Identification and Characterization of Events in Social Media

Hila Becker

Submitted in partial fulfillment of the
requirements for the degree
of Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2011

ABSTRACT

Identification and Characterization of Events in Social Media

Hila Becker

Millions of users share their experiences, thoughts, and interests online, through social media sites (e.g., Twitter, Flickr, YouTube). As a result, these sites host a substantial number of user-contributed documents (e.g., textual messages, photographs, videos) for a wide variety of events (e.g., concerts, political demonstrations, earthquakes). In this dissertation, we present techniques for leveraging the wealth of available social media documents to identify and characterize events of different types and scale. By automatically identifying and characterizing events and their associated user-contributed social media documents, we can ultimately offer substantial improvements in browsing and search quality for event content.

To understand the types of events that exist in social media, we first characterize a large set of events using their associated social media documents. Specifically, we develop a taxonomy of events in social media, identify important dimensions along which they can be categorized, and determine the key distinguishing features that can be derived from their associated documents. We quantitatively examine the computed features for different categories of events, and establish that significant differences can be detected across categories. Importantly, we observe differences between events and other non-event content that exists in social media. We use these observations to inform our event identification techniques.

To identify events in social media, we follow two possible scenarios. In one scenario, we do not have any information about the events that are reflected in the data. In this scenario, we use an online clustering framework to identify these unknown events and their associated social media documents. To distinguish between event and non-event content, we develop event classification techniques that rely on a rich family of aggregate cluster statistics, including temporal, social, topical, and platform-centric characteristics. In addition, to

tailor the clustering framework to the social media domain, we develop similarity metric learning techniques for social media documents, exploiting the variety of document context features, both textual and non-textual.

In our alternative event identification scenario, the events of interest are known, through user-contributed event aggregation platforms (e.g., Last.fm events, EventBrite, Facebook events). In this scenario, we can identify social media documents for the known events by exploiting known event features, such as the event title, venue, and time. While this event information is generally helpful and easy to collect, it is often noisy and ambiguous. To address this challenge, we develop query formulation strategies for retrieving event content on different social media sites. Specifically, we propose a two-step query formulation approach, with a first step that uses highly specific queries aimed at achieving high-precision results, and a second step that builds on these high-precision results, using term extraction and frequency analysis, with the goal of improving recall. Importantly, we demonstrate how event-related documents from one social media site can be used to enhance the identification of documents for the event on another social media site, thus contributing to the diversity of information that we identify.

The number of social media documents that our techniques identify for each event is potentially large. To avoid overwhelming users with unmanageable volumes of event information, we design techniques for selecting a subset of documents from the total number of documents that we identify for each event. Specifically, we aim to select high-quality, relevant documents that reflect useful event information. For this content selection task, we experiment with several centrality-based techniques that consider the similarity of each event-related document to the central theme of its associated event and to other social media documents that correspond to the same event. We then evaluate both the relative and overall user satisfaction with the selected social media documents for each event.

The existing tools to find and organize social media event content are extremely limited. This dissertation presents robust ways to organize and filter this noisy but powerful event information. With our event identification, characterization, and content selection techniques, we provide new opportunities for exploring and interacting with a diverse set of social media documents that reflect timely and revealing event content. Overall, the work

presented in this dissertation provides an essential methodology for organizing social media documents that reflect event information, towards improved browsing and search for social media event data.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

The ease of publishing content on social media sites brings to the Web an ever increasing amount of content captured during—and associated with—various types of events. Event content shared on social media sites such as Flickr, YouTube, Twitter and others varies widely, ranging from planned, known occurrences such as a concert or a parade, to spontaneous, unplanned incidents such as an earthquake or death of a celebrity. By automatically identifying and characterizing these events and their associated user-contributed social media documents (e.g., Flickr photographs, YouTube videos, Twitter messages), we can enable rich search and presentation of all event content. In this dissertation we present approaches for leveraging the wealth of social media documents available on the Web for event identification and characterization.

As motivation for identifying events in social media, consider a person who is thinking of attending the opening gala of "Celebrate Brooklyn!," an annual arts festival that takes place in Brooklyn, New York every summer. Prior to purchasing a ticket, this person could search the Web for relevant information that would aid in making an informed decision. Unfortunately, Web search results are far from revealing for this relatively minor event: the event's website contains basic details about the event (e.g., time, location), and traditional news coverage is low, with some articles providing the list of performers, and others discussing various related topics. Overall, these Web search results do not convey what this person should expect to experience at this event. In contrast, user-contributed content may provide a better representation of prior instances of the event from an attendee's perspective, or timely announcements relevant to the event (e.g., "tickets for the Celebrate Brooklyn! opening gala are sold out"). A user-centric perspective, as well as coverage of a wide span of events of various types and scale, make social media sites a valuable source of event information.

Our problem is most similar to the event detection and tracking task [APL98; KA04; YPC98], whose objective is to identify events in a continuous stream of news documents (e.g., newswire, radio broadcast transcripts). However, our problem exhibits some fundamental differences from traditional event detection that originate from the focus on social media sources. Specifically, most of the work on event detection focuses on identifying, clustering, and searching over a corpus of event documents from broadcast news. These news articles adhere to certain grammatical, syntactical, and stylistic standards that are appropriate for their venue of publication. Therefore, most state-of-the-art event detection approaches leverage natural language processing tools such as named-entity extraction and part-of-speech tagging to enhance the document representation [HGM00; MAMS04; ZZW07]. In contrast, social media documents contain little textual narrative, usually in the form of a short description, title, or keyword tags. Importantly, this text is often noisy, which renders traditional event detection techniques undesirable for social media documents.

We are interested in identifying events, and their associated social media documents, with two goals in mind: timeliness and breadth. To address our timeliness goal, we leverage information from social media sites that enable the exchange of short textual messages (e.g., Twitter, Facebook, Google+), as these messages often contain revealing and timely event information. To address our breadth goal, we then identify event content across all types of social media sites (e.g., photo-sharing, video-sharing, and social networking sites), as such cross-site event content is useful for augmenting and enhancing event content identified on any social media site individually.

To identify events in a timely manner, as they occur or as soon as their social media documents are produced, we follow two possible scenarios. In one scenario, we do not have any advanced knowledge of the events that may be present in a stream of social media documents. In the other scenario, we assume that we have some advanced knowledge of the event, in the form of associated context features (e.g., title, time, location). These unknown vs. known event identification scenarios guide our identification techniques in this dissertation. For the unknown scenario, we focus on a specific class of events that we refer to as *trending events* (Chapter 2), due to their unique temporal characteristics. For the known scenario, we focus on a specific class of events that we refer to as *planned events* (Chapter 2), for which prior information is posted on event aggregation sites (e.g., Last.fm events, EventBrite). Note that these classes of events are not mutually exclusive, but rather indicate the types of events we focus on in each identification scenario (see Chapter 2).

In the unknown identification scenario, we identify trending events by leveraging information from social media sites that enable the exchange of short textual messages (e.g., Twitter, Facebook, Google+). These short textual messages can typically reflect unknown events as they happen, making them particularly useful for timely event identification. As a unique advantage, for unplanned events (e.g., the Iran election protests, earthquakes), users of these social media sites sometimes spread news prior to the traditional news media [KLPM10; SOM10]. However, identifying events on such sites is a challenging task, as shared messages are brief and often exhibit low quality (e.g., with typos and ungrammatical sentences).

As we discussed, in the unknown identification scenario we specifically focus on identifying trending events, where the frequency of documents associated with such an event in a stream of social media documents exhibits an unusual, increasing trend during the time period associated with the event. To inform the trending event identification process, it is useful to understand the range and characteristics of all types of trends (Chapter 2) that exist in social media, including (but not limited to) those that represent trending events. With a thorough understanding of the different types of trends and, consequently, trending events that exist in social media, we can help our event identification approaches distinguish between trending events and trends that do not reflect event information, and also among specific trending event types. Characterizing different types of trends along various dimensions can be useful for automatically identifying and differentiating among them. To this end, we collect, analyze, and characterize content associated with trends on Twitter, given this site's desirable properties (e.g., presence of real-time event content, diversity of event content) for identifying events in a timely manner. A strong grasp of trending event characteristics can help provide critical information for applications such as event browsing and search, which could be enhanced with trending event information from social media.

For our analysis of trending events, we specifically focus on one social media site, namely, Twitter, due to its transient, large-scale publicly available content. In particular, we collect a set of textual terms and phrases that exhibit trending behavior on Twitter, characterized by an unusual increase in message frequency during a particular time period in a Twitter message stream. While some of these trends (Chapter 2) might refer to events, others might include non-event information. To organize and understand this content, we define a taxonomy of trend types, which includes trending events, using Twitter messages associated with each trend. Unlike related efforts in this area, which focused on characterizing or analyzing content from individual events on Twitter [Yb10; NGS+09], or characterizing aggregate trend characteristics for manually identified terms [SOM10; DNKS10], the taxonomy we define in this study is based on a large set of automatically identified trends.

Given this taxonomy, we demonstrate how different types of trends (and trending events) can be distinguished from one another according to various descriptive characteristics. Specifically, we identify important dimensions according to which trends can be categorized (e.g., content, interaction), as well as the key distinguishing features of trends that can be derived from their associated messages (e.g., hashtag usage, forwarding behavior). We quantitatively examine the computed features for different categories of trends, and establish that significant differences can be detected across categories. As a key contribution, we identify and distinguish between endogenous, platform-centric trends and trending events that reflect real-world occurrences (Chapter 3).

As we learn from our study, Twitter messages generally reflect useful trending event information for a variety of trending events of different types and scale. To collect trending events for our study, we use simple techniques [NBG11] that can surface terms and phrases that might not necessarily be associated with any trending event. Additionally, these terms and phrases are often insufficient as stand-alone identifiers for each event (e.g., "sparklehorse" referring to the band's lead singer's suicide) and, therefore, provide a noisy indicator of event content when found in Twitter messages. To address these two issues, we develop more advanced techniques to create a robust real-time framework for trending event identification on Twitter.

Identifying trending events on Twitter is a challenging task since much of the shared content is not related to any particular trending event [NBL10], but rather consists of mundane conversations (e.g., "good morning" and "thank you" messages) or Twitter-centric discussions (e.g. using the #whenimolder hashtag, describing what Twitter users would like to do when they get older). However, despite the ubiquity of non-event content, informative event messages also abound (e.g., "the health care reform bill was passed"). Therefore, our techniques must distinguish between event and non-event content on Twitter, to identify trending events and their associated messages.

Several related research efforts have focused on identifying trending events in social media [CR09; POL10; SOM10; SST$^+$09]. Chen and Roy [CR09] presented techniques for *retrospective* identification of trending events on Flickr, finding patterns in the data after an event has occurred rather than identifying the event as soon as its associated content is posted. Recent work on Twitter has started to process data as a stream, as it is produced, but has mainly focused on identifying events of a particular type (e.g., news events [SST$^+$09], earthquakes [SOM10]). Other work aimed to identify the first Twitter message associated with a trending event [POL10]. Our focus is on *online identification* of trending events, and their associated social media documents, for a variety of events, regardless of their type.

We develop techniques for identifying trending events in real-time using an online clustering framework. This framework provides a scalable, incremental solution, to handle the high volume and changing nature of social media documents. We use this clustering framework to group together topically similar social media documents. To identify all event clusters, we compute a variety of revealing features using collective characteristics of all the documents in each cluster. These include social interaction characteristics, temporal characteristics, content coherence characteristics, and endogenous [NBG11] characteristics. Since the clusters constantly evolve over time, we periodically update the characteristics for each cluster and compute characteristics of newly formed clusters. We train a classifier over these characteristic features and use it to determine which clusters contain event content at any point in the stream (Chapter 4).

In addition to Twitter, which we focus on for timely identification given the reasons discussed above, we can use our clustering framework with other types of social media documents (e.g, Flickr photos, YouTube videos). While these social media documents may not always be instantaneously shared like the messages on Twitter, they often provide useful and rich event content, to complement and enhance identified event content on Twitter. Unlike Twitter, however, where messages are textual and brief, social media documents on sites such as Flickr include rich context features such as user-provided annotations (e.g., title, tags, description) and automatically generated information (e.g., content creation time, geo-coordinates). Using this rich context, which includes both textual and non-textual features, we can define appropriate document similarity metrics to enable online clustering of media to events. As a key contribution, we explore a variety of techniques for learning multi-feature similarity metrics for social media documents in a principled manner (Chapter 5).

In the second scenario for identifying events in a timely manner, we focus on the challenge of automatically identifying social media documents related to known, planned events (e.g., concerts, parades, conferences) from user-contributed event aggregation platforms (e.g., Last.fm events, EventBrite, Facebook events). These event aggregation platforms provide revealing, structured information (e.g.,title, description, time, location) for planned events, but this information is often noisy or incomplete. Our goal is to automatically identify social media documents for these planned events on *multiple* social media sites, with focus on achieving both high precision and high recall. For this, we define a two-step query formulation process that automatically constructs queries for each planned event given its associated context features.

As an example of a planned event, consider the opening night gala of the "Celebrate Brooklyn!" festival. To formulate queries for this event, our techniques leverage explicitly provided event features such as title (e.g., "Celebrate Brooklyn! Opening Gala"), description (e.g., "Singer/songwriter Andrew Bird will open the 2011 Celebrate Brooklyn! season"), time/date (e.g., June 10, 2011), location (e.g., Brooklyn, NY), and venue (e.g., "Prospect Park"). While these provided event features are generally informative, they also present many challenges for our techniques. Specifically, these features may be inaccurate (e.g., time/date using the wrong time zone), incomplete (e.g., missing state in the location feature), missing altogether, overly specific (e.g., "Celebrate Brooklyn! Opening Night Gala & Concert with Andrew Bird" for the title feature) or too broad (e.g., "Opening Night Concert" for the title feature). Therefore, we develop a variety of query formulation strategies, designed to overcome the various challenges that this data presents.

Our two-step query formulation technique starts with a precision-oriented step, which combines the planned event features into several queries aimed at retrieving *high-precision* results. These precision-oriented queries complement each other and are used to retrieve social media documents collectively. The second, recall-oriented, step uses these high-precision results along with text processing techniques such as term extraction and frequency analysis to build additional queries, aimed at improving the (generally low) recall of the precision-oriented step. Unlike prior work [SOM10; Yb10], our strategies are fully automatic and do not require manually selected terms or phrases to retrieve documents for each event. Additionally, in contrast to related efforts by Benson et al. [BHB11] we do not impose any assumptions on the type of event (i.e., concerts) or desired information in the corresponding documents (i.e., artist and venue names).

Importantly, prior research on identifying events in social media focused on tailoring approaches to one specific social media site. To address our breadth goal, we focus on identifying planned events across all types of social media sites (e.g., photo-sharing, video-sharing, and social networking sites). Our query formulation strategies can be applied to each social media site individually or to all social media sites simultaneously. We propose a couple of cross-site query formulation and retrieval techniques, and demonstrate how event content identified on one social media site can be used to improve the identification process on another social media site (Chapter 6).

Overall, we show that social media sites contain substantial, useful information about events. With the techniques we develop in this dissertation, we can effectively identify different types events and their associated social media documents across various social media sites. Regardless of the technique we use, the type of event, or the social media site, any single event might have hundreds or thousands of associated social media documents. While some of these associated documents might contain interesting and useful information (e.g., event time, location, participants, opinions), others might provide little value (e.g., using heavy slang, incomprehensible language without accompanying media) to people interested in learning about an event. Techniques for effective selection of quality event content may then help improve applications such as event browsing and search. Therefore, we propose and evaluate a variety of centrality-focused techniques for selecting a subset the social media documents associated with each of our identified events.

Selecting a representative subset of document for each event is a challenging task. As one challenge, seemingly related documents with good textual quality might not be truly relevant to the event (e.g., "I am going to celebrate in brooklyn tomorrow" for the "Celebrate Brooklyn!" opening gala). As another challenge, relevant, high-quality documents might not be useful (e.g., "I can't stop thinking about the Celebrate Brooklyn! opening gala") as they do not provide much information about the event in question. Therefore, we focus on selecting a subset of documents for each event that exhibit high textual quality, high relevance to the event, and clear usefulness to a user looking for information about the event.

While related efforts focused on summarizing or otherwise presenting social media documents related to events [DNKS10; NGS+09; SKC10], our goal in this work is to select a subset of the documents to be presented in their unaltered form, complete with any associated digital media, tags, URLs, and other contextual information (e.g., time, author name). Presenting event documents along with all of their context features provides a rich, multi-dimensional view of an event. Towards this goal, we experiment with centrality-based techniques to select event documents based on their similarity to a centroid representation of an event or to other event documents (Chapter 7). Although there are other useful document features that could be used for this content selection task, they generally produce poor results when used in isolation. Instead, these features could be incorporated with our centrality based approaches in a disciplined way (e.g., using a trained ranking function), a task that we reserve for future work.

In summary, the contributions of this dissertation are as follows:

- A qualitative study of trends and trending events in social media, yielding in a taxonomy of trends and events, and a complementary quantitative study, examining the differences between trend and trending event types along various descriptive characteristics (Chapter 3)

- An online clustering framework for unknown identification of trending events, and their associated social media documents, along with classification techniques for separating event and non-event content (Chapter 4)
- Similarity metric learning approaches for unknown identification of trending events, and their associated social media documents, and approaches for leveraging social links to improve event clustering results (Chapter 5)
- Query formulation techniques for identifying social media documents for planned events from multiple social media sites, showing that we can effectively identify event documents on each social media site individually, and on multiple social media sites simultaneously, by leveraging identified event content on one site to enhance the identification process on another site (Chapter 6)
- Content selection techniques for choosing a subset of social media documents associated with each identified event, based on the documents' quality, relevance, and usefulness to the event (Chapter 7)

In Chapter 2 we discuss several alternative definitions of events in the literature and provide the event definitions that we use in this dissertation. We describe additional related work in Chapter 8, and then present our conclusions and discuss directions for future work in Chapter 9.

As the amount of social media content grows, research will have to identify robust ways to organize and filter that content. In this dissertation we aim to provide scalable techniques for organizing social media documents associated with events. With our event identification, characterization, and content selection techniques, we provide new opportunities for exploring and interacting with social media event data.

Chapter 2

Event Definition and Characterization

The definition and characterization of “event” has received substantial attention across academic fields, from philosophy [Eve02] to cognitive psychology [ZT01]. An event is often defined as an abstract concept [Eve02], or with respect to its manifestation in a specific domain (e.g., time series, textual news, social media). Even within a specific domain, researchers often disagree on what precisely constitutes an event [GSFW94], or agree on a definition that is admittedly problematic and does not cover all possible cases [Mak03]. Still, it is important to formally and precisely define the concept of an event to convey the various phenomena we study, identify, and characterize in this dissertation.

