Effective Event Identification in Social Media

Fotis Psallidas
fotis@cs.columbia.edu
Columbia University

Hila Becker
hila@google.com
Google, Inc.

Mor Naaman
mor.naaman@cornell.edu
Cornell Tech

Luis Gravano
gravano@cs.columbia.edu
Columbia University

Abstract

Online social media sites are extensively used by individuals to produce and distribute content related to real-world events. Unfortunately, this social media content associated with an event is generally not provided in any structured and readily available form. Thus, identifying the event-related content on social media sites is a challenging task. Prior work has addressed the event identification task under two different scenarios, namely, when the events are known ahead of time, as is sometimes the case for planned events, and when the events are unknown, as is the case for spontaneous, unplanned events. In this article, we discuss both the unknown- and known-event identification scenarios, and attempt to characterize the key factors in the identification process, including the nature of social media content as well as the behavior and characteristics of event content over time. Furthermore, we propose enhancements to our earlier techniques that consider these factors and improve the state-of-the-art unknown-event identification strategies. Specifically, we propose novel features of the social media content that we can exploit, as well as the modeling of the typical time decay of event-related content. Large-scale experiments show that our approach exhibits improved effectiveness relative to the state-of-the-art approaches.

1 Introduction

Online social media sites (e.g., Flickr, YouTube, Twitter) serve as the main outlet for individuals to distribute and receive meaningful content about real-world events. This content may appear in various forms, including status updates, photos, and videos, that can be created or posted before, during, and after an event. Furthermore, for known and planned events, structured information (e.g., title, time, location) might be available through event-aggregation social media sites (e.g., Facebook Events, Meetup, EventBrite). Such prior knowledge, however, is not available for unknown or spontaneous events (e.g., natural disasters). By automatically identifying the social media content related to either known or unknown events, which is the focus of this article, we can enhance powerful event browsing and search.

In the known-event identification scenario, information is available explicitly online. For instance, consider the 2013 NBA All-Star game in Houston, Texas. Structured information about the event (e.g., time and location) may be available on a related Facebook Event page. Because this information may be limited, though, users may seek to obtain additional content for the event from other social media sites (e.g., check what Twitter users discuss about the event or watch the game again on YouTube after the event). Thus, automatically identifying social media content related to known events enhances the event-based experience a user may seek by providing meaningful information before, during, and after an event.

Bulletin of the IEEE Computer Society Technical Committee on Data Engineering

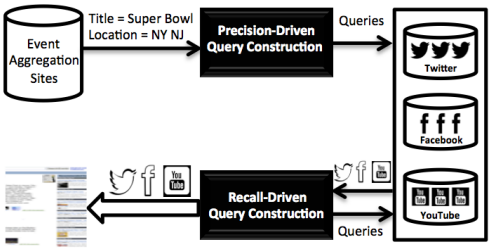


Figure 1: Known-event identification: Retrieving relevant content for the Super Bowl event, starting with its (known) title and location.

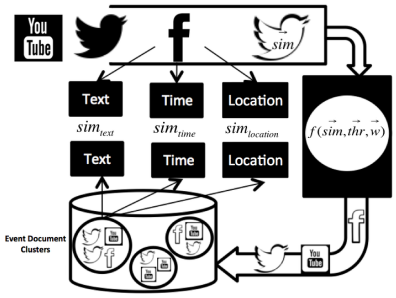


Figure 2: Unknown-event identification: Online clustering over a stream of social media content using time, text, and location as weak indicators.

In contrast, in the unknown-event identification scenario there is no a priori knowledge of the events. For instance, consider an unplanned presidential announcement on a political issue. Such an event, although it is not known beforehand, prompts individuals to react and discuss, often extensively (e.g., through Twitter messages discussing the event or YouTube videos of the announcement). Just as in the known-event scenario, users also turn to social media in search of content for the event, both during the event and after it has ended.

