.; and Gravano, L. 2011b. Hip and trendy: Characterizing emerging trends on Twitter. *JASIST*. To appear.

Diakopoulos, N.; Naaman, M.; and Kivran-Swaine, F. 2010. Diamonds in the rough: Social media visual analytics for journalistic inquiry. In *VAST'10*.

Events. 2002. In Stanford Encyclopedia of Philosophy. Retrieved June 2nd, 2010 from `http://plato.stanford.edu/entries/events/`.

Kwak, H.; Lee, C.; Park, H.; and Moon, S. 2010. What is Twitter, a social network or a news media? In *WWW'10*.

Manning, C. D.; Raghavan, P.; and Schütze, H. 2008. *Introduction to Information Retrieval*. Cambridge Univ. Press.

Naaman, M.; Boase, J.; and Lai, C.-H. 2010. Is it really about me?: Message content in social awareness streams. In *CSCW'10*.

Petrović, S.; Osborne, M.; and Lavrenko, V. 2010. Streaming first story detection with application to Twitter. In *NAACL'10*.

Sakaki, T.; Okazaki, M.; and Matsuo, Y. 2010. Earthquake shakes Twitter users: Real-time event detection by social sensors. In *WWW'10*.

Sankaranarayanan, J.; Samet, H.; Teitler, B. E.; Lieberman, M. D.; and Sperling, J. 2009. Twitterstand: News in tweets. In *GIS'09*.

Witten, I. H., and Frank, E. 2005. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann.

Yardi, S., and boyd, d. 2010. Tweeting from the town square: Measuring geographic local networks. In *ICWSM'10*.

Zacks, J. M., and Tversky, B. 2001. Event structure in perception and conception. *Psychological Bulletin* 127.