In this chapter, we survey the alternative definitions of events in the literature over a variety of domains and connect these definitions to our task of identifying and characterizing events in social media (Section 2.1). In addition to events, we present several definitions of related concepts (e.g., activity, topic, trend), which are used in the literature to generalize or extend the definition of event (Section 2.2). These concepts, and particularly “trend,” are useful for our study of events in social media, presented in Chapter 3. We draw on all of these different definitions to define an event in the context of our work. As we will show, an event is a complex concept and defining it, or its embodiment in social media, is a difficult task [GSFW94; Mak03]. Therefore, instead of providing another definition of an event, we define specific types of events in social media, on which we focus in this dissertation (Section 2.3).

2.1 Events in the Literature

We present and discuss various efforts to define events in the context of three tasks: topic detection and tracking in news documents (Section 2.1.1), event extraction from free text (Section 2.1.2), and multimedia event detection (Section 2.1.3).

2.1.1 Topic Detection and Tracking

Event detection in broadcast news was notably studied as part of a broad research effort known as Topic Detection and Tracking (TDT) [All02]. TDT research focused on a variety of tasks concerning event-based organization of textual news document streams (e.g., newswire, news broadcast transcripts). These tasks include new event detection, focused on identifying the first document in a text document stream that corresponds to a previously unknown event, and retrospective event detection, aimed at grouping documents according to their event content [ACD+98]. Importantly, to evaluate these different TDT tasks, the Linguistics Data Consortium (LDC) prepared annotated corpora based on a set of guidelines for what constitutes an event (as well as other concepts such as activity, topic, and trend, described in Section 2.2).

The definition of event for TDT research has evolved throughout the years, with several researchers proposing alternatives to address problems in earlier versions of the definition. Initially, an event was defined as "some unique thing that happens at some point in time" [ACD+98]. This definition touches on an important aspect of an event, namely, its necessary association with a defined time period. Later, a notion of location was introduced, defining an event as "something that happens at some specific time and place" [YCB+99]. Although most news events indeed happen at a specific location, some events may be global or virtual (e.g., World Peace Day, a virtual trade show) so their location is not well defined.

As a significant drawback, the scope of an event according to these definitions is vague and may be interpreted in a variety of ways. For instance, under this definition, the earthquake in Japan that took place on May 21, 2011 is an event. However, textual news documents that report on this event might mention the tsunami alert caused by the earthquake, the response of the rescue crews, and the damage to the Fukashima nuclear power plant, which are all acceptable events according to the definition. Since the scope of an event according to this definition is open, it is unclear if the events in our examples should be considered as separate events or one collective event.

To address the ambiguity of the initial TDT definitions, a new definition was proposed [All02], stating that an event is "a specific thing that happens at a specific time and place along with all necessary preconditions and unavoidable consequences." This amended definition considers our previous examples of the Japan earthquake and subsequent reactions to be a single event. While this definition makes some clarifications regarding event boundaries, it introduces other questions, namely, what can be considered necessary preconditions and unavoidable consequences for an event [Mak03]. For certain events (e.g., the 2008 Mumbai terror attacks), some of the necessary preconditions and unavoidable consequences are unknown or subject to debate.

Overall, the TDT-inspired definitions of an event introduce some useful ideas (e.g., an event's association with a specific time period), but they are also somewhat ambiguous and do not cover all possible types of events. In Section 2.2 we discuss some of the additional concepts introduced by the TDT effort to facilitate event detection in textual news streams.

2.1.2 Event Extraction

Event extraction is a task that involves identifying instances of specific types of events, and their associated attributes, in free text [Gri10]. Extracting such events from text has been the focus of numerous studies as part of a National Institute of Standards and Technology (NIST) initiative for Automatic Content Extraction (ACE)[1]. The ACE event extraction task explicitly defines a set of event types (e.g., conflict) and subtypes (e.g., attack) to be extracted from various text sources (e.g., newswire, Blogs, conversation transcripts), using a set of predefined templates that include event attributes (e.g., attacker, target). A template of the "attack" event subtype applied to the sentence "Yesterday, a number of

[1] http://www.nist.gov/speech/tests/ace/

Attribute	Description	Example
Attacker	The attacking/instigating agent	demonstrators
Target	The target of the attack (including unintended targets)	Israeli soldiers
Instrument	The instrument used in the attack	stones and empty bottles
Time	When the attack takes place	yesterday
Place	Where the attack takes place	a Jewish holy site at the town's entrance

Table 2.1: Attack event template and sample extracted attributes.

demonstrators threw stones and empty bottles at Israeli soldiers positioned near a Jewish holy site at the town's entrance" can be found in Table 2.1.

The guidelines for the ACE event extraction task[2] provide a few generic and unexplained definitions for an event. According to these guidelines, an event is "a specific occurrence involving participants," and "something that happens." These definitions by themselves do not explain how to address events with ambiguous semantic scopes (e.g., the May 25, 2011 Japan earthquake) and allow for undefined temporal scopes (e.g., "Thanksgiving Day Parade"). However, instead of defining all possible events abstractly, the ACE event extraction task defines specific events according to their expression in free text. This operational definition outlines the spans of text that can be used to identify an event, namely, the event trigger and extent. The event trigger is the word that most clearly expresses the event's occurrence (e.g., "threw" in our example sentence), and the extent, which indicates the scope of the event, is defined as the sentence in which the event trigger is mentioned.

In addition to the event itself, as identified via a keyword trigger (e.g., "attack," "threw," "bomb"), the ACE event extraction task requires that a set of participants be identified for each event. This set of participants varies by event type (e.g., attacker for an "attack" event, defendant for a "trial-hearing" event). The requirement to have a set of participants indicates an implicit assumption that events should have one or more participants. However, not all events have a clearly defined set of participants (e.g., an earthquake, a solar

[2] http://projects.ldc.upenn.edu/ace/docs/English-Events-Guidelines_v5.4.3.pdf

eclipse). Indeed, the assumption of participants might be appropriate to this particular event extraction task since it does not involve identifying all possible events, but rather a specific subset of events that follow predefined templates. Other interesting event attributes used in this task include time and place. Even though these attributes were not mentioned in the event definition, they are both present in the predefined templates of almost all event types and subtypes (with the exception of a "phone-write" event, which occurs when two or more people engage in a discussion remotely and hence does not require a location attribute).

Contrary to the TDT definitions that are very broad but ambiguous, the operational definitions used for the ACE event extraction tasks are particular but restrictive. Focusing on a restricted class of events is often useful to eliminate ambiguity and enable precise annotations for evaluation purposes. At the same time, this type of definition only applies to supervised event detection tasks, where the classes of events that should be detected are known a priori.

2.1.3 Multimedia Event Detection

Research on detecting events in multimedia content (e.g., videos, audio clips) has received considerable attention over the past several years. In a notable effort, work on Multimedia Event Detection (MED)[3] aims to detect evidence of events in multimedia content using audio and video streams of multimedia clips. Unlike the previously described event detection tasks, the MED task does not allow the use of human-annotated textual context features (e.g., title, tags) that often accompany such clips. As part of the TREC Video Retrieval Evaluation, another large-scale NIST research effort, the specific goal of MED is to develop event detection techniques to enable quick and accurate search of user-defined events in multimedia collections. An event according to the MED 2011 evaluation plan[4] is "a complex activity occurring at a specific place and time," involving "people interacting with other people and/or objects." Additionally, an event "consists of a number of human actions, processes, and activities that are loosely or tightly organized and that have a significant

[3] http://www.nist.gov/itl/iad/mig/med11.cfm

[4] http://www.nist.gov/itl/iad/mig/upload/MED11-EvalPlan-V03-20110801a.pdf

Event Name	Making a cake
Definition	One or more people make a cake.
Evidential Description	
Scene	indoors, typically a kitchen in a home, restaurant or other setting
Objects/People	ingredients (like eggs, flour, cake mix), bowls, spoons, mixers, cake pans, ovens, potholders, candles
Activities	selecting ingredients, combining ingredients, pouring batter into pan, putting cake into oven, setting timer, removing cake from oven, testing cake for doneness, decorating cake

Table 2.2: Example of an “event kit” for the MED task.

temporal and semantic relationships to overarching activity” and “is directly observable.”

Once again, the association with a specific place, time, and participants is explicitly mentioned in the event definition. As we discussed above, any definition that mentions a specific location or requires a set of participants is prone to ambiguity as it does not fit all possible types of events (e.g., virtual events, natural disasters). Additionally, the notions of “activity” and “processes” are broad and remain undefined. This definition implies that an event must center around human activities, which is not always the case for events in general (e.g., an earthquake might be the result of seismic activity, but it does not by itself consist of a human action or activity). We discuss some definitions for “activity” and the connection between activities and events in the next section.

Similar to the ACE event extraction task, the MED event detection task is supervised, which means that the set of events that it aims to identify (e.g., making a cake, assembling a shelter) is predefined. Each event has an associated predefined “event kit,” consisting of a name, definition, and description for each event (see Table 2.2). In addition, each event kit includes a set of illustrative video examples (Figure 2.1), each containing an instance of the event. These examples help form the definition of each event but, according to the MED guidelines, they “do not demonstrate all possible variability or potential realizations.”

Figure 2.1: Examples for the “Making a cake” event.

Interestingly, a multimedia event such as “making a cake” is very different in nature than an ACE- or TDT-style event. Although the definition of events for the MED task states that an event should have a specific time and place, making a cake can happen at different times and different places, making it more similar to an ACE event *type* than a TDT or ACE event. If making a cake was considered newsworthy, a similar event for the TDT (or ACE) task might instead be described as “Alton Brown making a cake in the French Culinary Institute in New York City on August 20, 2011 at 1pm.”

2.2 Related Concepts: Topics, Trends, and Activities

The discussion of events in the literature often includes a variety of related concepts that are used to generalize or build upon the notion of an event. We present an overview of three such related concepts: activity, topic, and trend.

Activity: The notion of an activity is introduced in the TDT guidelines[5] to clarify and refine the definition of a topic. Specifically, an activity is defined as “a connected set of actions that have a common focus or purpose.” For our Japan earthquake example, the earthquake is an event, and the earthquake-related disaster relief efforts are activities. In our work, we make no distinction between events and activities as defined above, and consider all such activities as valid events. Often, event-related activities fall within the scope of the event (e.g., the earthquake and subsequent relief efforts would be considered part of the same event) and, therefore, considered part of the event itself. As we discussed (Section

[5] http://projects.ldc.upenn.edu/TDT2/Guide/label-instr.html

2.1), defining the scope of the event is conceptually difficult and, therefore, researchers often resort to using signals such as temporal proximity of topically similar events and activities to operationally define the event scope.

Topic: The initial definition of topic for the TDT task was the same as the definition of event (see Section 2.1.1). However, it was later changed to "an event or activity, along with all directly related events or activities." Intuitively, a topic in TDT is a set of news documents that are strongly connected by some real-world event (e.g., the earthquake in Japan and all other stories triggered by it, including tsunami alerts and the nuclear meltdown) [All02]. As with the event definition, it exhibits scope problems, since the word "related" is left open to interpretation. To resolve this issue, the LDC modified the TDT annotation guidelines to include "rules of interpretation," which outline the types of events that may be considered related to other types of events. Unfortunately, Makkonen [Mak09] observed that while an event is defined as something that happens (Section 2.1.1), a topic, according to this definition, is a human judgment and, therefore, an event cannot be a topic, which contradicts the definition.

Yang et al. [YCB+99] describe the difference between an event and a topic "in the conventional sense" by stating that events are instances of topics. For example, the Thanksgiving Day Parade is an event, and is an instance of the general topic of parades. This observation contradicts the definition of events in the ACE event extraction task (Section 2.1.2) where, for example, the "building a shelter" event would be considered a topic according to Yang et al. [YCB+99], and specific incidents of shelter-building would be considered events under this topic. The idea that events are instances of a topic was also reinforced by Filatova et al. [FHM06], who used the concept of a "domain" to describe a set of events of a particular type. As another insight about the relationship between topics and events, Yang et al. observed that a time gap between bursts of documents on a particular topic often indicates that each burst corresponds to a different event. In other words, the temporal scope of an event is smaller than that of the topic to which it belongs.

Trend: The concept of a "trend" and its relation to events is of particular interest and importance to the work described in this dissertation. Kontostathis et al. [KGP+04] define an emerging trend[6] as “a topic area that is growing in interest and utility over time.” While seemingly intuitive, this definition does not explain what the term “topic” encompasses, and the necessary amount or speed of the topic’s growth over time. As we discussed, the notion of a topic can be interpreted in different ways and its scope varies in different contexts. Additionally, the amount or speed of growth necessary for a topic be designated a trend varies according to the data source (e.g., trending topics on Twitter are generally short-lived, often changing on an hourly basis, whereas trending topics in Computer Science research publications rise and fall less rapidly due to the difference in temporal dynamics and content of the data in these two sources).

Trend detection tasks over textual data collection generally aim to identify topic areas that are previously unseen or are *rapidly* growing in importance within the corpus [KGP+04]. This idea of rapid growth, sometimes referred to as a “burst,” is prevalent in the literature [Kle03; HCL07; WZHS07] and often used to describe trending behavior in text document streams. In this context, Kleinberg [Kle03] observed that the emergence of a topic in a text document stream is signaled by a sharp rise in frequency of the features associated with the topic. We use this observation to guide our study of trends and trending events in social media (Chapter 3) and our definition of trending events in the next section.

2.3 Events in Social Media

If a tree falls in the forest and nobody tweets (or posts any information) about it, is it an event? According to some event definitions (Section 2.1) it might be considered an event, but it is *not* an event in social media since it does not have a corresponding realization in social media documents. Instead of providing an abstract, ambiguous, or arguable definition of an event, for the purpose of this work we focus on specific types of events, which we define precisely with respect to a stream of social media documents.

In Chapter 1, we described two alternative event identification scenarios that we consider, namely, the *unknown* scenario, where events are identified in an unsupervised manner,

[6] Emerging trends are sometimes also referred to as “trending topics,” or “emerging topics.” All trends discussed in this dissertation are emerging trends, so we simply refer to them as “trends,” for brevity.

and the *known* scenario, where some basic event features (e.g., title, time, location) are available to our techniques. In the unknown scenario, since we have no information about the events in the stream, we must rely on other signals that could indicate the presence of event content. For this reason, in this scenario we focus on a class of events that we refer to as *trending events*, which are events that exhibit bursty temporal patterns [Kle03]. Formally, we provide the following definitions:

Definition 1 *A document stream is a time-ordered sequence of documents; each document is represented as a set of features, or terms.*

Definition 2 *A trending time period for a feature over a document stream is a time period where the document frequency of the feature in the document stream is substantially higher than expected.*

Definition 3 *A trending event is a real-world occurrence e with (1) an associated time period T_e, (2) a stream of documents D_e about the occurrence and published during time T_e, and (3) one or more features that describe the occurrence and for which T_e is a trending time period over document stream D_e.*

The document stream in our definition refers to a stream of social media documents (e.g., a Flickr photo, a YouTube video, a Twitter message), which can be represented using a variety of associated context features (e.g., title, tags). These document representations always include textual terms, among other features that depend on the document's source (see Chapter 5). Within the document stream, social media documents are always processed chronologically, as they are posted. Each feature has an expected document frequency value based on its historically observed document frequency in the stream. A deviation from this expected value (a "sharp rise" [Kle03] or "burst" [WZHS07]) in a specific time period indicates a trending time period for the feature in the document stream. This deviation may be defined in different ways [Kle03], and we experiment with alternative definitions in Chapters 3 and 4. Finally, determining whether such trending feature describes the real-world occurrence in Definition 3 is based on a human judgment.

In the known event identification scenario, we know some basic features of the events that we identify. Therefore, for this scenario, we focus on a class of events that we refer to as *planned events*, for which a set of associated context features is available. Regardless of the source of these context features (e.g., user-contributed event aggregation sites, shared calendars), we require that, at a minimum, these features consist of a title and time. Formally, we define planned events as follows:

Definition 4 *A planned event is a real-world occurrence e with an associated time period T_e and a corresponding published event record consisting of, at least, the following two event context features: (1) title, indicating the subject of event e as determined by a human being, and (2) time, indicating the time at which e is planned to occur.*

In our work, we make a couple of assumptions about the relationship between events and social media documents. First, we consider documents that are significantly related to an event as being associated with the event, even if the documents were produced before or after the event. In other words, the time period T_e associated with event e can start before and end after the actual start and end date of event e. For instance, in our "Celebrate Brooklyn!" example, a photograph of a participant in front of the box office represents the author's experience in the context of the event and will, therefore, be associated with the event for our purpose. For planned events, the published time T'_e corresponds to the time at which e is planned to occur, and therefore T_e spans T'_e. Second, we assume that each social media document corresponds to exactly one event. However, our approaches can generally be extended to handle cases where a single social media document contains information pertaining to several events.

It is important to note that planned events and trending events are not mutually exclusive event types. Rather, they represent characteristics of an event along two orthogonal dimensions. Therefore, an event can be either trending or non-trending, and at the same time be either planned or unplanned. To illustrate this point, Figure 2.3 presents examples of events that represent the various combinations of these dimensions. Unplanned events such as an earthquake are also likely to be trending events due to the sudden onset of information about such a relatively rare phenomenon. Planned events such as the Canadian Cheese Festival are sometimes non-trending, as the "buzz" generated by such special-interest

	Planned	Unplanned
Trending	American Idol season premier	Japan earthquake
Non-Trending	Canadian Cheese Festival	Minor car accident

Table 2.3: Examples of different types of events.

events on social media is often low[7]. However, there is an overlap between trending and planned events, for events such as the American Idol premier, which generate substantially high volume of information within a short time period, relative to the expected amount of discussion about this event over that time period. Unfortunately, we cannot identify unplanned non-trending events (e.g., a minor car accident that may have generated very few reactions on social media), due to the lack of available context and signal.

According to our event definitions, events in social media 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 World Wide Web conference. Non-event content, of course, is prominent on social media sites where people share various types of content such as personal updates, random thoughts and musings, opinions, and information [NBL10]. In the next chapters, we use our event definitions (i.e., trending and planned events) in the context of different social media event identification and characterization scenarios (i.e., unknown and known scenarios), to distinguish event content from the vast amount of non-event content posted on social media sites. Specifically, for the unknown identification scenario in Chapters 3, 4, and 5, we focus on trending events, as defined in Definition 3. Then, for the known identification scenario in Chapter 6, we focus on planned events as defined in Definition 4. Finally, in Chapter 7, we refer to any event that falls under the planned or trending definition (or both), since this encompasses the spectrum of events that we can identify in social media under both the known and unknown scenarios.