Attempting to automatically identify event-related social media content in both scenarios is a challenging task. First, social media content is noisy and heterogeneous. For instance, Twitter messages are short, informal, and often grammatically incorrect. The event identification process should then account for these characteristics of social media content. Furthermore, social media content is not always related to an event. For instance, a Facebook post that states "@alice I am impressed by your new song!!" corresponds to chatter or social activity, and is not event-related. The event identification process should then discard such non-event content. Finally, the manner in which event-related discussions evolve in social media affects the event identification process. For example, the fact that an event has been inactive for a long time might serve as a strong indicator that new social media content is unlikely to correspond to the event, as we argue in Section 3.

Most related research identifies event-related social media content by considering only a subset of these factors. For instance, Shakaki et al. [7] identify event-related social media content by using a predefined set of terms (e.g., "earthquake") as keywords that social media documents should contain. However, social media documents are noisy and heterogeneous, and might not contain a specific keyword or have arbitrary variations thereof (e.g., "eaaarthquaaake"). Improving on this point, Sankaranarayanan et al. [8] use both textual and temporal features to reduce the impact of the noisy data, but they ignore other features, such as location. Also, to further distinguish between event and non-event documents their methodology assumes the existence of "seeders," which are treated as always producing event-related content. However, these seeders might distribute documents unrelated to events that the identification process should differentiate from the event-related ones.

In this article, we first summarize our earlier work on known- and unknown-event identification [2, 3, 4], which addresses many of the factors discussed above (Section 2). Furthermore, we also propose novel techniques to better address these key factors in the context of the unknown-event identification scenario (Section 3). More precisely, we introduce novel features to address the noise and heterogeneity of the social media content, as well as a time decay function that fits the behavior of event-related discussions over time. Finally, we evaluate our techniques using a large real-world dataset compiled from photo-sharing social media site Flickr (Section 4). Our experiments show that our proposed techniques substantially improve the effectiveness for the unknown-event identification scenario compared to our baselines.

2 Event Content Identification in Social Media

Major social media sites are popular venues for publishing rich and diverse information about a variety of real-world events. As an example of such an event, consider the 2014 NFL's Super Bowl in New York and New Jersey. Event-aggregation platforms (e.g., Facebook Events, Meetup, EventBrite) might list a description of this event, including some of its prominent features (e.g., title, time, location). Also, individuals share their reactions and discuss the event on different social media sites (e.g., Flickr, YouTube, Twitter). Such content is

highly valuable, because it offers a user perspective of events that would otherwise not be found on the Web. Unfortunately, this meaningful user-contributed information is not readily available in any structured form, so it is generally unclear what social media content refers to which event. Overall, automatically identifying event content poses interesting challenges, because the social media content is noisy and highly heterogeneous.

To address the event identification task, we consider two substantially different scenarios. The first scenario corresponds to events that are known ahead of time, as is the case for planned events announced, say, on Facebook Events or Meetup. In the second scenario, we do not have have any a priori knowledge of an event, either because the event occurred unexpectedly (e.g., an earthquake or an automobile accident) or because the event was not announced online (e.g., a local street fair or a birthday celebration). Both scenarios of the event identification task, namely, *known-event identification* and *unknown-event identification*, present distinctive challenges and opportunities, which we discuss next:

Known-Event Identification: In this scenario, key properties of the events (e.g., title, time, location) are known ahead of time, usually posted on event-aggregation social media sites. Social media content related to these known events could reside in multiple social media sites, each contributing different information about the event. For instance, YouTube might contain videos for the Super Bowl event, whereas Twitter users might discuss the event by sharing short text messages, or *tweets*. To retrieve cross-site social media documents associated with the Super Bowl event, we could then use the APIs provided by major social media sites, such as the Twitter or YouTube APIs, to formulate high-precision queries using the event's known features (e.g., a [2014 NFL Super Bowl] query using the event title "2014 NFL Super Bowl") and extract the related tweets and YouTube videos. However, such highly specific queries tend to retrieve event-related documents with high precision but with low recall, meaning that they miss many relevant event documents.