[7] This is not a general observation about special-interest events but rather an illustrative example of an event that could be simultaneously planned and non-trending.

Chapter 3

Characterization of Trending Events in Social Media

Social media sites such as Facebook, Twitter, Google+, and others allow users to publicly and rapidly share streams of lightweight content artifacts, from short status messages to links, pictures, and videos. These sites have already shown considerable impact on the way in which people share and consume information about events, as evidenced during major global events such as the 2009 Iran election or the reaction to the 2010 earthquake in Haiti [KLPM10], and in response to local events and emergencies [SPS08; SPHV10]. The often-public shared content on these sites ranges from personal status updates to opinions and information [NBL10]. In aggregate, however, the postings by hundreds of millions of users of Facebook, Twitter, and other sites expose global interests, happenings, and attitudes in almost real time [KLPM10]. Importantly, this content often reflects events as they happen [KLPM10; SOM10; SST^{+}09], making such sites particularly useful for addressing our timeliness goal for event identification (see Chapter 1).

The interests and happenings as reflected in social media change rapidly. This strong temporal nature of shared information allows for the detection of significant trends in the data stream, which often correspond to trending events that we wish to identify. Such trending events may reflect a varied set of real-world occurrences, including local events (e.g., a baseball game or “fire on 34th street”), global news events (e.g., Michael Jackson's death), and televised events (e.g., the final episode of ABC's popular series "Lost"). Unfortunately, not all trends reflected in social media data correspond to events. Such non-event trends range from Internet-only and platform-specific memes (e.g., a "fad" of users describing various things they object to using the #idonotsupport keyword), to hot topics of discussion (e.g., health care reform or the tween idol Justin Bieber). Therefore, before we can successfully identify trending events in social media, we must understand the scope of information that exhibits such trending behavior, with the particular goal of characterizing and distinguishing between trends that reflect event information and trends that reflect other non-event content. For this, we study, categorize, characterize, and compare *all* types of trends on one such popular social media site, namely, Twitter.

Social media content on Twitter indeed reflects an ever-updating live image of happenings, interests and attitudes in our society, which often include information related to events. However, the lack of a well-established structure and semantics for this data significantly limits its utility. In this chapter, we aim to characterize the features that can help identify and differentiate between the types of trends and trending events that we can find on Twitter. Better understanding of the semantics of Twitter trends provides critical information for techniques that build on this emerging data. In particular, understanding the differences between trends that reflect events and trends that do not is of critical importance to our goal of identifying events in social media. The outcome of our study will be a more robust and nuanced reflection of trends and, in turn, trending events that captures key aspects of relevance and importance.

This chapter offers the following contributions:

- A taxonomy of trends, including trending events, which can be detected from Twitter using popular, widely accepted methods
- Automatic characterization of the data associated with each trend along a number of key dimensions, including social network features, time signatures, and textual features
- Analysis of differences between trend types according to each characteristic

We begin with an introduction to Twitter and then formally describe our dataset of Twitter trends and their associated messages. Later, we describe a qualitative study exposing the types of trends found on Twitter. Finally, we identify and analyze trends using the unique social, temporal, and textual characteristics of each trend that can be automatically computed from Twitter content. The bulk of this chapter appeared in [NBG11].

3.1 Background: Twitter

Twitter is a popular social networking site, with hundreds of millions of registered users as of March 2011. Twitter's core function allows users to post short messages, or *tweets*, which are up to 140 characters long. Twitter supports posting (and consumption) of messages in a number of different ways, including through Web services and "third party" applications. Importantly, a large fraction of the Twitter messages are posted from mobile devices and services, such as Short Message Service (SMS) messages. A user's messages are displayed as a "stream" on the user's Twitter page.

In terms of social connectivity, Twitter allows a user to follow any number of other users. The Twitter contact network is directed: user A can follow user B without requiring approval or a reciprocal connection from user B. Users can set their privacy preferences so that their updates are available only to each user's followers. By default, the posted messages are available to anyone. In this work, we only consider messages posted publicly on Twitter. Users consume messages mostly by viewing a core page showing a stream of the latest messages from people they follow, listed in reverse chronological order.

The conversational aspects of Twitter play a role in our analysis of the Twitter temporal trends. Twitter allows several ways for users to directly converse and interact by referencing each other in messages using the @ symbol. A retweet is a message from one user that is "forwarded" by a second user to the second user's followers, commonly using the "RT @username" text as prefix to credit the original (or previous) poster (e.g., "RT @justinbieber Tomorrow morning watch me on the today show"). A reply is a public message from one user that is a response to another user's message, and is identified by the fact that it starts with the replied-to user @username (e.g., "@mashable check out our new study on Twitter trends"). Finally, a mention is a message that includes some other username in the text of the message (e.g., “attending a talk by @informor”). Twitter allows users to easily see all recent messages in which they were retweeted, replied to, or mentioned.

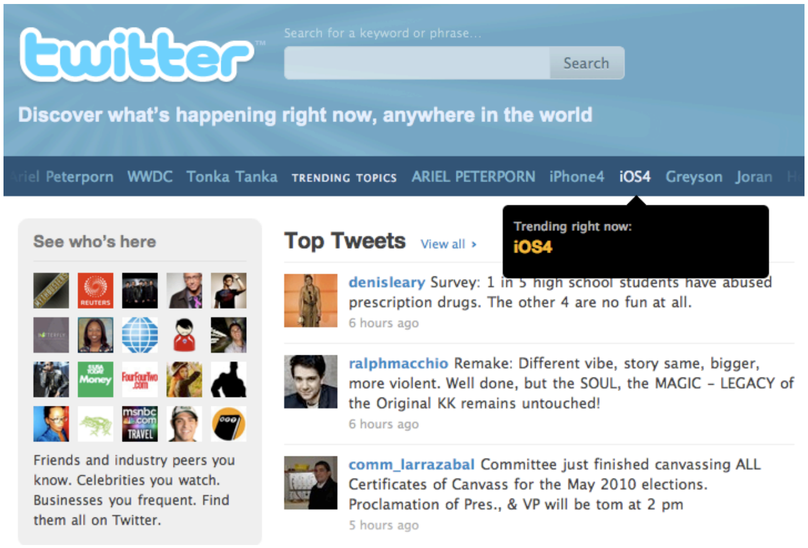
Finally, Twitter supports a hashtag annotation format so that users can indicate what their posted messages are about. This general “topic” of a tweet is, by convention, indicated with the hash sign #. For example, #iranelections was a popular hashtag with users posting about the 2009 Iran election events.

3.2 Trends on Twitter

Because of the quick and transient nature of its user posts, Twitter is an information system that provides a “real time” reflection of the interests and thoughts of its users, as well as their attention. As a consequence, Twitter serves as a rich source for exploring the mass attention of millions of its users, reflected in trends that can be extracted from the site.

Similar to our definition of trending events (Chapter 2), a trend on Twitter (sometimes referred to as a trending topic) consists of one or more terms and a time period, such that the volume of messages posted for the terms in the time period exceeds some expected level of activity (e.g., in relation to another time period or to other terms). Unlike trending events, trends do not have to reflect a real-world occurrence. According to this definition, trends on Twitter include our examples above, such as Michael Jackson’s death (with terms “Michael” and “Jackson,” and time period June 25, 2009), the final episode of Lost (with terms “Lost” and “finale,” and time period May 23, 2010), and the health care reform debate (with term “HCR” and time period May 25, 2010). This definition conveys the observation by Kleinberg [Kle03] that the “appearance of a topic in a document stream is signaled by a burst of activity, with certain features rising sharply in frequency as the topic emerges” but does not enforce novelty (i.e., a requirement that the topic was not previously seen). In Twitter’s own (very informal) definition, trends “are keywords that happen to be popping up in a whole bunch of tweets.” Figure 3.1 captures Twitter’s home page with several trending topics displayed at the top.

In this chapter, each trend t is then identified by a set R_t of one or more terms and a time period p_t. For example, Figure 3.1 highlights one trend t that is identified by a



Friends and industry peers you know. Celebrities you watch. Businesses you frequent. Find them all on Twitter.

denisleary Survey: 1 in 5 high school students have abused prescription drugs. The other 4 are no fun at all.
6 hours ago

ralphmacchio Remake: Different vibe, story same, bigger, more violent. Well done, but the SOUL, the MAGIC – LEGACY of the Original KK remains untouched!
6 hours ago

comm_larrazabal Committee just finished canvassing ALL Certificates of Canvass for the May 2010 elections. Proclamation of Pres., & VP will be tom at 2 pm
5 hours ago

Figure 3.1: Trending terms, on the dark blue (middle) banner, on Twitter's home page.

single term, iOS4 (referring to the release of Apple's mobile operating system). To analyze a trend t, we study the set M_t of associated messages during the time period that contain the trend terms (in our example, all messages with the string "iOS4"). Note that, of course, alternative representations of trends are possible (e.g., using lists of terms and other context features, as we discuss in Chapter 4). However, for this work we decided to concentrate on the above term-based formulation, which reflects a model commonly used in other systems (e.g., by Twitter as well as other commercial engines such as OneRiot).

While identifying trends, and specifically trending events, is an interesting research problem that we explore in this dissertation (Chapter 4), in this chapter we focus instead on *characterizing* the trends that can be identified on Twitter with existing baseline approaches. This characterization, and particularly the comparison of characteristics of trending events versus non-event trends, will later help inform our trending event identification techniques (Chapter 4).

For the characterization task, we collect detected trends from two different sources. First, we collect local trends identified and published hourly by Twitter; the trends are available via an application programmer interface (API) from the Twitter service. Second, to complement and expand the Twitter-provided trends, we run a simple burst-detection algorithm over a large Twitter dataset to identify additional trends. We describe these two trend-collection methods next.

3.3 Collecting Trend Data

In this section, we describe the two methods we use to compile trends on Twitter, and also how we select the set of trends for analysis and how we get the associated messages, or tweets, for each trend. The set of trends T that we will analyze in this chapter consists of the union of the trends compiled using both methods below. We use two methods in order to control, at least to some degree, for bias in the type of trends that may be detected by one method, but not another. While other algorithms for trend detection exist, we strongly believe our selected methods will provide a representative sample of the type of trends that can be detected. The set of detected trends might be skewed towards some trend types in comparison to other methods, but this skewness does not affect the analysis in this work. We further address this issue in the limitations discussion below.

In subsequent sections, we qualitatively examine a subset T_{Qual} of the trends in T to extract the key types of trends that are present in Twitter data, and develop a set of dimensions according to which trends can be categorized. We then use the categories to compare the trends in (a different) subset of T, T_{Quant}, according to several features computed from the data associated with each trend, such as the time dynamics of each trend and the interaction between users in the trend's tweets. We examine whether trends from different categories show significant difference in their computed features.

3.3.1 Tweets Dataset

The "base" dataset used for our study consists of over 48,000,000 Twitter messages posted by New York City users of Twitter between September 2009 and March 2010. This dataset is used in one of our methods described below to detect trends on Twitter (i.e., to generate part of our trend set T). The dataset is also used for identifying the set of tweets M_t for each trend t in our trend set T. (Recall that T consists of all the trends that we analyze, compiled using both methods discussed below.) We collected the tweets via a script for querying the

Twitter API. We used a “whitelisted” server, allowed to make a larger number of API calls per day than the default quota, to continuously query the Twitter API for the most recent messages posted by New York City users (i.e., by Twitter users whose location, as entered by the users and shown on their profile, is in the New York City area). This querying method results in a highly significant set of tweets, but it is only a subsample of the posted content. First, we do not get content from New York users who did not identify their home location. Second, the Twitter search API returns a subsample of matching content for most queries. Still, we collected over 48,000,000 messages from more than 855,000 unique users. For each tweet in our dataset, we record its textual content, the associated timestamp (i.e., the time at which the tweet was published), and the user ID of the user who published the tweet.

3.3.1.1 Trend Dataset I: Collecting Twitter’s Local Trending Terms

As mentioned above, one of our trend datasets consists of the trends computed by, and made available from, the Twitter service. Twitter computes these trends hourly, using an unpublished method. This source of trend data is commonly used in research efforts related to trends on Twitter (e.g., Kwak et al. [KLPM10], Cheong and Lee [CL09]). The Twitter-provided trends are computed for various geographic scales and regions. For example, Twitter computes and publishes the trends for New York City, as well as for the United States, and across all the Twitter service (e.g., those shown in Figure 3.1). From the data, we can observe that location-based trends are not necessarily disjoint: for example, New York City trends can reflect national trends or overlap with other cities’ trends.

We collected over 8,500 trends published by Twitter for the New York City area during the months of February and March of 2010. The data included the one or two terms associated with each published trend, as well as the trend’s associated time period, expressed as a date and time of day. We use the notation T_{tw} (for “Twitter”) to denote this set of trends.

3.3.1.2 Trend Dataset II: Collecting Trends with Burst Detection

We derived the second trend dataset using a simple trend-detection mechanism over our Tweets dataset described above. This simple approach is similar to those used in other efforts [NGS+09] and, as noted by Phelan et al. [PMS09], it "does serve to provide a straightforward and justifiable starting point." The trend-detection mechanism relies conceptually on the *tf-idf* score [MRS08] of terms, highlighting terms that appear in a certain time period much more frequently than expected for that time of day and day of the week. We tune this approach so that it does not assign a high score to weekly recurring events, even if they are quite popular, to ensure that we include a substantial fraction of trends that represent "one-time," non-recurring events, adding to the diversity of our analysis.

Specifically, to identify terms that appear more frequently than expected, we assign a score to terms according to their deviation from an expected frequency. Assume that M is the set of all messages in our Tweets dataset, R is a set of one or more terms to which we wish to assign a score, and h, d, and w represent an hour of the day, a day of the week, and a week, respectively. We then define $M(R, h, d, w)$ as the set of every Twitter message in M such that (1) the message contains all the terms in R and (2) the message was posted during hour h, day d, and week w. With this information, we can compare the volume in a specific day/hour in a given week to the same day/hour in other weeks (e.g., 10 a.m. on Monday, March 15, 2010, vs. the activity for other Mondays at 10 a.m.).

To define how we score terms precisely, let $Mean(R, h, d) = (\sum_{i=1,\ldots,n} |M(R, h, d, w_i|)/n$ be the number of messages with the terms in R posted each week on hour h and day d, averaged over the weeks w_1 through w_n covered by the Tweets dataset. Correspondingly, $SD(R, h, d)$ is the standard deviation of the number of messages with the terms in R posted each week on day d and hour h, over all the weeks. Then, the score of a set of terms R over a specific hour h, day d, and week w is defined as $score(R, h, d, w) = (|M(R, h, d, w)| - Mean(R, h, d))/SD(R, h, d)$.

Using this definition, we computed the score for every individual term in our dataset (in other words, we computed the scores for all R sets where each R is a set with a single 1-gram that appears in M). We computed the score for each R over all h, d, and w values for the weeks covered by our Tweets dataset. For each day d and week w, we identified

Notation	Data Source	Selection for Analysis
T_{tw}	Twitter's own trends as retrieved from the Twitter API	Selected from complete set of trends published by Twitter
T_{tf}	Trends computed from raw Twitter data using term frequency measures	Selected from top-scoring terms for each day

Table 3.1: Summary of event datasets.

the R and h pairs such that (1) $M(R, h, d, w)$ contains at least 100 messages and (2) the $score(R, h, d, w)$ value is among the top-30 scores for day d and week w across all term-hour pairs. Each selected pair defines a trend with set of terms R and associated time period specified by h, d, and w. (Note that certain terms could be repeated if they scored highly for multiple hours in the same day; such repetition is also possible for the trend set T_{tw}. We compute a trend's "real" peak after we choose the trends for analysis, as described below.) We use the notation T_{tf} (for "term frequency") to denote the resulting set of 1,500 trends.

For reference, the sources and properties of the event datasets are summarized in Table 3.1.

3.3.2 Selecting Trends for Analysis

After identifying the above two sets of trends, namely, T_{tw} and T_{tf}, our goal is to perform both a quantitative and a qualitative analysis of these trends. To be meaningful, this analysis will rely on a manual coding of the trends, but an exhaustive manual processing of all trends in T_{tw} and T_{tf} would, unfortunately, be prohibitively expensive. Therefore, our analysis will focus on a carefully selected subset of the two trend sets (see Section 3.4). This selection of trends should (1) reflect the diversity of trends in the original sets and (2) include only trends that could be interpreted and understood by a human, through inspection of the associated Twitter messages.

For both sets T_{tw} and T_{tf}, we performed a random selection of trends to serve as an initial dataset. For each trend in this initial selection, we attempted to identify the topic reflected in the trend by inspecting associated messages (posted on the corresponding day, and with the corresponding terms). If we could not identify the topic or reason for the trend, we

Trend Terms	Explanation	Date	# Tweets
#TEDxNYED	A New York City conference on media, technology, and education	March 6, 2010	556
Sparklehorse	The suicide of Mark Linkous, of the band Sparklehorse	March 8, 2010	230
Burger	Reaction to a tweet by Lady Gaga: “once you kill a cow, you gotta make a burger”	March 12, 2010	3249
Masters	Tiger Woods’s announcement of his return to golf at the Masters	March 16, 2010	693
itsreallyannoying	Twitter meme: users sharing their annoyances	March 23, 2010	2707
Seder	Passover-eve meal	March 28, 2010	316
iPad	Launch of the Apple iPad	March 29, 2010	1714

Table 3.2: Sample trends and their explanation.

removed it from the selected set, to satisfy condition (2). In addition, after the first round of coding trends according to the categories described below, we manually inspected the trends from the initial sets T_{tw} and T_{tf} that were not yet selected for analysis. Instead of randomly choosing among them, we randomly chose a date and then purposefully selected additional trends from that date from underrepresented categories, satisfying condition (1). Note that we attempted to create a comprehensive, but not necessarily proportional, sample of trends in the data. In other words, some types of trends may be over- or under-represented in the selected trends dataset. At the same time, the sample of trends in each category is representative of trends in the category overall. Our aim here is to provide insight about the categories of trends and features of trends in each category, rather than discuss the magnitude of each category in the data, a figure likely to shift, for example, with changes to the detection algorithms.