In our previous work [2], we have addressed these challenges by proposing a two-step query formulation approach. Figure 1 illustrates this process for our example. In the first step, we use highly specific queries, using the known event properties of an event, to achieve high-precision results. For instance, in Figure 1 we use the title and the location of the Super Bowl to construct queries that retrieve related YouTube videos, Facebook posts, and tweets. As we mentioned, however, these high-precision queries result in low recall. Thus, the second step builds on these high-precision cross-site results, using term extraction and frequency analysis, aiming to improve recall and contribute to the high quality and diversity of the identified information. Interestingly, we also proposed ways to leverage the event-related social media content retrieved from one social media site, retrieved using high-precision queries, to obtain additional content from other social media sites [2]. This task is important in the case where the high-precision queries do not return sufficiently many results from some of the available social media sites (e.g., querying YouTube using the title of a local event might not yield any results). Thus, we proposed using the content from one social media site (e.g., Twitter) to build recall-oriented queries to be used on other social media sites (e.g., YouTube).

Unknown-Event Identification: In contrast to the known-event identification scenario, in the unknown-event identification scenario we do not have any a priori information about the events and their properties. Therefore, we are presented with a stream of event-related social media documents without any knowledge of the events that may be reflected in the stream. For example, a Twitter stream (see Table 1) may contain many tweets related to an event (e.g., a high-profile announcement of U.S. President Obama about the Middle-East) interspersed with messages related to other events (e.g., Apple's announcement of new iPhone models) as well as messages unrelated to events (e.g., Bob mentioning to Alice his favorite illusionist). Automatically identifying unknown-event content in this scenario poses some interesting challenges both due to the high rate of the social streams and the noisy nature of the data.

To address these challenges, we proposed an online clustering framework (see Figure 2) that leverages the multiple features associated with each social media document (e.g., publisher, text, and time for the tweets in Table 1) [3]. These features help define weak indicators of event-related content and collectively produce stronger document-similarity judgments, to decide when two social media documents correspond to the same event, than when used individually. To see why, consider the first tweet in Table 1, from The New York Times,

Publisher	*Text*	*Time*
New York Times	Apple Shows Off 2 New iPhones, One a Lower-Cost Model http://nyti.ms/19EP2DI	10:06 PM - 10 Sep 13
iPhone News	Apple changes up colors: 'space gray' comes to iPhone 5s & iPods http://dlvr.it/3xdkm5 #iPhone	03:44 AM - 11 Sep 13
Wall Street Journal	President Obama addresses the nation on #Syria. Follow our live blog: http://on.wsj.com/1aoGPV0	04:05 AM - 11 Sep 13
Bob	@alice He is my favorite illusionist davidcopperfield.com ...	04:08 AM - 11 Sep 13
Huffington Post	Obama now: "I will not put American boots on the ground in Syria." http://wapo.st/1aoHmX8	04:10 AM - 11 Sep 13
IGN	Apple iPhone 5s vs. iPhone 5c: Which phone should you buy? - MobileThe iPhone 5c is $100 cheaper than the iPhone	04:14 AM - 11 Sep 13
New York Times	Breaking News: Obama Asks Congress to Postpone a Vote on Syria Action	04:22 AM - 11 Sep 13

Table 1: Twitter messages including both event and non-event content.

discussing the iPhone 5c release. Judging solely based on the text, the tweet might (erroneously) be linked to earlier Apple announcements involving iPhones. To link the tweet to the correct event, however, we can exploit the fact that the publication time of the New York Times tweet (first row of Table 1) is close to the publication time of other tweets that explicitly discuss the iPhone 5c release. In contrast, considering the publisher and text together does not assist in correctly determining the correct event content in the example above.

Different features of social media documents thus have significantly varying impact on the final clustering —and hence unknown-event identification— decision. Based on this observation, we have proposed [3] an ensemble learning methodology that decides for each feature (e.g., text, time, and location in Figure 2) (a) how more indicative of event-related content it can be compared to the rest of the features (e.g., the time feature is considered more revealing than the publisher one in our example), and (b) under what circumstances its judgment on the similarity of two social media documents (e.g., $sim_{text}(d, d')$, $sim_{time}(d, d')$, $sim_{location}(d, d')$ in Figure 2, where d, d' are social media documents) is considered trustworthy. More precisely, we deployed ensemble learning methods to learn and associate each feature with a weight and a threshold that capture the importance of the features. Furthermore, based on the set of weights $\vec{w}$, thresholds $\vec{thr}$, and individual feature similarity functions, we constructed a "consensus function" $f(\vec{sim}(d, C), \vec{w}, \vec{thr})$ that plays the role of the final similarity function between a document d and a cluster C in the document clustering process (see Figure 2). Cluster C can be alternatively represented as its centroid vector c (i.e., $\vec{sim}(d, C) = \vec{sim}(d, c)$) or by all the cluster documents (i.e., $\vec{sim}(d, C) = \frac{1}{|C|} \cdot \sum_{d' \in C} \vec{sim}(d, d')$).

Social media documents, however, are not always event-related. For instance, the tweet by Bob (see Table 1) does not relate to any event. Therefore, after clustering the social media documents, we need to identify which clusters correspond to events and which ones do not. For this, we deployed event classification techniques that operate on the output of the clustering process. Specifically, they help distinguish between event-related clusters and non-event ones (e.g., clusters that contain event-related social media documents, such as those related to the iPhone 5c release, and clusters that contain chatter and social activity, like Bob's message). Technically, our event classification techniques rely on a rich family of aggregate cluster statistics, including temporal (e.g., frequency of the terms associated with a cluster), social (e.g., proportion of retweets and mentions of a tweet within a cluster), topical (e.g., topical coherence in a cluster), and platform-centric (e.g., tags and presence of multi-word hashtags), as best suitable indicators for event and non-event content separation.

The overall process above yields high-quality results [3, 4] on the unknown-event identification task. Nonetheless, its effectiveness can be further improved. For instance, the text feature of Figure 2 would treat the terms Obama and Syria as equal to the terms follow and blog of the Wall Street Journal tweet in Table 1. The former play an important role with respect to the corresponding event, while the latter only add noise. Furthermore, events evolve in certain ways over time. For instance, by the time tweets were discussing the iPhone 5c model release (see Table 1), discussions about previous iPhone releases were limited. If the first social document about the iPhone 5c release does not explicitly mention the 5c model (e.g., as is the case with the first tweet by The New York Times in Table 1), the clustering procedure may erroneously relate the new event to the iPhone 4 release event it has already identified, hence compromising the effectiveness of the clustering procedure. Next, we introduce a variety of refinements to our techniques to improve the effectiveness of the unknown-event identification task. (Improving the known-event identification task remains the subject of our future work.)

Feature Type	*Example*	Jaccard Similarity
Whole URL	mashable.com/2013/09/08/bruno-mars-super-bowl-halftime-show-confirmed rollingstone.com/music/news/bruno-mars-will-perform-at-super-bowl-20130908	0.0
Parsed URL	2013, 09, 08, bruno, mars, super, bowl, halftime, show, confirmed music, news, bruno, mars, will, perform, at, super, bowl, 20130908	0.25
Parsed Query Part of URL	bruno, mars, super, bowl, halftime, show, confirmed, bruno, mars, will, perform, at, super, bowl, 20130908	0.36

Table 2: Extracted URL features for two URLs and the Jaccard similarity for each feature.

3 Improving Unknown-Event Identification Effectiveness

The textual features of the social media documents (e.g., text, publisher of the tweets in Table 1) and how events behave over time have a significant impact on the effectiveness of the document clustering procedure. How to refine the clustering procedure to benefit from these factors is a challenging task that we discuss next. Specifically, we discuss new features for the unknown-event identification process, namely, "URLs" and "bursty vocabularies." Then we show how to leverage the typical temporal characteristics of event content.