The result of this process was a set of trends T that combines trends from both T_{tw} and T_{tf}. We split the set T into two subsets. The first subset of selected trends, T_{Qual}, consisting

Initial set of Twitter trends (T_{tw})	$> 8{,}500^*$
Initial set of burst detection trends (T_{tf})	$> 1{,}500^*$
Selected trends for qualitative analysis (T_{Qual})	50
Selected trends for quantitative analysis (T_{Quant})	200

* Including duplicate trending terms in different hours.

Table 3.3: Details of the trend datasets produced and used in this work.

of trends in T through February 2010, was used for the qualitative analysis described in Section 3.4. The second subset of T, T_{Quant}, consisting of trends in T from March 2010, was used for the quantitative analysis described in Section 3.7. Table 3.2 lists several of the trends selected for the analysis: for each trend t, we list its description, time period, and number of associated messages (i.e., the cardinality of M_t). Next, we explain how we identify M_t for each trend t.

Table 3.3 provides a summary of the data sets described in this section, along with their respective size.

3.3.3 Identifying Tweets Associated with Trends

For our statistical analysis of trend features, for each trend t in T_{Quant} we need to know the set of tweets M_t associated with t. Each trend includes the terms that identify the trend and the associated time period, as discussed (e.g., a trend might consist of term “Passover” on March 29, 2010, for the hour starting at 4 p.m.). To define M_t, we first collected every message in our Tweets dataset that contains all of t's terms and such that it was posted up to 10 days before or after the time period for t. We sorted these messages according to the time at which they were posted and we aggregated them into hourly bins. Since the identifying term(s) may be popular at various times (e.g., as is the case for a trend that persists for several hours), we identified the peak time for the trend by selecting the bin with the largest number of messages. Finally, after anchoring the trend in its new associated time period, we retrieved all messages posted up to 72 hours before or after the new time period; this set is M_t, the set of messages associated with trend t. On average, the set M_t for each trend in T_{Quant} consists of 1,350 tweets, and the median cardinality of M_t is 573.

3.4 Trend Taxonomy and Dimensions

We now describe the qualitative analysis that we performed to characterize the Twitter trends in the T_{Qual} set of trends described above. The analysis was geared to identify the different types of trends that occur in Twitter data from one metropolitan area and relied on a taxonomy of the trends.

Several research effort showed that many Twitter trends correspond to trending events (Chapter 2) that are reflected on Twitter by its users [KLPM10; SST^{+}09]. Additionally, trends and trending events in social media have been characterized by researchers in the past. Dayan and Katz [DK92] characterized media events according to three generic types of scripts that these events tend to follow, namely, “contest,” “conquest,” and “coronation,” for events such as a presidential debate, an unfolding visit by a leader to a foreign state, and a leader’s funeral, respectively. Boll and Westermann [BW03] present discussion of events in the area of personal multimedia collections. However, the taxonomies available in these literatures do not capture the variety of trends that emerge in a social media site such as Twitter, which is our focus here.

Our qualitative analysis of trends is based on a variation of the affinity diagram method, an inductive process [LS99] to extract themes and patterns from qualitative data. For this analysis we used sticky notes to represent each trend in T_{Qual} and recorded the terms and the explanation of the trend if needed, which happened when the terms associated with the trend did not immediately offer an idea of the content. We then put together the different items into groups and categories in an iterative process of comparing, contrasting, integrating, and dividing the grouped trends. According to the affinity process, we considered the relationship between categories as well as the items that are grouped and linked together.

Indeed, the categories that emerged could be described and differentiated according to one key dimension: whether the trends in the category are exogenous or endogenous. Trends in exogenous categories capture trending events that originated outside of the Twitter system (e.g., an earthquake). Trends in endogenous categories are Twitter-only activities that do not correspond to external events (e.g., a popular post by a celebrity). Having this dimension at the top level of the taxonomy reflects and highlights the substantial differences on Twitter between exogenous and endogenous trends regarding their importance and use scenarios. The top level of the taxonomy thus separates trends that reflect real-world (i.e., non-virtual) events from trends that reflect activities that only pertain to the Twitter system.

The groups of trends that emerged are described below, with sample trends to illustrate each category.

Exogenous Trends

- Broadcast-media events:
  - Broadcast of local media events: "fight" (boxing event),"Ravens" (football game).
  - Broadcast of global/national media events: "Kanye" (Kanye West acts up at the MTV Video Music Awards), "Lost Finale" (series finale of Lost).
- Global news events:
  - Breaking news events: "earthquake" (Chile earthquake), "Tsunami" (Hawaii Tsunami warning), "Beyoncé" (Beyoncé cancels Malaysia concert).
  - Nonbreaking news events: "HCR" (health care reform), "Tiger" (Tiger Woods apologizes), "iPad" (toward the launch of Apple's popular device).
- National holidays and memorial days: "Halloween," "Valentine's."
- Local participatory and physical events:
  - Planned events: "marathon," "superbowl" (Super Bowl viewing parties), "patrick's" (St. Patrick's Day Parade).
  - Unplanned events: "rainy," "snow."

Endogenous Trends

- Memes: #in 2010 (in December 2009, users imagine their near future), "November" (users marking the beginning of the month on November 1).
- Retweets (users "forwarding" en masse a single tweet from a popular user): "determination" (users retweeting LL Cool J's post about said concept).

- Fan community activities: “2pac” (the anniversary of the death of hip-hop artist Tupac Shakur).

Needless to say, the above set of categories might not be comprehensive (i.e., other trends that are not in our data might not comfortably fit in any of these categories). However, we developed this set of categories after an exhaustive, thorough analysis of a large-scale set of trends, as described above. Therefore, we believe that this categorization is both sufficiently broad and, at the same time, simple enough to enable a meaningful study of the “trends in trend data.”

Our quantitative analysis (Section 3.7) focused on a limited number of dimensions extracted from the taxonomy that capture key differences between trends. We identified the dimensions to focus on according to two criteria: (a) significance, or the importance of being able to extract differences between the selected trend categories, and (b) the likelihood that these categories will result in measurable differences between trends.

The first dimension we examined is the high-level exogenous and endogenous categories of trends. Such comparison will allow us to reason about this most distinguishing aspect of any Twitter trend. Importantly, this comparison is critical to our understanding of trending events on Twitter, which we leverage in the following chapters.

Within exogenous trends, in this work we chose to concentrate on two important dimensions. First, whether the exogenous activity falls into the local participatory and physical events category above. These “local trends” represent physical events, located in one geographic area (e.g., the New York marathon) that are currently underrepresented in the detected trends, but naturally play an important role in local communities. The second dimension chosen is whether the exogenous trends are breaking news events, global news events that are surprising and have not been anticipated (e.g., an earthquake), as opposed to all other events and trends that are planned or expected (e.g., a vote in the Senate, or a holiday). This dimension will allow us to separate “news-worthy” versus “discussion-worthy” trends, which may lead to a different manner in which systems use and display these different trend types.

Similarly, within endogenous trends in this work we chose to investigate the differences between trends in the two main categories of this group of events, namely, memes and retweets, as explained above.

Next, these dimensions help us guide the quantitative study of the trends detected in Twitter data, as we label each trend according to categories derived from the dimensions above.

3.5 Characterization of Trends and Events

The next step in our analysis is to characterize each Twitter trend using features of its associated messages. These features are later used to reason about differences between the various trend dimensions described in the previous section. For this analysis, we use the trend set T_{Quant} defined above. For each trend t, we compute features automatically, based on its associated set of tweets M_t. These features range from aggregate statistics of the content of each individual message (e.g., number of hashtags, URLs) to social network connections between the authors of the messages in M_t, and the temporal characteristics of M_t.

Content Features Our first set of features (see Table 3.4) provides descriptive characteristics for a trend t based on the content of the messages in M_t. These features include aggregate characteristics such as the average length of a message in M_t and the percentage of messages with URLs or hashtags, or measures of the textual similarity of the tweets in M_t.

Interaction Features The interaction features (see Table 3.5) capture the interaction between users in a trend's messages as indicated on Twitter by the use of the "@" symbol followed by a username. These interactions have somewhat different semantics on Twitter, and include "retweets" (forwarding information), replies (conversation), or mentions of other users.

Time-based Features The time-based features (see Table 3.6) capture different temporal patterns of information spread that might vary across trends. To capture these features for a trend t, we fit a family of functions to the histogram describing the number of Twitter

Content Features	Explanation
Average number of words and characters	Let $words(m)$ be the number of words in a tweet m and let $char(m)$ be the number of characters in tweet m. Then the average number of words per message is $\frac{sum_{m \in M_t} words(m)}{\|M_t\|}$, and the average number of characters per message is $\frac{sum_{m \in M_t} char(m)}{\|M_t\|}$.
Proportion of messages with URLs	Let $U_t \subseteq M_t$ be the set of messages with URLs out of all messages for trend t. Then the proportion of messages with URLs is $\|U_t\|/\|M_t\|$.
Proportion of unique URLs	Let $URL(m)$ be the set of URLs that appear in tweet m. The set of unique URLs for t is $\|UU_t\|$, where $UU_t = \{u : u \in URL(m)$ for a message $m \in M_t\}$, and the proportion of unique URLs is $\|UU_t\|/\|M_t\|$. (Note that the set semantics ensures that each unique URL is only counted once.)
Proportion of messages with hashtags	Let $H_t \subseteq M_t$ be the set of messages with hashtags in M_t . Then the proportion of messages with hashtags is $\|H_t\|/\|M_t\|$.
Proportion of messages with hashtags, excluding trend terms	Let $H_t \subseteq M_t$ be the set of messages with hashtags in M_t, excluding messages where the hashtag is a term in R_t, the set of terms associated with trend t. Then the proportion of messages with hashtags excluding the trends terms is $\|H_t\|/\|M_t\|$.
Top unique hashtag?	Whether there is at least one hashtag that appears in at least 10% of the messages in M_t. This measure captures agreement on the terms most topically related to the trend.
Similarity to centroid	We represent each message $m \in M_t$ as a *tf-idf* vector, where the idf value is computed with respect to all messages in the Tweets dataset.We compute the average *tf-idf* score for each term across all messages in M_t to define the centroid C_t. Using C_t, we then compute the average cosine similarity $\frac{sum_{m \in M_t} sim(C_t,m)}{\|M_t\|}$ as well as the corresponding standard deviation. These features help indicate content cohesiveness within a trend.

Table 3.4: Content features for a trend t.

Interaction Features	Explanation
Proportion of retweets	Let $RT_t \subseteq M_t$ be the set of messages in M_t that are "retweets" (i.e., these messages include a string of the form "RT @user"). Then the proportion of retweets is $\|RT_t\|/\|M_t\|$.
Proportion of replies	Let $RP_t \subseteq M_t$ be the set of messages in M_t that are "replies" (i.e., these messages begin with a string of the form "@user"). Then the proportion of replies is $\|RP_t\|/\|M_t\|$.
Proportion of mentions	Let $MN_t \subseteq M_t$ be the set of messages in M_t that are "mentions" (i.e., these messages include a string of the form "@user" but are not replies or retweets as defined above). Then the proportion of mentions is $\|MN_t\|/\|M_t\|$.

Table 3.5: Interaction features for a trend t.

messages associated with the trend over the time period spanned by the tweet set M_t (by construction, as discussed, M_t has the matching messages produced up to 72 hours before and after t's peak). We aggregate all messages in M_t into hourly bins. We refer to all bins before the peak as the head of the time period, while all bins after the peak are the tail of the time period.

We proceed to fit the bin volume data, for both the head and the tail of the time period, separately, to exponential and logarithmic functions. Using the least squares method, we compute logarithmic and exponential fit parameters for the head and tail periods for each trend, considering the full time period of 72 hours, which we refer to as the Log72 fit and the Exp72 fit, respectively. We proceed in the same manner for a limited time period of 8 hours before and after the peak, which we refer to as the Log8 fit and the Exp8 fit, respectively. The focus on the shorter time periods will allow us to better match rapidly rising or declining trends [LBK09].

In sum, our features for each trend thus include the fit parameters for 8-hour and 72-hour spans for both the head and the tail periods; and for each period and span, we calculate the logarithmic and exponential fit parameters. In addition, for each combination we also computed the R^2 statistic, which measures the quality of each fit.

Time-based Features	Explanation
Exponential fit (head)	Best fit parameters (p_0, p_1, p_2) and goodness of fit R^2 for function $M(h) = p_1 e^{-p_0 \|h\|} + p_2$, where $M(h)$ represents the volume of messages during the h-th hour before the peak. Computed for 72- and 8-hour periods before the peak.
Logarithmic fit (head)	Best fit parameters (p_0, p_1) and goodness of fit R^2 for function $M(h) = p_0 log(h) + p_1$, where $M(h)$ represents the volume of messages during the h-th hour before the peak. Computed for 72- and 8-hour periods before the peak.
Exponential fit (tail)	Similar to above, but over 72- and 8-hour periods after the trend's peak.
Logarithmic fit (tail)	Similar to above, but over 72- and 8-hour periods after the trend's peak.

Table 3.6: Time-based features for a trend t.

Participation Features Trends can have different patterns of participation, in terms of authorship of messages related to the trend. The participation features (see Table 3.7) characterize a trend using statistics about the participation of authors that produced the trend's associated messages; in particular, we capture the skew in participation (i.e., to which extent a small portion of authors produced most of the content).

Social Network Features Our final group of features for a trend t focuses on the set A_t of the authors of the messages in M_t. Specifically, the social network features (see Table 3.8) capture the properties of the social network G_t of authors. To model this network, we used the Twitter API to collect the list of followers for each author, consisting of other Twitter users in A_t that subscribe to the author's message feed. (We ignore followers that are not among the A_t authors. We also ignore followers of authors who restrict access to this information, and those who have suspended Twitter accounts.) In other words, our social network graph is a directed graph $G_t(A_t, E_t)$, such that there exists an edge $e \in E_t$ from a_1 to a_2 if and only if a_1 is a follower of a_2 on Twitter. We computed various features

Participation Features	Explanation
Messages per author	Let $A_t = \{a : a \text{ is an author of a message } m \in M_t\}$. Then the number of messages per author is $\lvert M_t\rvert / \lvert A_t\rvert$.
Proportion of messages from top author	We designate $a' \in A_t$ as the top author if a' produced at least as many messages in M_t as any other author. Then the proportion of messages from top author is $\lvert\{m : m \in M_t \text{ and } m \text{ was posted by } a'\}\rvert / \lvert M_t\rvert$.
Proportion of messages from top 10% of authors	Let $A10_t$ be the set of the top 10% of the authors in terms of the number of messages produced in M_t. Then the proportion of messages from top 10% authors is $\lvert\{m : m \in M_t \text{ and } m \text{ was posted by } a \in A10_t\}\rvert / \lvert M_t\rvert$.

Table 3.7: Participation features for a trend t.

of the social network graph G_t for each trend t, capturing the connectivity and structure of connections in the graph [WF94].

3.6 Categorizing Trends in Different Dimensions

In addition to the automatically extracted features, we manually categorized the trends in T_{Quant} according to the dimensions picked for analysis (e.g., whether the trend belongs to the “exogenous” or “endogenous” category). We manually associated every trend with one category in each dimension. Later, we examined how the categories differ according to the automatically computed features described above.

We required a content description of each trend in order to properly label it according to the categories introduced in the previous section. The trend detection methods only output the trend terms and a time period. This type of output (e.g., “Bacall on March 8th”) was often not enough to discern the content of the trend to correctly assign it to different categories. We examined each of the trends to generate a short description. The sources used for this examination were, first, the actual Twitter messages associated with the trend. If that examination did not prove informative enough, we used news search tools (e.g., Google News) to inspect corresponding news reports for that day and those terms. At the end of the process, we had a description for 200 of the trends in our trend dataset

Social Network Features	Explanation
Level of reciprocity	The fraction of reciprocal connections out of the total number of connections $\lvert E_t \rvert$, where authors $a_1, a_2 \in A_t$ form a reciprocal connection if $(a_1, a_2) \in E_t$ and $(a_2, a_1) \in E_t$.
Maximal eigenvector centrality	The eigenvector centrality of an author measures the importance of this author in A_t by computing the eigenvector of the largest eigenvalue in the adjacency matrix of the network graph. We pick the author with the highest eigenvector centrality value over all $a \in A_t$. A high value suggests the existence of a dominant node in the network.
Maximal degree centrality	The degree centrality of an author $a \in A_t$ is the fraction of authors it is connected to. We compute the highest degree centrality value over all $a \in A_t$. A high value suggests the existence of a dominant node in the network.

Table 3.8: Social network features for a trend t.

T_{Quant}, after removing twenty-nine trends that could not be resolved (e.g., “challenging” on March 14, 2010) from our dataset. We computed these features for the 200 resolved trends in our trend dataset T_{Quant}. This data is the basis for our analysis, described below.

We mapped each of the trends into categories based on the dimensions for analysis. Two people independently annotated each of the trends. If an annotator could not assign a value for some dimension, either a “not applicable” or an “unknown” label was used. In each dimension, after removing trends marked “not applicable” or “unknown” by at least one of the annotators, the inter-annotator agreement of the labeled trends in each dimension was very high (the remaining number of trends for each dimension is reported below, in the analysis). For the final analysis in each dimension we removed all “not applicable” and “unknown” entries for that dimension, as well as any remaining disagreements between the annotators. In other words, we ignored those trends for which we had reason to doubt the assignment to a category.

3.7 Quantitative Analysis

The main drivers for our analysis are the coded categories of trends, as detailed above. In other words, we compared the samples of trends according to their categorization in different dimensions (e.g., exogenous vs. endogenous) and according to the features we computed from the data (e.g., the percentage of messages with URLs). Our hypotheses, listed below, are guided by intuitions about deviations in the characteristics of trends in different categories, and are geared towards confirming the expected deviations between the trend categories. Such confirmation would allow, later on, for the development of automated systems to detect the trend type or provide better visualizations or presentation of the trend data. We continue by listing the key hypotheses that guided our analysis.

Exogenous vs. Endogenous Trends H1. We hypothesize that exogenous and endogenous trends will have different quantitative characteristics. In particular:

- H1.1 Content features of exogenous trends will be different than those of endogenous trends; in particular, they will have a higher proportion of URLs and a smaller proportion of hashtags in tweets.
- H1.2 Interaction features of exogenous trends will be different than those of endogenous trends; in particular, exogenous trends will have fewer retweets (forwarding), and a similar number of replies (conversation).
- H1.3 Time features of exogenous trends will be different for the head period before the trend peak but will exhibit similar time features in the tail period after the trend peak, compared to endogenous trends.
- H1.4 Social network features of exogenous trends will be different than those of endogenous trends, with fewer connections (and less reciprocity) in the social network of the trend authors.

Breaking News vs. Other Exogenous Trends H2. We hypothesize that breaking news events will have different quantitative characteristics compared to other exogenous trends. In particular:

- H2.1 Interaction features of breaking events will be different than those of other exogenous trends, with more retweets (forwarding), but fewer replies (conversation).
- H2.2 Time features of breaking events will be different for the head period, showing more rapid growth, and a better fit to the functions' curve (i.e., less noise) compared to other exogenous trends.
- H2.3 Social network features of breaking events will be different than those of other exogenous trends.

Local Events vs. Other Exogenous Trends H3. We hypothesize that local participatory and physical events will have different quantitative characteristics compared to other exogenous trends. In particular:

- H3.1 Content features of local events will be different than those of other exogenous trends.
- H3.2 Interaction features of local events will be different than those of other exogenous trends; in particular, local events will have more replies (conversation).
- H3.3 Time features of local events will be different than those of other exogenous trends.
- H3.4 Social network features of local events will be different than those of other exogenous trends; in particular, local events will have denser networks, more connectivity, and higher reciprocity.

Memes vs. Retweet Endogenous Trends H4. We hypothesize that memes will have different quantitative characteristics compared to retweet trends. In particular:

- H4.1 Content features of memes will be different than those of retweet trends.
- H4.2 Interaction features of memes will be different than those of retweet trends; in particular, retweet trends will have significantly more retweet (forwarding) messages (this hypothesis is included as a "sanity check" since the retweet trends are defined by having large proportion of retweets).

- H4.3 Time features of memes will be different than those of retweet trends.
- H4.4 Participation features of memes will be different than those of retweet trends.
- H4.5 Social network features of memes will be different than those of retweet trends; in particular, meme trends will have more connectivity and higher reciprocity than retweet trends.

We performed our analysis on the 200 resolved trends in T_{Quant}. The analysis was based on a pairwise comparison of trends according to the trends' categorization in different dimensions, following our hypotheses above. For each such pair, we performed a set of two-tailed t-tests to show whether there are differences between the two sets of trends in terms of the dependent variables, namely, our automatically extracted trend features. However, since each sub-hypothesis involved multiple dependent variables (e.g., we computed seven different social network features), we controlled for the multiple t-tests by using the Bonferroni correction, which asks for a significance level of α/n when conducting n tests at once. Therefore, we only report here results with significance level of $p < .008$.

As is common in studies of social-computing activities, many of our dependent variables were not normally distributed, but rather they were most often skewed to the right. Following Osborne [Osb02], we used logarithms (adding a small constant to handle zero values as needed) or square root functions to transform these variables in order to improve their normality. For most variables, such transformation indeed generated a normal distribution. In the cases where we performed a variable transformation, whenever we find significant differences between the transformed means in the analysis, we also report here the original variable means and medians. For variables that were still skewed after the transformation, we performed the Mann-Whitney test for non-normal distributions, and note when that is the case. For the one dependent variable in our data that was nominal, we used the CHI-square test. Finally, following Asur and Huberman [AH10], in the analysis we considered the temporal features only for trends that peaked on a United States weekday (i.e., Monday through Friday), as the temporal aspects in particular might be influenced by the different patterns of Twitter usage during weekends.

Categories Compared		Content	Interaction	Time	Participation	Social
Exogenous vs. Endogenous	Hypothesis:	H1.1	H1.2	H1.3	None	H1.4
	Found:	Yes*	Yes*	No	No	Yes
Breaking News vs. Other Exogenous	Hypothesis:	None	H2.1	H2.2	None	H2.3
	Found:	No	Yes	No	No	No
Local vs. Other Exogenous	Hypothesis:	H3.1	H3.2	H3.3	None	H3.4
	Found:	No	Yes*	No	No	No
Memes vs. Retweets	Hypothesis:	H4.1	H4.2	H4.3	H4.4	H4.5
	Found:	Yes	Yes	Yes*	Yes	Yes

Table 3.9: Summary of results. Starred entries represent partial findings or findings that diverged somewhat from the detailed hypothesis.

3.8 Experimental Results

We report below the results from our analysis. For convenience, an overview of the results and findings as they related to the hypothesis is provided in Table 3.9.

3.8.1 Exogenous vs. Endogenous Trends

Exogenous trends were found to be different than endogenous trends in content, interaction, time and social features, supporting most of the hypotheses under H1 as shown in Table 3.9. In our dataset we had 115 exogenous trends and 55 endogenous trends (for some parts of the analysis the numbers are lower due to missing data). The detailed numerical results are shown in Tables 3.10 and 3.11. In terms of content features (H1.1), exogenous trends had a higher proportion of messages with URLs than endogenous trends (results were similar for the proportion of unique URLs appearing in the trend's content). In addition, the average term length for exogenous trends was somewhat shorter than the length of terms used in endogenous trends. We found only some differences in the presence of hashtags in the content: exogenous trends did not have a higher proportion of messages with hashtags, even when excluding the trending terms. However, fewer exogenous trends had a unique hashtag appearing in at least 10% of the messages compared to endogenous trends. This finding

	URL Proportion*		Top Unique Hashtag+		Term Length (chars)†	
	Exo	Endo	Exo (Y/N)	Endo (Y/N)	Exo	Endo
N	115	55	47/68	36/19	115	55
Mean	.34	.144	-	-	5.31	6.13
Median	.307	.058	-	-	-	-

* log-transformed, t=6.117, p<.001

$^+$ χ^2=9.00, p<.002

† t=-5.119, p<.001

Table 3.10: Quantitative analysis results of exogenous/endogenous categories.

	Retweet Proportion+		Reply Proportion*		Reciprocity†	
	Exo	Endo	Exo	Endo	Exo	Endo
N	115	55	115	55	114	54
Mean	.32	.47	.094	.083	.26	.33
Median	.26	.38	.081	.028	-	-

* sqrt-transformed, t=-3.865, p<.001

$^+$ log-transformed, t=2.98, p<.003

† t=-6.87, p<.001

Table 3.11: Quantitative analysis results of exogenous/endogenous categories.

indicates less agreement between authors of exogenous trends on the ad-hoc "semantics" of the trend (in other words, the chosen community representation for what that trend content is about), which may stem from the fact that exogenous trends are seeded at once from many users who choose different hashtags to represent the trend.

In terms of interaction features (H1.2), we found that exogenous trends had a smaller proportion of retweets in the trend's tweets compared to endogenous trends. This finding suggests that users created more original content based on exogenous sources, rather than retransmit and forward content that was already in the "system" as often happens for endogenous trends. Interestingly, we also found that exogenous trends tend to have more "conversation": the proportion of replies in exogenous trends was higher than endogenous ones. In terms of time features (H1.3), the hypothesis was not supported: our data did not show exogenous trends to have different time features for the head period. The tail

	Retweet Proportion*		Reply Proportion+		*Exp72* tail $R^{2\dagger}$		*Log72* tail $R^{2\ddagger}$	
	Breaking	Other	Breaking	Other	Breaking	Other	Breaking	Other
N	33	63	33	63	26	37	26	39
Mean	.39	.28	.058	.11	.0119	.0041	.589	.386
Median	.35	.21	.049	.101	-	-	-	-

* sqrt-transformed, t=3.2, p<.002

+ log-transformed, t=-2.7, p<.008

† t=3.554, p<.001

‡ t=4.508, p<.001

Table 3.12: Quantitative analysis results of breaking/other categories.

period time parameters were, as we hypothesized, not found to be different for exogenous and endogenous trends.

Finally, in terms of social network features (H1.4), we found differences in the level of reciprocity between exogenous and endogenous trends. Social network connections in exogenous trends had less reciprocity than those of endogenous trends. Other differences were found but with marginal significance.

3.8.2 Breaking News vs. Other Exogenous Trends

Trends corresponding to breaking events were found to have different interaction characteristics from other exogenous trends, but no other differences were found, giving only partial support to hypothesis H2 (Table 3.9). In our dataset, we had 33 breaking events and 63 other exogenous events (for some parts of the analysis the numbers are lower due to missing data). The detailed numerical results are shown in Table 3.12.

In terms of interaction features (H2.1), we found that breaking exogenous trends have a larger proportion of retweet messages than other exogenous events. Breaking trends also have a smaller proportion of reply messages than other exogenous events. These findings show the informational nature of breaking events, which focus on information transmission rather than conversation.

Hypotheses H2.2 and H2.3 were not confirmed, though, finding no significant differences in time features between breaking exogenous events and other exogenous events. It is noted,

	Retweet Proportion*	
	Local	Other
N	12	96
Mean	.18	.345
Median	.148	.2

* sqrt-transformed, t=-4.82, p<.001

Table 3.13: Quantitative analysis results of local/other categories.

however, that we found that the R2 quality of fit on the Exp72 time fit parameters for the tail period was significantly different between breaking and other events, with breaking events having better fit on average than other events. Similar yet marginal differences were found for Log72 fit parameters. This difference might suggest that the breaking events, after the peak, are less noisy than other exogenous events with discussion levels dropping more "smoothly."

3.8.3 Local Events vs. Other Exogenous Trends

We found limited support that local events have different characteristics than other exogenous trends (H3). In particular, our data surfaced differences between interaction features of local events and other exogenous trends (Table 3.9). In our dataset, we had 12 local events and 96 other exogenous trends (for some parts of the analysis the numbers are lower due to missing data). We note that the analysis was limited by the small number of local events in our trends dataset. The detailed numerical results are shown in Table 3.13.

We could confirm only one difference in terms of interaction features between local and other exogenous trends (H3.2), where local events have a smaller proportion of messages that are retweets than other exogenous trends. In addition, our analysis suggests that local events might be more conversational, in terms of the proportion of messages that are replies, than other exogenous trends; however, the result for replies is not significant at the level we require for reporting in this paper, and thus cannot be fully confirmed. Therefore, we can provide only partial support to H3.2.

Finally, we found no support for H3.3, as the differences in time features between local

	URL Proportion*		Unique URL Proportion+		Hashtag Proportion†		Length (terms)‡	
	Memes	RTs	Memes	RTs	Memes	RTs	Memes	RTs
N	27	22	27	22	27	22	27	22
Mean	.044	.24	.035	.064	.989	.275	13.19	17.83
Median	.032	.103	.029	.06	.998	.092	-	-

* log-transformed, t=-4.231, p<.001

+ log-transformed, t=-2.759, p<.008

† log-transformed, t=5.552, p<.001

‡ t=-5.156, p<.001

Table 3.14: Quantitative analysis results of meme/retweet categories.

	Length (chars)*		Term Lengths (chars)+		Top Unique Hashtag†		Retweet Proportion‡	
	Memes	RTs	Memes	RTs	Memes (Y/N)	RTs (Y/N)	Memes	RTs
N	27	22	27	22	27/0	7/15	27	22
Mean	80.3	106.1	6.65	5.57	-	-	.309	.692
Median	-	-	-	-	-	-	.277	.739

* t=-4.621, p<.001

+ t=4.017, p<.001

† χ^2=27.99, p<.001

‡ sqrt-transformed, t=-8.633, p<.001

Table 3.15: Quantitative analysis results of meme/retweet categories.

events and other exogenous trends could not be confirmed.

3.8.4 Memes vs. Retweet Endogenous Trends

Looking at endogenous trends, “retweet” trends were found to be different than “meme” trends (H4) in content, interaction, time, participation, and social network features (see Table 3.9). In our dataset, we had 29 memes and 27 retweet trends (for some parts of the analysis the numbers are lower due to missing data). The detailed numerical results are shown in Tables 3.14-3.17.

In terms of content (H4.1), we found several differences between the retweet trends and meme trends. Retweet trends have a larger proportion of messages with URLs than meme

	Reply Proportion*		*Log8* head p_1+		Messages/author†		Messages/top author‡	
	Memes	RTs	Memes	RTs	Memes	RTs	Memes	RTs
N	27	22	7	15	27	22	27	22
Mean	.029	.079	.055	-.109	2.067	1.171	.042	.019
Median	.018	.0565	-	-	-	-	.018	.01

* log-transformed, t=-3.704, p<.001

+ t=3.549, p<.002

† Mann-Whitney Z=5.44, p<.001

‡ log-transformed, t=2.793, p<.008

Table 3.16: Quantitative analysis results of meme/retweet categories.

	Messages/top-10% authors*		SCC size (avg)+		Reciprocated Ties†	
	Memes	RTs	Memes	RTs	Memes	RTs
N	27	22	27	22	27	22
Mean	.382	.182	1.861	1.332	.363	.3
Median	.383	.157	1.582	1.239	-	-

* sqrt-transformed, t=8.814, p<.001

+ log-transformed, t=3.53, p<.001

† t=3.936, p<.001

Table 3.17: Quantitative analysis results of meme/retweet categories.

trends, and a higher proportion of unique URLs. More meme trends have a single hashtag that appears in more than 10% of the trend's messages. Accordingly, meme trends have a larger proportion of hashtags per message than retweet trends, but are not different when we remove the trending terms from consideration (indeed, memes are often identified by the hashtag that the relevant messages contain). Finally, retweet trends have more textual terms in the tweets than meme trends, and the retweet trend tweets are longer on average than meme trend tweets, and are even longer when counting characters in URLs (not reported here). However, these differences may be attributed to the "RT @username" phrase added to individual retweet messages, which were more common, of course, for retweet trends (this observation was true at the time the data was collected; since then, Twitter has changed its format for retweet messages, so that "RT @username" does not always appear in the data).

Supporting H4.2, retweet trends naturally have a significantly greater proportion of messages that are retweets than meme trends. Retweet trends also have a greater portion of replies, showing that they are slightly more conversational than meme trends. The retweet and meme trend categories are not different in their proportions of mentions. Regarding time features, addressing H4.3, we found one difference between the time fit parameters: the head fit parameter Log8 head p_1, indicating different growth for retweet trends. This finding may suggest that retweet trends develop in a different manner than meme trends.

Looking at the participation features (H4.4), we found a number of significant differences between retweet and meme trends, supporting the hypothesis. Meme trends have more messages per author on average than retweet trends in a statistically significant manner (we performed the Mann-Whitney test due to the non-normal distribution of this parameter). In addition, meme trends have a higher proportion of messages from the single top author than retweet trends, as well as a higher proportion of messages from the top 10% of authors than retweet trends. These results show a significant difference in participation between these types of trends, where meme trends are more skewed, and a limited number of users are responsible for a fairly significant part of the content, and retweet trends are more "democratic" and participatory.

Finally, retweet trends were significantly different than meme trends in a number of social network features, confirming H4.5. In terms of the proportion of reciprocation in the trends' authors social network, retweet trends had a lower level of reciprocated ties than meme trends. Meme trends also had a higher average size of strongly connected components, than retweet trends. These findings suggest that retweet trends are supported by a network that, while showing the same density, builds on directional, informational ties more than meme trends that are supported by communication and reciprocity.

3.9 Discussion

The results of our quantitative analysis provide a strong indication that we can use the characteristics of the messages associated with a trend to reason about the trend, for example, to better understand the trend's origin and context. In particular, we found that exogenous trends, corresponding to real-world occurrences that originate from outside the Twitter system but reflected in the activity of users in the system, are different in a number of important features from endogenous trends, which are topics that start and develop in the Twitter "universe." Connections between the authors of messages in endogenous trends tend to be more symmetrical (i.e., with higher reciprocity) than in exogenous trends, suggesting perhaps that endogenous trends require stronger ties to be "transmitted." However, we also expected the density of the endogenous trend networks and the average degree of their nodes to be higher, but did not find any such differences. Differences between these two categories of trends were not evident in the temporal features, where the results did not support our hypothesis of a more rapid curve leading to the peaks of the exogenous trends. However, the differences between these categories are further supported by the deviations in content and interaction features between the categories: more URLs, unique URLs, and unique hashtags, as well as a smaller proportion of retweets, show that exogenous trends generate more independent contributions than endogenous trends do.

In a deeper examination of the differences between categories of exogenous trends, we found only interaction differences between trends representing "breaking" events and other type of exogenous trends. Specifically, breaking events are, naturally perhaps, more "informational" and less "conversational" in nature than other trends. Significantly, we could not confirm the hypothesis from Sakaki et al. [SOM10] that breaking events will be more disconnected, as multiple contributors will independently contribute messages with less in-network coordination. However, one possible reason for not seeing this effect in the data is the long period of content (72 hours before and after a trends peak) over which we calculate the author networks. Perhaps focusing on the connection between authors in the first hours of a trend would capture these differences between breaking events and other trends.

Trends capturing local events were found to be only slightly different than other exogenous trends mainly with respect to the interaction features. People discuss more, and forward information less, in the context of local events as compared to other exogenous trends. Note again that we have a low number of local events represented in our trend dataset and these findings should be considered tentative. Yet, it is possible that differences between local events and other trends would be even more pronounced when more data is available.

Finally, we have shown that even endogenous trends, which grow and develop from within the Twitter system and are not a reflection of external events, could have different categories that are varied across a number of key features. Retweet trends, where users respond and forward a message from a single popular user, are different in many characteristics (including content, interaction, time, participation and social characteristics) than meme trends.

Before we conclude, we list several important considerations about our study, acknowledging a few limitations and biases in the work. One limitation is in the dataset used in this work, which is incomplete because of two reasons. First, we generated the initial set of trends using two specific, albeit well-established methods. (As we discussed, the focus of this chapter is not on trend detection but rather on the analysis and characterization of the trends and trending events.) However, other methods for trend detection and identification of related content might assist in capturing additional trends and trending events. For instance, in Chapters 4 and 5, we develop comprehensive techniques for unknown identification of trending events, exploiting both textual and non-textual information. At the same time, we believe that the sample of trends reflects the span of trend categories that can reasonably be detected by any method. The second reason why our dataset is incomplete relates to the selection of the tweets that we used for both trend extraction and characterization: specifically, we only included content from New York City users who disclosed their hometown location in their profile and hence excluded content from other local users without an explicit profile location. (Automatically matching locations and users with no explicit geographical information in their profiles is the subject of interesting future work.) In addition, we defined each trend using terms, and we retrieved the messages associated with each trend via simple keyword search. In the next chapter, we explore alternative techniques for identifying messages for trending events, specifically using a clustering framework, so that we could associate a message with a trending event even without requiring a strict term overlap.