URLs: URLs in event-related social streams are ubiquitous. Individuals use them to share meaningful event-related external content. Furthermore, using URLs assists users to adhere to the limited-length text that characterizes the social media documents. For instance, we can directly discuss the confirmation of the Super Bowl's halftime performer using an appropriate URL instead of explicitly describing the choice of performer. To capitalize on this behavior, we propose the modeling of three new textual features as event indicators, namely, *whole URL*, *parsed URL*, and *parsed query part of URL* (see Table 2). Intuitively, the *whole URL* feature captures the similarity of social media documents that use the same URL. However, multiple sources might discuss the same event-related content, a case where the *whole URL* feature fails to identify their similarity (see Table 2). In such cases, the *parsed URL* feature appropriately tokenizes the URL, as illustrated in Table 2, and uses the extracted tokens to identify the underlying similarity. However, different sources use different URL patterns, so the *parsed URL* might introduce noise into the underlying similarity metric. For instance, the *parsed URL* for the Mashable URL in Table 2 contains the terms 2013, 09, and 08 while the Rolling Stones *parsed URL* contains the terms music and news, which are both not descriptive and add noise to the similarity computation. To address this challenge, we finally introduce the *parsed query part of URL* as a highly indicative feature of event content. We note, however, that there are cases (e.g., http://tinyurl.com/nove79c[1]) where the *parsed query part of URL* feature fails to recognize the underlying similarity and the *parsed URL* feature performs substantially better.

Bursty Vocabulary: The social media content related to an event tends to revolve around a central topic. In social media documents related to an event, this central topic is expressed by a set of terms that is significantly more frequent than the rest of the terms (e.g., NFL, Super, and Bowl for the 2014 NFL Super Bowl event). However, events that span a wide time range typically exhibit a different set of these *bursty* terms at different points of their lifetime. Figure 4 shows three different time windows for the Super Bowl event. Initially, terms like Super, Bowl, and NFL are the most bursty. Then, the announcement of the location of the Super Bowl (i.e., New York and New Jersey) triggers terms like NY and NJ to exhibit bursty behavior. Finally, the announcement of Bruno Mars as the Super Bowl halftime performer causes terms such as Bruno and Mars to exhibit bursty behavior. To capture the terms associated with an event that exhibit a bursty behavior within a given time window, we introduce the notion of *bursty vocabulary* per time window.

Technically, we tailor the notion of bursty vocabulary within the clustering framework [3] using the following methodology: For each cluster, at the end of a time window W_i, we extract the bursty vocabulary $BurstyV_i$, which we model as a weak indicator for the next time window W_{i+1}. To determine if a term is about to be bursty in the next time window, we follow a technique similar to the one proposed in [5]. More precisely, we say that a term *t* is bursty in W_{i+1} if the expected number of occurrences of *t* in W_{i+1} is higher than the average number of occurrences that the term had in the previous time windows. To compute the expected occurrences of a term *t* in the time window W_{i+1} we can model the probability of the occurrences of the term *t* by a hyper-geometric distribution [5]. Here, however, we use the less computationally expensive binomial distribution, which in fact

[1]The url http://tinyurl.com/nove79c resolves to http://www.usatoday.com/story/sports/nfl/2013/09/07/bruno-mars-super-bowl-halftime-show-metlife-stadium-february-2014/2779895.

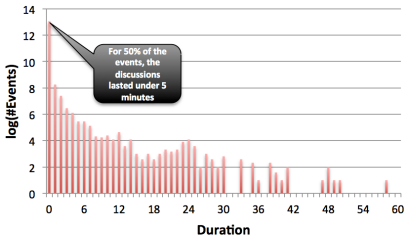


Figure 3: Histogram of the log-scaled number of events as a function of their discussion time (step $\Delta t = 15$ days).