Furthermore, our analysis focuses on a single system (i.e., Twitter) and a single location (i.e., New York City). Other dynamics and trend characteristics may exist in other systems and locations (e.g., involving Facebook data and concerning users based in Paris, France). Indeed, the dynamics we observed, and some of the characteristics we extracted, are unique to Twitter. However, Twitter is an important communication and information service that has already made considerable impact on our society, and is important to study regardless of generalization to other social media sites. Moreover, we have no reason to believe that, other than message volume, trends involving New York City users are significantly different from trends for other locations.

The metrics that we have used to characterize trends can be extended or further developed. For example, for the time-based characterization, one could experiment with different fitting functions, identifying peaks in different ways (e.g., considering the expected volume of tweets for each time of the day), using different time periods before and after the peak, and so forth. In another example, the social network characteristics could consider the social network of authors that appeared in the first 24 hours of the trends, following Yardi and boyd [Yb10], which might produce networks of different characteristics. These different methods could expose more pronounced differences between trend categories. Still, the wide-ranging set of metrics presented here has already helped identify key differences between types of trends and trending events. These differences bring useful insight into the nature of trends and trending events in social media, which we use to inform our unknown identification techniques (Chapters 4 and 5).

3.10 Conclusions

Emerging temporal trends in social media sites such as Twitter are a significant and revealing source of information for, and about, trending events. In this chapter, we categorized and characterized trends on Twitter, and showed that different types of trends exhibit significant differences in terms of various automatically computed characteristics. Our findings suggest directions for automatically distinguishing between different types of trends, perhaps using machine learning or model-based approaches, utilizing the trend characteristics we propose above as well as others. Importantly, our study reveals and distinguishes between endogenous, platform-centric trends, and the trending events on which we focus in this dissertation. Given these findings, we can enable a robust classification of trends into the various trend categories we identified in general, and into trending events (i.e., exogenous trends) and non-event trends (i.e., endogenous trends) in particular. This separation will prove valuable for identifying trending events in social media, as we will see in the next chapter.

Chapter 4

Identification of Unknown Events and Their Content

Short messages posted on social media sites such as Twitter can typically reflect events as they happen. For this reason, the content of such social media sites is particularly useful for timely, unknown event identification, which is the problem that we address in this chapter. As we discussed, in this unknown identification scenario we focus on a class of events defined as *trending events* (Chapter 2). Twitter content associated with trending events can provide a set of unique perspectives [DNKS10; Yb10], reflecting the points of view of users who are interested or even participate in an event. At the same time, much of the content on Twitter does not correspond to any particular trending event, making the separation between trending event and non-event content challenging and essential for the unknown event identification task.

From our study of trending event content on Twitter (Chapter 3), we found that trending events exhibit temporal patterns that can be identified using burst detection techniques [Kle03]. However, these techniques often also identify endogenous trends that do not correspond to trending events. Therefore, to identify the *trending events* in a stream of Twitter messages we must separate them from *all* non-event content, including the endogenous trends that exhibit similar temporal behavior. In this chapter, we present techniques for identifying trending events in social media by *automatically* distinguishing them from the abundant non-event content. Specifically, we identify each trending event—and its associated Twitter messages—using an online clustering technique that groups together topically similar tweets. We then compute revealing features for each cluster, to determine which clusters correspond to events. Importantly, we design features that capture many aspects of each cluster, beyond bursty temporal patterns, to distinguish between trending events and non-event trends.

In summary, the contributions presented in this chapter are:

- We propose a general online clustering framework, suitable for large-scale social media content, which employs a post-clustering classification step to identify trending event content
- We identify revealing cluster features, to learn event classification models
- We validate the effectiveness of our techniques using a dataset of over 2.6 million Twitter messages

The clustering framework that we propose in this chapter is posed as a general framework for the unknown identification scenario that could be customized to handle content from a variety of social media sites. In this chapter, we tailor this framework to handle content from Twitter, for reasons outlined above. However, in Chapter 5 we dive deeper into the customizable aspects of this framework, particularly the clustering similarity metric, and discuss its optimization with respect to social media content from sites such as Flickr, where documents include a diverse set of revealing context features.

We begin by outlining our general clustering framework (Section 4.1) and then proceed to describe its use for unknown identification of trending events on Twitter (Section 4.2). The bulk of this chapter appeared in [BNG11a].

4.1 Clustering Framework

We cast the problem of identifying trending events and their associated social media documents as a clustering problem. Ideally, each cluster should correspond to one event and consist of all of the social media documents associated with the event. In this section, we discuss the choice of general clustering algorithm for our unknown event identification problem.

To cluster social media documents, the algorithm of choice should be scalable, to handle the large volume of data in social media sites, and not require *a priori* knowledge of the number of clusters, since social media sites are constantly evolving and growing in size. Therefore, traditional clustering approaches that require knowledge of the number of clusters, such as K-means and EM [Ber02], are not well suited for this problem. Other alternatives such as scalable graph partitioning algorithms [KAKS97] do not capture the highly skewed event distribution of social media event data due to their bias towards balanced partitioning (we experimented with graph partitioning algorithms, but do not discuss their results here because of their poor performance for our task).

Threshold-based techniques are preferable for our clustering task since they can be tuned using a training set and subsequently generalized to unseen data points. Hierarchical clustering algorithms [Ber02], while relying on threshold tuning, are also not appropriate since they require processing a fully specified similarity matrix, which does not scale to the large size of our data. Furthermore, online or incremental clustering algorithms, which are able to handle a constant stream of new documents, are also desirable in our setting, where new documents are continuously being produced.

Based on these observations, we propose using a single-pass incremental clustering algorithm with a threshold parameter that can be tuned in a principled manner during a training phase. Single-pass incremental clustering has been shown to be an effective technique for event detection in textual news documents (e.g., [APL98; YPC98]). Such a clustering algorithm considers each element in turn, and determines the suitable cluster assignment based on the element's similarity to any existing clusters. Specifically, given a threshold μ, a similarity function σ, and documents to cluster $d_1, \ldots, d_n$, the algorithm considers each document d_i in order, and computes its similarity $\sigma(d_i, c_j)$ against each existing cluster c_j, for $j = 1, \ldots, k$. (Initially, $k = 0$.) Different versions of the algorithms differ on how this similarity σ is computed, as we describe next. If there is no cluster whose similarity to d_i is greater than μ, we increment k by one and create a new cluster c_k for d_i. Otherwise, d_i is assigned to a cluster c_j with maximum $\sigma(d_i, c_j)$.

Conceptually, the similarity $\sigma(d, c)$ between a document d and a cluster c can be computed by comparing the features of d to those of the cluster c; or by directly comparing d to the documents in cluster c. For efficiency, we represent each cluster using the centroid of its documents. The centroid for a cluster of documents c is defined as $\frac{1}{|c|}\sum_{d \in c} d$. For the textual features we use in this chapter, our centroids are simply the average *tf-idf* score per term. In the next chapter, we experiment with more complex content representations and, consequently, introduce alternative centroid definitions. The similarity score $\sigma(d, c)$ is then defined as the similarity between document d and the centroid of cluster c for a suitable document similarity metric. This definition then avoids comparing document d against every document in cluster c.

The general clustering algorithm that we described relies heavily on a similarity metric σ for two documents, or for a document and a cluster centroid. For social media sites such as Twitter, where content is limited to a short textual message, we represent each message as a *tf-idf* weight vector of its textual content, and use the cosine similarity metric, as defined by Kumaran and Allan [KA04], as the clustering similarity function σ. In Chapter 5, we explore the use of this algorithm with social media documents from Flickr, which include a variety of revealing context features. In that chapter, we explore the crucial issue of learning a similarity metric using a combination of similarities, explicitly tailored to each type of context feature.

We have explored different threshold settings and other variations of this clustering algorithm, including a periodic second pass to handle fragmentation, which is a known drawback of this incremental clustering approach. In Chapter 5 we explore ways to handle such fragmentation using explicit social links to unify clusters that correspond to the same event. Additional optimizations and variations of this clustering algorithm are described in Chapter 9.

4.2 Separation of Event and non-Event Content on Twitter

Trending 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 World Wide Web 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 [NBL10].

As we discussed in Chapter 3, non-event content also includes forms of Twitter activity that trigger substantial message volume over specific time periods, which is a characteristic of trending event content. In this section, we present techniques for differentiating between messages about trending events and non-event messages, where non-event messages include those for trending activities that are Twitter-centric but do not reflect any trending events.

Formally, we define the problem we address in this section as follows (Figure 4.1):

Problem Definition 1 *Consider a time-ordered stream of Twitter messages M. At any point in time t, our goal is to identify trending 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.*

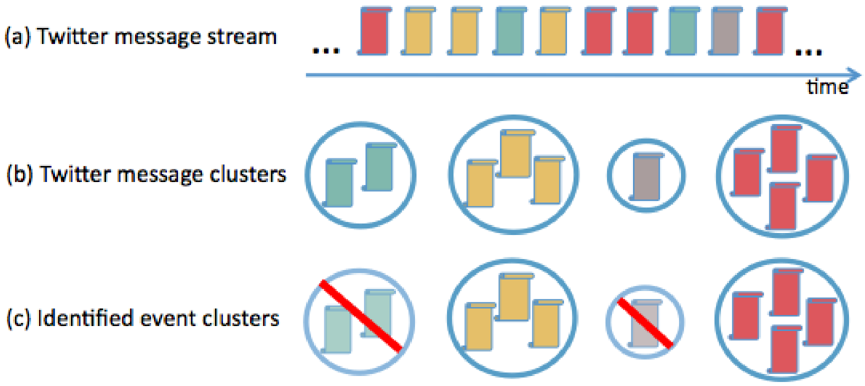

(a) Twitter message stream
...
...
time
(b) Twitter message clusters
(c) Identified event clusters

Figure 4.1: Conceptual diagram: Twitter event identification.

4.2.1 Identification of Event Clusters

We use the incremental, online clustering algorithm described above in order to effectively cluster a stream of Twitter messages in real time. For scalability, we use a centroid representation of each cluster, which consists of summary statistics of all messages in the cluster. The centroid of a cluster is the average weight of each term across all documents in the cluster. We represent each message as a *tf-idf* weight vector of its textual content, and use the cosine similarity metric [KA04], as the clustering similarity function σ. Based on our experiments on training data, we perform traditional text processing steps such as stop-word elimination and stemming, and also double the weight of hashtag terms as they are often indicative of the message content.

To identify all *trending event* clusters in the stream, we compute a variety of revealing features using statistics of the cluster messages (Section 4.2.2). 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 4.2.3) 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.

4.2.2 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. While each of these features may not necessarily indicate event content in isolation, combining them with other revealing features in a principled way (e.g., using a trained classifier) can help identify event clusters, as we will see. 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.

4.2.2.1 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 trending event in Chapter 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 the expected message volume for any frequent cluster term or a set of frequent cluster terms. Specifically, we aggregate the number of messages containing each term into hourly bins and define $M_{t,h}$ as the number of messages posted during hour h and containing term t, and M_h as the total number of messages posted during hour h.

For the n most frequent terms in the cluster, where n is determined empirically, we compute two types of features to reveal the trending behavior that is characteristic of trending events. First, we compute the deviation from expected volume for a term at the time when we compute the features (i.e., at the time when we invoke the classifier; see Section 4.2.3). This metric captures a single-point representation of trending behavior for each term. Second, we compute the quality of fit of an exponential function to the term's binned data leading up to the time when we invoke the classifier. The exponential fit captures the rate of increase in message volume over time. A good quality fit signifies a true exponential rise in related content, an indication of trending behavior [LBK09].

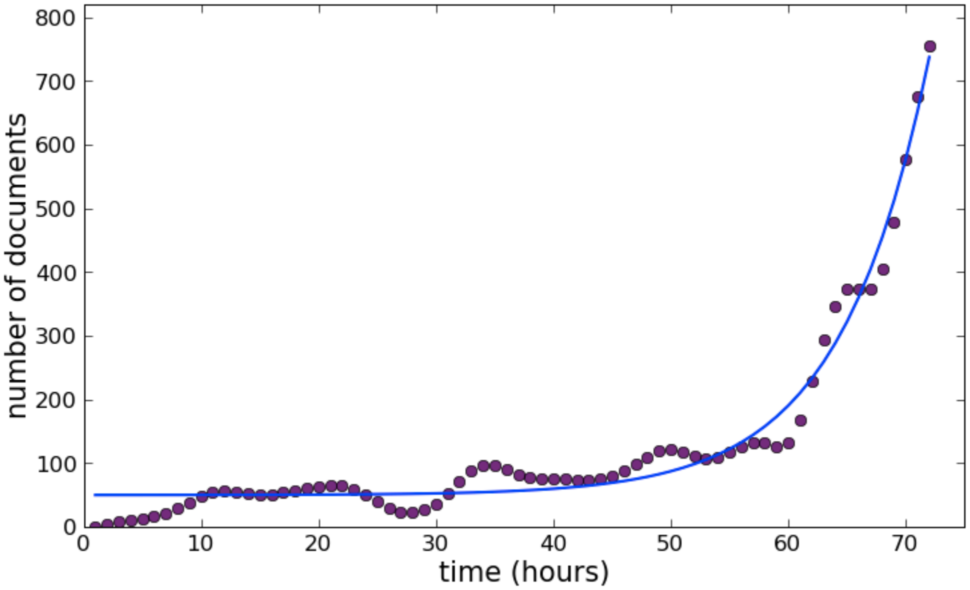


Figure 4.2: Documents per hour with the term "valentine" for 72 hours prior to 2 p.m. on Valentine's Day.

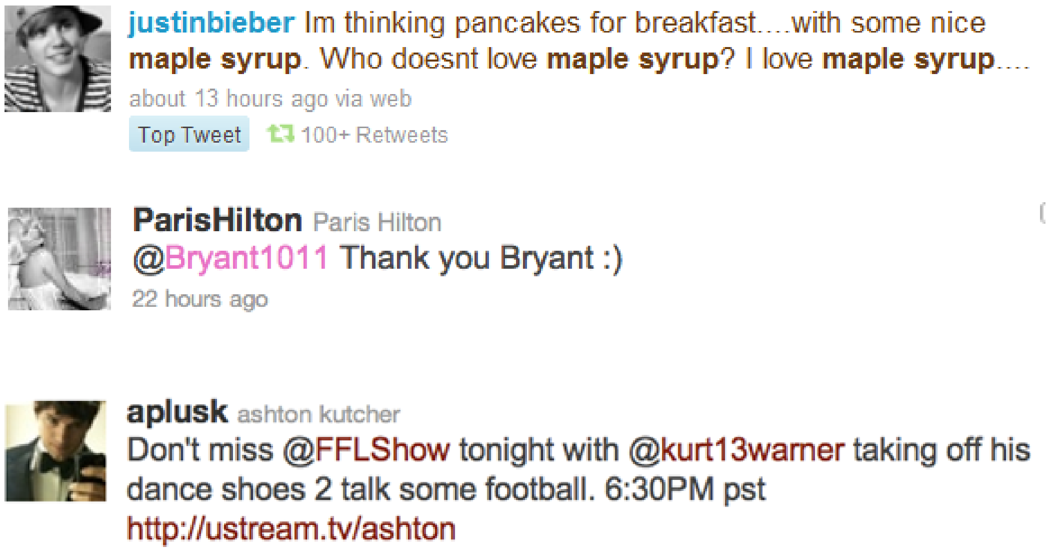
We compute the expected number of messages for a term t at the end of hour h_0 by averaging the number of messages containing t in the preceding hours $(1, \ldots, h_0 - 1)$, weighted by the total number of messages at each hour to account for the varying volume of messages across different hours; formally, $\mu_{t,h_0} = \sum_{i=1}^{h_0-1} \frac{M_{t,i}}{M_i}/(h_0 - 1)$. Correspondingly, σ_{t,h_0} is the standard deviation of the number of messages containing t over the preceding hours. We define the deviation from expected message volume for term t at hour h_0 as $(\frac{M_{t,h_0}}{M_{h_0}} - \mu_{t,h_0})/\sigma_{t,h_0}$. The deviation from expected volume features in a cluster, then, include a set of deviation values for the most frequent terms, as well as an average value over all top terms. This value is generated by weighting the top terms by their relative support in the cluster messages (e.g., if terms t_1, t_2, and t_3 appeared in 300, 200, and 100 cluster messages, respectively, their weights would be 0.5, 0.33, and 0.17).

The second set of temporal features reflects the degree to which the volume of messages containing a term t exhibits an exponential growth in the hours leading up to h_0. We compute a histogram using $M_{t,i}$, where $i = (h_0 - 72), \ldots, (h_0 - 1)$; this models the volume of messages with the term for the 72 hours leading up to h_0. This method generally reflects the trending behavior in the social Web [LBK09]. We use the least squares method to fit an exponential function to the histogram, smoothed using a moving average, and compute the R^2 statistic to measure the quality of the fit. Figure 4.2 shows an example of this exponential trending behavior for the term "valentine" around Valentine's Day, 2010.

4.2.2.2 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 (Chapter 3). As we discussed in the previous chapter, user interactions on Twitter include retweets, replies, and mentions. Our social features include the percentage of messages containing each of these types of user interaction out of all messages in a cluster.

To motivate the use of these features, consider the Twitter messages in Figure 4.3. Clusters that include a high percentage of retweets, especially of a single post by a popular Twitter user (e.g., Justin Bieber's message, retweeted over 100 times in Figure 4.3), may correspond to endogenous trends that do not contain trending event information (Chapter 3). Similarly, a high percentage of cluster messages containing replies (e.g., Paris Hilton's reply in Figure 4.3) may indicate non-event content, since when people spread event information they tend to do so via general broadcast messages rather than individual conversations. On the other hand, many celebrities, politicians, companies, venues, and shows own Twitter accounts (e.g., Ashton Kutcher's show @FFLShow with guest @kurt13warner in Figure 4.3). Therefore, unlike retweets, a high percentage of Twitter mentions of one of these entities might imply that the cluster refers to an event, where the entity is an active participant or the subject of the event.



justinbieber Im thinking pancakes for breakfast....with some nice **maple syrup**. Who doesnt love **maple syrup**? I love **maple syrup**....
about 13 hours ago via web

ParisHilton Paris Hilton
@Bryant1011 Thank you Bryant :)
22 hours ago

aplusk ashton kutcher
Don't miss @FFLShow tonight with @kurt13warner taking off his dance shoes 2 talk some football. 6:30PM pst http://ustream.tv/ashton

Figure 4.3: Examples of social interaction on Twitter.

4.2.2.3 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 characteristics of the events they describe (e.g., "Couric," "Obama," and "interview" are common among messages describing Katie Couric's interview of President Obama).