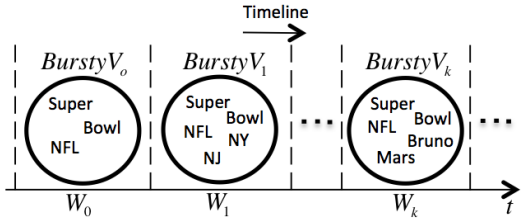
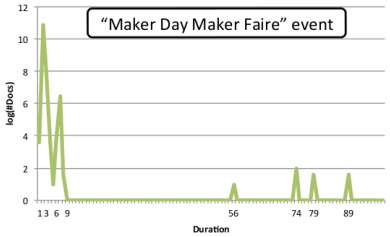
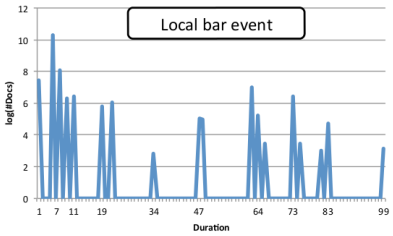


Figure 4: Bursty vocabulary of the Super Bowl event in various time windows (W_i and $BurstyV_i$ denote the i-th time window and its bursty vocabulary).



(a)



(b)

Figure 5: Example of two event discussions, over duration time, expressed as the log-scaled number of social media documents discussing the event (step $\Delta t \approx 4$ and 7 days for (a) and (b), respectively).

coincides with the hyper-geometric distribution for large volumes of data [5]. Moreover, the time windows are determined by partitioning the incoming document space into chunks of B documents, for a value of B that we can determine experimentally.

Clustering with Time Decay: Beyond developing new features for clustering, we can improve the effectiveness of the event identification process by exploiting the typical temporal behavior of event-related content. Figure 3 shows that the number of events as a function of the duration of their discussions in social media tends to follow a power law distribution with a noisy tail (Figure 3 was derived from the *Upcoming* dataset that we describe in Section 4). Consequently, small-scale events (e.g., street fairs and birthdays) tend to dominate the event space while they are discussed significantly less than large-scale events. As a result, the clustering procedure considers many events that have ended as alive, thus adding noise to the identification process of active events. To alleviate this problem, we could attempt to identify the end of the discussions for an event. However, event-related discussions might not exhibit a clear, or any, ending time. Figure 5 shows two events from the *Upcoming* dataset that last more than one year and are discussed at different points in time at different rates.

To address these challenges, we leverage an interesting observation of event-related discussions in social media: these discussions generally attract massive participation initially, followed by a decline in attention. This decline might be permanent or be followed by a massive restart, as shown in the examples of Figure 5. As an illustration of the latter, consider our Super Bowl example. The announcement of the location of the Super Bowl (i.e., New York and New Jersey) triggers a discussion of the event that is followed by a decay. Then, the announcement of the halftime performer again triggers the discussion, followed by another decay. Based on this observation, we introduce a time decay function to the clustering framework that weighs the consensus function $f(\vec{sim}(d,C), \vec{thr}, \vec{w})$ described in Section 2, to best capture the behavior of discussions over time. Technically, the new similarity score between an incoming document d and a cluster C is computed as: $sim_{decay}(d,C) = f(\vec{sim}(d,C), \vec{thr}, \vec{w}) \cdot e^{-a \cdot \frac{|T_d - T_{e_C}|}{|T_{e_C} - T_{s_C}|}}$, where T_d is the time of creation of document d, T_{e_C} is the maximum time of document creation in the cluster C (i.e., $T_{e_C} = max(\{T_{d'} : d' \in C\})$), T_{s_C} is the minimum time of document creation in the cluster C (i.e., $T_{s_C} = min(\{T_{d'} : d' \in C\})$), and a is the decay constant that we can decide experimentally. Using this time-decay function, the clustering process (a) penalizes clusters that have been inactive for a long time (e.g., as is likely the case for small-scale events) and (b) re-triggers events that have been inactive for some time if the similarity score without the time-decay factor is strong enough (e.g., as is often the case for large-scale events). Both of these properties capture the observations above and adequately improve the clustering procedure, as we show next in our experimental evaluation.

	Baseline	*Parsed Urls*	*BurstyV*	*TimeDec*	*BurstyV+TimeDec*
NMI	0.89703	0.90328	0.92192	0.92933	**0.9414**
B-Cubed	0.80345	0.7897	0.82095	0.81768	**0.83919**

Table 3: Effectiveness of proposed techniques over the *Upcoming* test dataset.

	Baseline	*Whole URL*	*Parsed URL*	*Parsed Query Part*
NMI	0.93517	0.91567	**0.95971**	0.92634
B-Cubed	0.82759	0.81333	**0.87162**	0.82592

Table 4: Effectiveness of URL features for the events of the *Upcoming* dataset associated with at least one URL.

4 Experimental Evaluation

In [3], we reported extensive experiments for the unknown-event identification task, showing that modeling multiple features as weak indicators of event related content, and using them collectively, can produce stronger judgments compared to using them individually. The similarity learning techniques described in Section 2 yielded better performance than the baselines on which we built, including traditional approaches that use text-based similarity. In this section, we report additional experiments to evaluate the contribution of the URL and bursty vocabulary features, as well as the time decay function, which we introduced in Section 3. Specifically, we summarize our experimental settings in Section 4.1 and report the experimental results in Section 4.2.

4.1 Experimental Settings

Data: We use the *Upcoming* dataset presented in [3]. This dataset includes 273,842 multi-featured Flickr photos that correspond to 9,613 real-world events from the Upcoming event catalog[2]. The features associated with each photo that we use as baseline indicators include the title, description, time shot, upload time, and location in longitude-latitude format. (See Section 4.2 for a discussion on other social media sites.)

Methodology: Our evaluation methodology mirrors that of [3]. Specifically, to initiate the clustering procedure we are first required to learn, for each feature, an associated threshold and weight, used in the construction of the consensus function (i.e., $f(\vec{sim}(d, C), \vec{w}, \vec{thr})$ in Section 2). To this end, we first sort the *Upcoming* dataset (descending order of upload time to imitate a real-world streaming scenario) and divide it into three equal parts. Then, we use the earliest two parts to learn the set of weights and thresholds. The features that we use include all the features of the Flickr photos as well as the bursty vocabulary and URL features. Consequently, we construct the final similarity function, using the centroid vector for the cluster representation, and we run our experiments on the last part of the *Upcoming* dataset, on which we report our results. To quantify the quality of our results, we use the well known NMI [10] and $B{-}Cubed$ [1] quality metrics.

Implementation: For the indexing of the textual features we deployed the Oracle Berkeley DB version 6.0.20[3], which assists on the construction of the tf-idf vectors, used to represent textual features. As similarity functions, we use the cosine similarity for textual features, sim_{title}, $sim_{description}$; the Haversine distance [9] for the location feature, $sim_{location}$; and the ℓ_1 norm for the time feature, $sim_{time\text{-}shot}$.

Techniques for Comparison: As a baseline approach, we consider the clustering procedure that models all the individual features (i.e., title, description, time shot, location features) as weak indicators. We evaluate four options: (a) *ParURLs*: Baseline + Parsed URLs, (b) *BurstyV*: Baseline + Bursty Vocabulary, (c) *TimeDec*: Baseline + Time Decay, and (d) *BurstyV + TimeDec*: Baseline + Bursty Vocabulary + Time Decay.

4.2 Experimental Results

As we show in Table 3, the BurstyV + TimeDec technique obtained the highest quality results. This technique combines the bursty vocabulary, which reduces the noise of the textual features, and the time decay function, which fits the typical behavior of events over time. In contrast, the BurstyV technique improves over the baseline, but is problematic in two scenarios. One scenario corresponds to large-scale event discussions that were inactive for a long time (e.g., as in the Figure 5(a) example) but do not necessarily exhibit a similar bursty vocabulary in the next active time window. Another problematic scenario corresponds to discussions that are highly active but tend to change their bursty vocabulary frequently. Both scenarios emphasize the importance of automatically adjusting the B parameter, which regulates the number of documents that have to be appended in a cluster in order to recompute its bursty vocabulary, as described in Section 3. Similarly, the TimeDec technique yields

[2]This web site, now defunct, was available at http://upcoming.org.

[3]http://www.oracle.com/us/products/database/berkeley-db/overview/index.htm

better results than the baseline, but suffers from the noise from unimportant terms, which is handled properly by the BurstyV technique.