To estimate this coherence of a cluster, we compute the average or median similarity of messages to the cluster centroid using the cosine similarity metric. Additionally, we compute the percentage of messages in the cluster containing the most frequent term, the second most frequent term, and so on. Finally, we look at how many of the most frequent terms are contained in at least $n\%$ of the messages in the cluster, for empirically determined values of n.

4.2.2.4 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, endogenous trends (Chapter 3), and the trending event clusters that we wish to identify. As discussed above, Twitter-centric activities often exhibit characteristics that resemble trending events, especially as captured by temporal features, which generally offer a strong signal for the presence of trending event content. To address this challenge, we design a set of features that targets commonly occurring patterns in non-event clusters with Twitter-centric behavior.

Twitter-centric discussions often exhibit unique hashtag usage characteristics (e.g., #whenimolder tag indicating discussion on things Twitter users wish to do when they get older). We design features to capture these characteristics and differentiate the Twitter-centric activities from other non-event content and from trending events. Specifically, we compute statistics relating to tag usage, including the percentage of cluster messages that contain tags, and the percentage of cluster messages that contain the most frequently used tag. A large value of the latter serves as an indication that the messages in the cluster revolve around a tagged conversation topic.

Importantly, we also determine if the most frequently used tag is a concatenation of multiple words. Multi-word tags are highly indicative of Twitter-centric discussions that do not correspond to trending events (e.g., #firstdaterulez, #BadWrestlingNames). Unfortunately, identifying them is a challenging task since they often contain short-hand notations, acronyms, and slang that may be difficult to parse. Using a dictionary-based method for parsing the tags into several terms may be inefficient and difficult to implement due to the variety of potential terms that may be included in the tags. We have experimented with identifying these multi-word tags using such an approach with limited success. Instead, we design capitalization-based features to detect such multi-word tags: we observed that when more than one letter of a tag is capitalized by some users, and this capitalization is consistent among these users, it frequently indicates that a tag consists of multiple words. Since we do not rely on a dictionary, our approach can be applied to tweets in any language that uses capitalization rules.

4.2.3 Event Classification

Using the above features, we train an event classifier by applying standard machine learning techniques (see Section 4.2.4). This classifier predicts which clusters correspond to events at any point in time (i.e., at any point in the stream). 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 the 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, with respect to any point in the stream. Note that a temporal component is built into some of the features, and we recompute the features prior to classification, so the temporal relevance of the top selected clusters is inherently captured by our classifier.

We compare the results of our classifier against several baseline approaches next.

4.2.4 Experiments

We evaluated our event identification strategies on a large dataset of Twitter messages. We describe this dataset and report the experimental settings (Section 4.2.4.1), and then turn to the results of our experiments (Section 4.2.4.2).

4.2.4.1 Experimental Settings

Data: Our dataset consists of over 2,600,000 Twitter messages posted during February 2010. We are interested in identifying events both with local and with broad geographical interest. To ensure that our dataset substantially covers local events, we decided to collect messages posted by users of one specific location, namely, New York City (i.e., by Twitter users whose location, as entered by the users and shown on their profile, is in the New York City area)[1]. We chose this location as it consistently generated a high volume of tweets. While the location as reported by Twitter users is not always accurate, it does provide a reliable approximation [HHSC11]. Since we do not currently use location-based signals in our identification approach (a task that is reserved for future work), focusing on messages from a specific geo-location does not reduce the generality of our results. We collected these messages via a script, which continuously requested the most recent messages from the Twitter API. For each collected Twitter message, we record its textual content, the associated timestamp (i.e., the time at which the tweet was posted), and the username of the user who posted the tweet.

We cluster our dataset in an online fashion as described in Section 4.1. We use the data from the first week in February to calibrate statistics such as term frequency over time, which are needed to compute our temporal features. We then use the second week of February to train our event classifiers and baselines. Finally, we report our results on test data selected from the latter half of February (i.e., Weeks 3 and 4).

Annotations: We use human annotators to label clusters for both the training and testing phases of our event identification experiments. These annotators were instructed to label each cluster according to four different categories: event, Twitter-centric activity, other non-event, and ambiguous. To ease annotation, as a representation of each cluster, the annotators were shown the 10 most frequent terms in the cluster, along with their respective counts, and sample Twitter messages from the cluster. For clusters with more than one central theme (e.g., with top keywords "south," "park," "west," "sxsw," and "cartman," referring to either the "South Park" show or the "South by Southwest" festival), the anno-

[1] Note that events with broad geographical interest are also naturally captured in our dataset.

tators used the ambiguous label. Ambiguous clusters were not used for training, but were treated as non-events for testing.

For the training set, we randomly selected 504 clusters 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. Each cluster was labeled by two annotators, and their agreement was measured using Cohen's kappa (κ=0.683), indicating substantial agreement. 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. These five hours were sampled uniformly at random from five bins partitioned according to the volume of messages per hour over these two weeks. This sampling technique assures that we test our classifiers during hours with different volumes of messages. 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 4.2.3), and 20 random clusters, for a total of 60 clusters per hour, or 100 clusters per method over the five hours. We used two human annotators to label each cluster and achieved substantial agreement (κ=0.83). We discuss our handling of annotator disagreements on the test set in the description of our evaluation.

Training Classifiers: We train a classifier to distinguish between trending event and non-event clusters (*RW-Event*). We extracted cluster-level features for each cluster in the training set, as described in Section 4.2.2. We also used a few additional features that did not fall under the groups described in Section 4.2.2, such as the cluster size and average length of cluster tweets. We used the Weka toolkit [WF05] to train our event classifier. 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 that identifies events 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. [SST$^+$09] as part of their approach for identifying news in Twitter messages. We train this Naïve Bayes classifier using Weka, with the same methodology described above.

Evaluation: We use our annotated test set of 100 randomly selected clusters to evaluate the performance of each classifier. For this, we use the macro-averaged F_1 metric [MRS08]. 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. Here, macro-averaged F_1 is preferable to its alternative, micro-averaged F_1 [MRS08], which weighs each instance equally, causing predictions on the larger non-event class to dominate the score. In this evaluation we omit test clusters on which our annotators disagree.

In addition to classification performance, we evaluate our *RW-Event* classifier's 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. We refer to this task as "event surfacing." Since the number of clusters in the stream may be large, we only classify clusters that have over 100 messages. Similarly, we do not classify clusters that did not have newly added documents in the hour prior to the time when we invoke the classifier.

As a baseline for the event surfacing task, we consider the event thread selection approach presented by Petrović et al. [POL10], which selects the fastest-growing threads in a stream of Twitter messages and then re-ranks them based on thread entropy and unique number of users. Exploratory experiments on our training data indicated that selecting clusters based on such re-ranking strategies (i.e., selecting clusters with the highest number of unique users and entropy above a threshold) yields similar results as selecting the fastest-growing clusters. Note that the re-ranking strategies were not used to select the top clusters, which is our goal, and optimizing the selection of fastest-growing clusters that have the highest number of unique users and low entropy is reserved for future work (in fact, similar features already exist in our models). In addition to the fastest-growing clusters baseline (*Fastest*), we compare our approach against a technique that selects clusters randomly (*Random*).

To evaluate the event surfacing task, we select two standard metrics, namely, *Precision@K* and *NDCG* [CMS09], which capture the quality of ranked lists with focus on the top results. Precision@K simply reports the fraction of correctly identified events out of the top-K selected clusters, averaged over all hours. Precision@K is set-based and does not consider the relative rank of the clusters. An alternative metric that is sensitive to the rank of the events in the top selected clusters is the normalized discounted cumulative gain (NDCG) metric. We use the binary version of NDCG [CMS09], to measure how well our approach ranks the top events relative to their ideal ranking. To handle annotator disagreements in this scenario, where we need to examine ordered lists, removing the disagreements from the evaluation is not desirable given the evaluation metrics used. Instead, we penalize the *RW-Event* classifier if *either* annotator disagreed with our classifier's prediction, but only penalize the baselines if *both* annotators disagreed with their predicted label. We thus give the "benefit of the doubt" to the baselines, hence making our results more robust.

4.2.4.2 Experimental Results

We begin by examining the performance of our *RW-Event* classifier against the *NB-Text* baseline classifier on the training and test sets. The performance on the training set reflects the accuracy of each classifier computed using 10-fold cross-validation. The test performance measures how well each classification model predicts on the test set of 100 randomly selected clusters.

Table 4.1 shows the F_1 scores of the classifiers on both the training and test sets. As we can see, the *RW-Event* classifier outperformed *NB-Text* over both training and test sets, showing that it is overall more effective in predicting whether or not our clusters contain trending event information. A deeper examination of our results revealed that the *NB-Text* classifier 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 4.2.

The next set of results describes how well our *RW-Event* classifier performs for the

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

Table 4.1: F_1 score of our classifiers on validation and test sets.

Description	Terms
Senator Bayh’s retirement	bayh, evan, senate, congress, retire
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 4.2: Sample events identified by the *RW-Event* classifier.

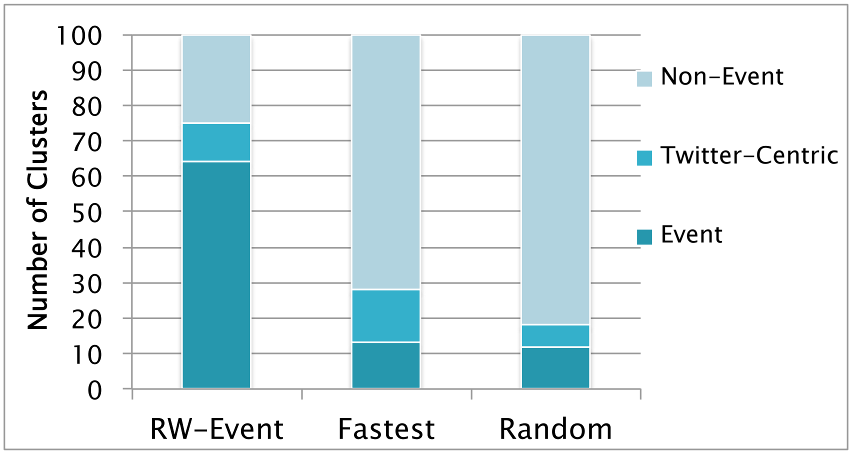


Figure 4.4: Distribution of labels for our classifier and baselines.

"event surfacing" task. Recall that the goal of this task is to identify the top events in the stream per hour. Figure 4.4 shows the distribution of labels for clusters surfaced by our classifier *RW-Event* and baselines *Fastest* and *Random* (Section 4.2.4.1) over the entire test dataset. In this figure, annotator disagreements were considered as non-event labels. As this figure clearly shows, *RW-Event* was able to identify about 5 times the number of events that each baseline technique identified. Interestingly, out of the random sample of 100 clusters, only 12% were labeled as events. Surprisingly, the number of events identified by *Fastest* was similar to the number of events identified by *Random*, implying that the growth rate of clusters is not an effective indication of event content. However, since *Fastest* identified the largest number of clusters that were labeled as "Twitter-centric," it is possible that the growth rate of clusters, to some extent, captures the trending behavior that is characteristic of trends in social media, which include both trending events and Twitter-centric endogenous trends (Chapter 3).

Since the event surfacing task aims to identify the top-K event clusters in the Twitter stream at each hour, it is important to consider the performance of our classifier and baselines with respect to the number of event clusters and relative rank of event clusters identified at each hour. For this, we report Precision@K (Figure 4.5) and NDCG@K (Figure 4.6) scores for varying K, averaged over the five hours selected for the test set. Not surprisingly, the proportion of events identified by the *Random* technique is very low, as most data on Twitter does not contain event information. The proportion of events identified by the *Fastest* technique was higher than that of *Random*. The *RW-Event* classifier performed well across the board, better than both baselines according to both precision and NDCG.

Next, we examine the performance of our classifier and baselines for each of our 5 test hours, using NDCG computed over the top 20 clusters per hour. Recall that we sampled the hours for our test set such that the volume of messages per hour varies (Section 4.2.4.1). Figure 4.7 shows the hourly performance of the alternative event surfacing techniques, for hours ordered according to increasing message volume. While our classifier still outperforms the baselines at each hour, this figure reveals several interesting points. First, even though the performance of all techniques was most similar during the highest-volume hour (i.e.,

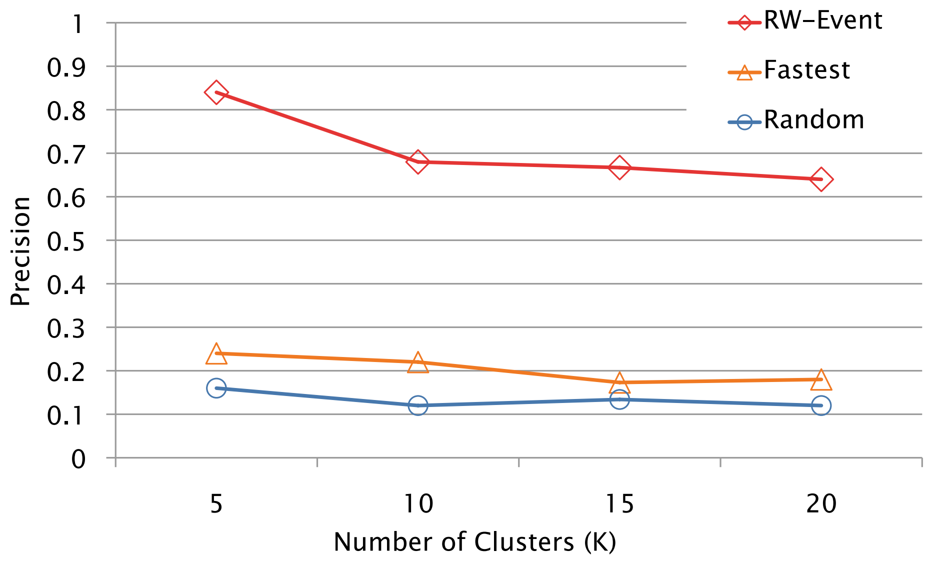


Figure 4.5: Precision@K for our classifier and baselines.

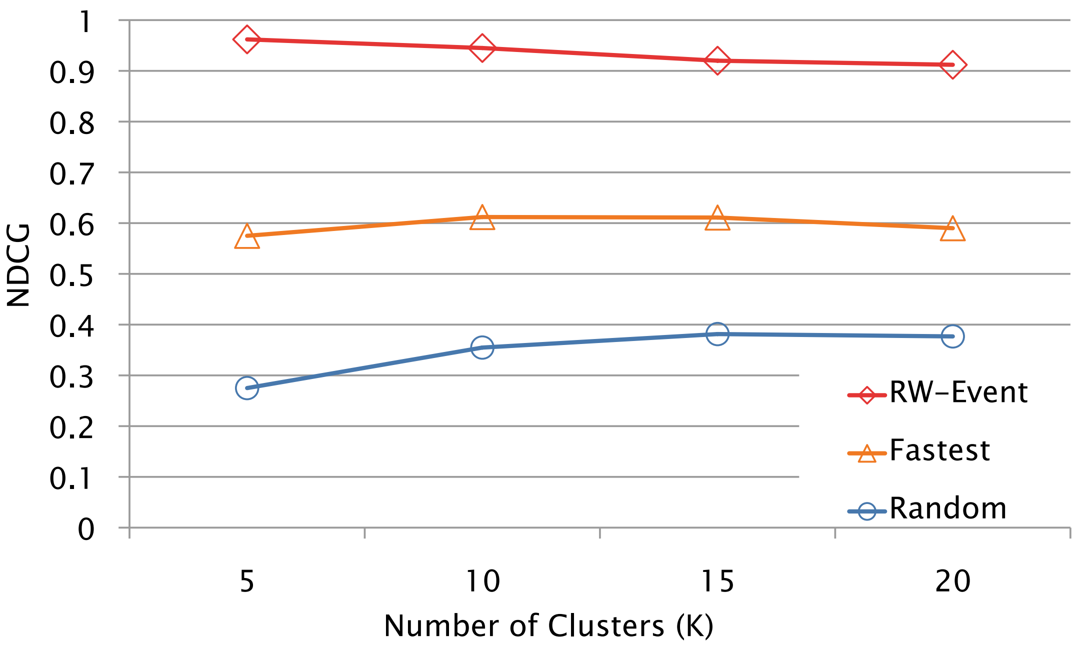


Figure 4.6: NDCG@K for our classifier and baselines.

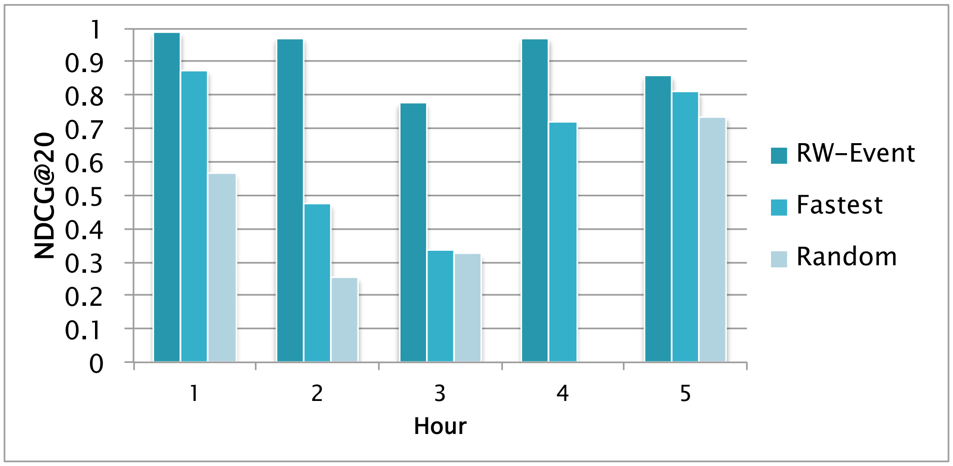


Figure 4.7: NDCG@20 of our classifier and baselines for each hour over the test set.

Hour 5), there is no indication that the hourly volume of messages effects the performance of either *RW-Event* or the baselines. Second, *RW-Event* and *Fastest* had the worst performance on the same hour (i.e., Hour 3), possibly indicating that there was less “fresh” event content during that hour, since, unlike *Random*, both of these techniques only consider clusters that are temporally relevant (e.g., as indicated by the number or percentage of new documents added to the cluster in recent hours). Since the temporal features we defined earlier were, not surprisingly, strongly correlated with the presence of event content in a cluster, reduced trending behavior for event clusters during a particular hour could explain this slight drop in performance.