In contrast, the ParUrls technique, which uses the *parsed URLs* feature, does not appear to further improve the baseline in this benchmark (the same behavior applies for the *whole URLs* and the *parsed query part of URLs*): Fewer than 11% of the *Upcoming* documents contain URLs, an expected behavior for photographs. In the absence of a URL feature, the corresponding indicator returns a zero similarity score, translating to a failure to detect the true document similarity in most cases. Fortunately, the learning procedure identifies this behavior and assigns a close–to–zero weight to the URL features. Thus the judgments from URL features tend to have no impact on the final decision. If we limit the benchmark to the set of events whose associated documents contain URLs, we observe an improvement over the baseline, as seen in Table 4. This suggests that the proposed URL features may be beneficial for social media sites with a more substantial presence of URLs in their documents (e.g., tweets tend to include URLs frequently, and the presence of URLs could be indicative of event-related content [6]). We now turn to the performance of the alternate URL features (see Table 4). The Parsed Query Part technique, which uses the *parsed query part of URL* feature, performs worse than the Parsed URL technique, which uses the *parsed URL* feature, because most of the URLs in this corpus do not actually have a query part.

5 Conclusions

Social media captures our shared experiences with increasing comprehensiveness. Social media thus serves as an important record of our culture and our society. Moreover, by making new types of information easily accessible on an unprecedented scale, social media has triggered an information revolution perhaps only comparable with the advent of the Web itself in the early 1990s. Still, the methods to retrieve and organize social media content are in their infancy. In our work, we have focused on an important slice of social media content, namely, the content that is associated with real-world events. Specifically, in this article we discussed the event identification task under two substantially different scenarios, known- and unknown-event identification. We showed how we can exploit rich features of the social media documents, as well as revealing temporal patterns of the relevant content, to identify event content effectively. Many open challenges remain for the problems of detection of events in social media and identification of event content, as well as for the presentation and organization of this information for a growing variety of tasks and stakeholders. Beyond events, we hope that our research will help understand, organize, and retrieve social media content around topics, people, places, and more from these new shared records.

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 University reporting requirements, Professor Gravano acknowledges ownership of Google stock as of the writing of this paper.

References

[1] E. Amigó, J. Gonzalo, J. Artiles, and F. Verdejo. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information Retrieval*, 12:461–486, 2009.

[2] H. Becker, D. Iter, M. Naaman, and L. Gravano. Identifying content for planned events across social media sites. In *Proceedings of the 5th ACM International Conference on Web Search and Data Mining (WSDM '12)*, 2012.

[3] H. Becker, M. Naaman, and L. Gravano. Learning similarity metrics for event identification in social media. In *Proceedings of the 3rd ACM International Conference on Web Search and Data Mining (WSDM '10)*, 2010.

[4] H. Becker, M. Naaman, and L. Gravano. Beyond trending topics: Real-world event identification on Twitter. In *Proceedings of the 5th International AAAI Conference on Weblogs and Social Media (ICWSM '11)*, 2011.

[5] G. Fung, J. Yu, P. Yu, and H. Lu. Parameter free bursty events detection in text streams. In *Proceedings of the 31st ACM International Conference on Very Large Databases (VLDB '05)*, 2005.

[6] M. Naaman, H. Becker, and L. Gravano. Hip and Trendy: Characterizing emerging trends on Twitter. *Journal of the American Society for Information Science and Technology*, 62(5):902–918, 2011.

[7] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes Twitter users: Real-time event detection by social sensors. In *Proceedings of the 19th ACM International Conference on World Wide Web (WWW '10)*, 2010.

[8] J. Sankaranarayanan, H. Samet, B. Teitler, M. Lieberman, and J. Sperling. Twitterstand: News in tweets. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (GIS '09)*, 2009.

[9] R.W. Sinnott. Virtues of the Haversine. *Sky and Telescope*, 68(2):159, 1984.

[10] A. Strehl and J. Ghosh. Cluster ensembles - A knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3:583–617, 2002.