Overall, Figure 4.7 shows that our classifier was very successful during three of the test hours (i.e., Hours 1, 2, and 4), and less successful during two of the test hours (i.e., Hours 3 and 5). Interestingly, these less successful test hours were also chronologically later than the hours on which *RW-Event* showed the best performance. Still, there are many factors that might contribute to this variation in performance, such as the availability of new event content, as discussed above. As another example, for Hour 5, there were several Twitter-centric clusters that were mistakingly identified by *RW-Event* as events. For some of these clusters, it was especially difficult to automatically distinguish the endogenous Twitter-centric characteristics from those of events (e.g., with single-term hashtags like #haiku).

Examining the general mistakes made by the *RW-Event* classifier, the most prominent misclassification occurs in cases where a Twitter user (usually a company or service) posts messages on a broad topic (e.g., job listings with tags such as #jobs, #nycjobs) using multiple Twitter accounts and a similar message "template," complete with hashtags. A possible reason for this behavior is that features of our model such as the number of messages from the top author were not adequately captured in the training process. Since we selected training data by sampling from the fastest-growing clusters per hour, many of our training examples did not exhibit this behavior and, therefore, we were not able to properly model it. We plan to explore this behavior further in future work.

4.3 Conclusions

In this chapter, we presented an end-to-end approach for unknown identification of trending events on Twitter. Specifically, we explored approaches for analyzing a stream of Twitter messages to distinguish between messages about trending events and non-event messages. We proposed a general clustering framework for the unknown event identification task, and customized it to group together topically similar Twitter messages. We then defined a rich family of revealing cluster-based features, including temporal, social, and topical features. Importantly, our insights into trending event behavior from Chapter 3 helped us identify a set of Twitter-centric features, which was used along with the other features to train state-of-the-art classification models, which, in turn were used to identify events and their associated Twitter messages. Our techniques offer a significant improvement over baseline and existing approaches, showing that we can identify trending event content in real time over a large-scale stream of social media content. We thus help unveil important information from, and about, real-world events as they are reflected through the eyes of hundreds of millions of users of Twitter and similar social media sites.

Additional social media sites such as Flickr and YouTube often also contain timely, unknown content related to events. These sites, unlike Twitter, have a set of associated context features (e.g., title, description, location) that can provide complementary cues for determining when social media documents correspond to the same event. In the next chapter, we explore ways to use these context features in concert, to learn a similarity metric for the clustering framework that we introduced in this chapter. Additionally, we propose ways to improve our clustering framework, using explicit social links (e.g., user comments) to connect fragmented event clusters.

Chapter 5

Similarity Metric Learning for Identification of Unknown Events

With the increasing use of cameras on mobile devices, photo- and video-sharing sites (e.g., Flickr, YouTube) are gradually becoming a valuable source of event information captured during trending events (e.g., natural disasters, political riots) [LPS+08]. While event documents on these sites generally lag behind their counterparts on social media sites such as Twitter, their accompanying multimedia content adds a useful dimension, as evidenced by the growing use of such content in mainstream media reports [LPS+08]. Such event documents exhibit opportunities for unknown event identification due to their wealth of associated context features, including user-provided annotations (e.g., title, tags), and automatically generated information (e.g., upload or content creation time). Individual features might be noisy or unreliable, but collectively they provide revealing information about events, and this information is valuable to address the unknown event identification problem.

In this chapter, instead of using a simple text-based document similarity metric for our clustering framework as we did in Chapter 4, we propose using a variety of context features in concert, to determine if two social media documents (or a social media document and a cluster) correspond to the same event. Specifically, in Section 5.1 we explore distinctive representations of social media documents and define appropriate similarity metrics for each document representation. We then develop a variety of techniques for combining these different similarities into a single metric for our clustering framework. We experiment with ensemble-based and classification-based similarity learning techniques, and use them in conjunction with a scalable, online clustering algorithm, to generate a clustering solution where each cluster corresponds to an event and includes the social media documents associated with the event.

Beyond context features, in Section 5.2 we explore the use of social links (e.g., comment and authorship connections) for enhancing the results of our cluster-based event identification approach. To understand the potential benefits of using social links for this task, we analyze a network of author comments associated with photographs in a large-scale Flickr data set. Our exploratory experiments, building on the results of Section 5.1, suggest that social links can provide a useful indication of document similarity for event identification.

In summary, the contributions presented in this chapter are:

- We develop several techniques for learning a combination of the feature-specific similarity metrics, and use them to indicate social media document similarity for our clustering framework

- We explore ways to use explicit social links between social media documents to improve the quality of the clusters produced by our framework

- We evaluate our alternative similarity metric learning techniques and social-link-based cluster merging techniques on two real-world datasets of social media event content from Flickr

In this chapter, we focus on learning multi-feature similarity metrics that can be used in conjunction with our proposed clustering framework (Section 4.1). To tailor this similarity metric to event content, our techniques rely on an assumption that the stream of social media documents to cluster contains only event documents. Techniques for filtering non-event content from a stream of social media documents have been previously developed [POL10; SST+09], using text classification approaches [POL10; SST+09] or documents produced by users who are classified as "seeders" of event information [SST+09]. An interesting direction for future work is to develop our own pre-clustering non-event filtering approach or use the multi-feature similarity metric in our end-to-end unknown event identification approach described in Chapter 4.

We proceed to describe several social media document representations using a variety of context features (Section 5.1.1) that we can incorporate into our clustering framework (Section 5.1.2), and propose alternative models for combining these similarities into a single clustering similarity metric (Sections 5.1.3 and 5.1.4). We evaluate our techniques on large-scale datasets of Flickr images (Section 5.1.5), and then explore ways to improve our clustering results using social links (Section 5.2). The bulk of this chapter appeared in [BNG10; BXNG10].

5.1 Learning Similarity Metrics for Clustering

Given a set of social media documents associated with events, the problem that we address in this chapter is how to identify the events that are reflected in the documents (e.g., President Obama's inauguration, or Madonna's October 6, 2008 concert in Madison Square Garden), and to correctly assign the documents that correspond to each event. As in the previous chapter, we cast our problem as a clustering problem over social media documents (e.g., photographs, videos), but with the particular requirement that each document includes a *variety* of "context features" with information about the document. Some of these features (e.g., title, description, tags) are manually provided by users, while other features (e.g., upload or content creation time) are automatically generated. Formally, we define the problem we address in this chapter as follows:

Problem Definition 2 *Consider a set of social media documents where each document is associated with an (unknown) event. Our goal is to partition this set of documents into clusters such that each cluster corresponds to all documents that are associated with one event.*

As we discussed, our setting assumes a stream of social media documents such that each document contains event information. However, we do not have any knowledge of the events that exist in the stream, or which social media document corresponds to which event.

For this, we employ our clustering framework (Section 4.1) with alternative multi-feature similarity metrics that we propose, learn, and evaluate in this section.

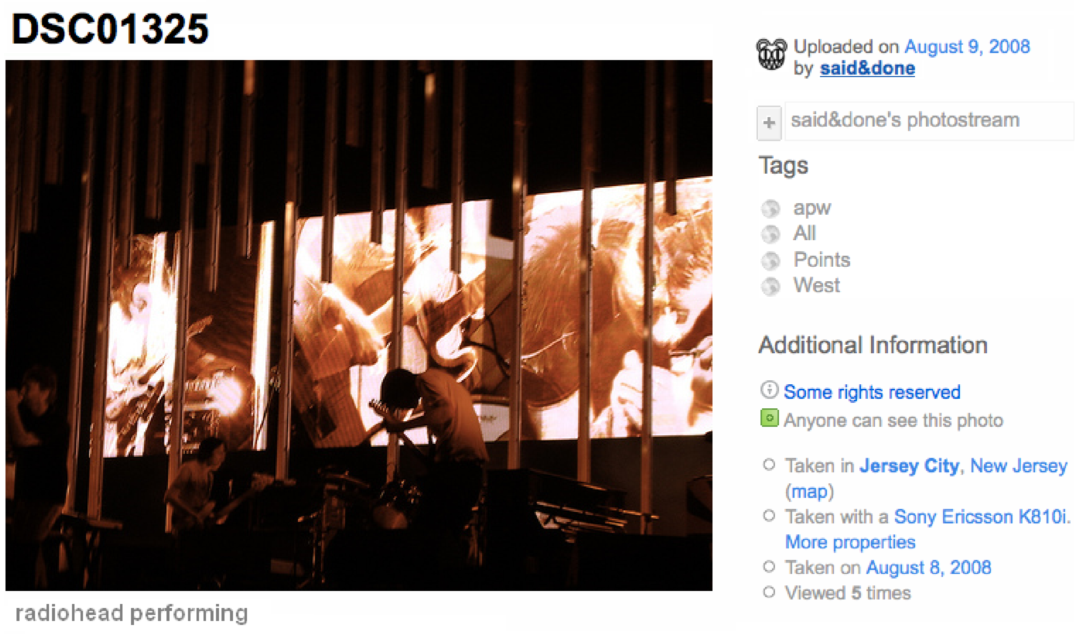
5.1.1 Social Media Document Representations

As a distinctive characteristic, social media documents, particularly from digital-media-sharing sites such as Flickr, include a variety of *context features* that are dependent on the type of document (e.g., a "duration" feature is meaningful for videos but not photographs). However, many such sites share a core set of features. These features include: *author*, with an identifier of the user who created the document (e.g.,"said&done" is the author of the photograph in Figure 5.1); *title*, with the "name" of the document (e.g., "DSC01325" in Figure 5.1); *description*, with a short paragraph summarizing the document contents (e.g., "radiohead performing" in Figure 5.1); *tags*, with a set of keywords describing the document contents (e.g., "apw, All, Points, West" in Figure 5.1); *time/date*, with the time and date when the document was published (e.g., August 9, 2008 in Figure 5.1);[1] *location*, with the location associated with the document (e.g., Jersey City, New Jersey in Figure 5.1). These context features, collectively, will prove helpful for capturing social media document similarity and, in turn, for identifying events and their associated documents, as we discuss next.

The context features of social media documents provide complementary cues for deciding when documents correspond to the same event. Individual features are often insufficient for this purpose, and all features collectively provide more reliable evidence. For example, the description of two images associated with the same event (e.g., the "All Points West" music festival) might be ambiguous or not very revealing (e.g., the description might read "my favorite band in concert" and "radiohead in concert"); but the images' time/date and location (e.g., August 8, 2008, Liberty State Park, New Jersey) provide strong evidence that they are likely to be about the same event.

In this section, we consider social media document representations using each individual feature, according to its type (e.g., textual or time data). In addition, we use one textual

[1] Often documents include their capture or creation time (e.g., capture time/date, August 8, 2008 in Figure 5.1).



DSC01325

radiohead performing

Uploaded on August 9, 2008
by said&done

said&done's photostream

Tags

- apw
- All
- Points
- West

Additional Information

Some rights reserved
Anyone can see this photo

- Taken in Jersey City, New Jersey (map)
- Taken with a Sony Ericsson K810i. More properties
- Taken on August 8, 2008
- Viewed 5 times

Figure 5.1: A Flickr photograph associated with the "All Points West" music festival event.

document representation that contains the textual representations of all the document features (title, description, tags, time/date and location). This representation, *all-text*, is commonly used in similar domains [MAMS04] and similar to the representation we used in Chapter 4 (although we did not use a textual representation of the time/date feature).

Next, we list the key *types* of features we extract from social media documents, and define individual similarity metrics for these feature types. It is possible, of course, to cluster the documents by using individual features according to an appropriate similarity metric. Such a clustering approach is not ideal, since it does not exploit the wealth of context features collectively; instead, the rest of this section describes strategies to consider the similarity metrics in concert.

Textual features: To exploit the various context features for our clustering task, we define a similarity metric for each feature, in a way that is appropriate for the feature's domain. Specifically, we represent each textual feature (e.g., title, description, tags) as a *tf-idf* weight vector and use the cosine similarity metric, as defined in [KA04], as the feature similarity metric. We considered alternative *tf-idf* formulas such as Okapi [RW99]; however, they did not perform as well, so we do not discuss them further.

In addition, we considered traditional text processing steps such as stop-word elimination and stemming, and examined the effect of each of these with respect to the individual textual features. Instead of applying the same text processing treatment to all features, we conjectured that only some features would benefit from stemming or stop-word elimination. For instance, since tag keywords are meant to be a select set of descriptive keywords for the contents of the social media document, stop-word removal may not be appropriate (e.g., removing the tag “All” in our “All Points West” example). We empirically determined the appropriate stemming and stop-word settings for each textual feature (see Section 5.1.5.1).

Time/date: For time/date, an important feature in social media documents, we represent values as the number of minutes elapsed since the Unix epoch (i.e., since January 1, 1970) and compute the similarity of two time/date values t_1 and t_2 as follows: if t_1 and t_2 are more than one year apart, we define their similarity as 0 (it is unlikely that the corresponding documents are associated with the same event in this case); otherwise, we define their similarity as $1 - \frac{|t_1 - t_2|}{y}$, where y is the number of minutes in a year.

Location: For location metadata associated with social media documents, we represent values as geographical coordinates (i.e., latitude-longitude pairs) and compute the similarity of two locations $\mathcal{L}_1 = (lat_1, long_1)$ and $\mathcal{L}_2 = (lat_2, long_2)$ as $1 - H(\mathcal{L}_1, \mathcal{L}_2)$, where $H(.)$ is the Haversine distance [Sin84], an accepted metric for geographical distance.

5.1.2 Clustering Quality Metrics and Parameter Settings

Recall that our clustering algorithm presented in Section 4.1 relies on two input parameters, a threshold μ and a similarity metric $\sigma(d, c)$. As we discussed, the similarity $\sigma(d, c)$ between a document d and a cluster c can be computed by comparing the features of d to those of the cluster c; or by directly comparing d to the documents in cluster c. In this chapter, we propose methods that use both approaches. In Section 5.1.3.2, we describe a simple similarity approach, comparing d to every document in the cluster c, and define $\sigma(d, c)$ as the average similarity score, for a suitable document similarity metric. In other words, we can define $\sigma(d, c) = \sum_{d' \in c} \frac{\sigma(d, d')}{|c|}$. This approach is not efficient because it requires comparing document d against every document in cluster c.

A more efficient approach is to represent each cluster using the centroid of its documents. Depending on the document representation we use (see Section 5.1.1), our centroids can be the average *tf-idf* score per term (for textual document features such as title, description, tags), the average time in minutes (for time/date), or the geographic mid-point (for location) of all documents in c. We use the centroid similarity approach in the majority of our techniques, described in detail in Sections 5.1.3.3 and 5.1.4.

To tune the clustering threshold for a specific dataset, we run the clustering algorithm on a subset of labeled training data. We evaluate the algorithm's performance on the training data using a range of thresholds, and identify the threshold setting that yields the highest-quality solution according to a given clustering quality metric. Although several clustering quality metrics exist (see [AGAV08]), we focus on Normalized Mutual Information (NMI) [MRS08; SGC02] and B-Cubed [AGAV08]. Both NMI and B-Cubed balance our desired clustering properties: maximizing the homogeneity of events within each cluster, and minimizing the number of clusters that documents for each event are spread across.

NMI is an information-theoretic metric that was originally proposed as the objective function for cluster ensembles [SGC02]. NMI measures how much information is shared between actual "ground truth" events, each with an associated document set, and the clustering assignment. Specifically, for a set of clusters $C = \{c_1, \dots, c_J\}$ and events $E = \{e_1, \dots e_K\}$, where each c_j and e_k is a set of documents, and n is the total number of documents, $NMI(C, E) = \frac{I(C,E)}{(H(C)+H(E))/2}$, where $I(C, E) = \sum_k \sum_j \frac{|e_k \cap c_j|}{n} \log \frac{n \cdot |e_k \cap c_j|}{|e_k| \cdot |c_j|}$, $H(C) = -\sum_j \frac{|c_j|}{n} \log \frac{|c_j|}{n}$, and $H(E) = -\sum_k \frac{|e_k|}{n} \log \frac{|e_k|}{n}$. NMI can be interpreted as the harmonic mean of cluster homogeneity and completeness, as defined by Rosenberg and Hirschberg [RH07]. We present a proof of this claim in Appendix A.

B-Cubed estimates the precision and recall associated with each document in the dataset individually, and then uses the average precision P_b and average recall R_b values for the dataset to compute $B\text{-}Cubed = \frac{2 \cdot P_b \cdot R_b}{P_b + R_b}$. For each document, precision is defined as the proportion of items in the document's cluster that correspond to the same event, and recall is defined as the proportion of documents that correspond to the same event, which are also in the document's cluster.

As we mentioned, the choice of clustering quality metric serves an important role in our clustering approach since it is used to tune the threshold parameter μ. Although NMI and B-Cubed capture the clustering properties that we are interested in, it is not always the case that the best threshold setting according to NMI is also the best setting according to B-Cubed. In order to select the threshold setting that optimizes both metrics, we use a single aggregate objective function, averaging NMI and B-Cubed. The threshold setting that yields the highest average NMI and B-Cubed value is considered Pareto optimal [Diw03], meaning that we cannot find a threshold with higher NMI value that does not have a lower B-Cubed value and vice versa.

5.1.3 Ensemble-based Similarity

Our first attempt at learning a similarity metric using the wealth of context features present in social media documents involves an ensemble algorithm, which considers each feature as a weak indication of social media document similarity, and combines all features using a weighted similarity consensus function. Ensemble clustering is an approach that combines multiple clustering solutions for a document set [DAR09; GMT05; SGC02]. The advantage of using an ensemble approach is its ability to account for different similarity metrics during the clustering process, by learning their optimal weighted contribution to the final clustering decision. In this section, we discuss ensemble clustering and show how we use it in conjunction with our clustering framework from Section 4.1 to learn a similarity metric for social media documents.

5.1.3.1 Training a Cluster Ensemble

The first step in any ensemble clustering approach is to select techniques for partitioning the data. These techniques, also referred to as *clusterers* ($C_1, \ldots, C_m$ in Figure 5.2(b)), produce mappings from documents to clusters. Each of these techniques should have a unique view of the data ($R_1, \ldots, R_m$ in Figure 5.2(a)), or use a different underlying model to generate the data partitions. For our ensemble, we select clusterers that partition the data using the different social media features and appropriate similarity metrics discussed in Section 5.1.1. In particular, we have separate clusterers for features such as title, description, tags, location, and time. Following the logic of Section 4.1, we use the single-pass incremental clustering algorithm for each feature individually, with its respective similarity metric from Section 5.1.1, as the clustering similarity function σ. We tune the threshold μ for each clusterer on a set of training data, and select the best threshold based on each clusterer's performance according to NMI and B-Cubed (see Section 5.1.2). This results in clusterers $C_1, \ldots, C_m$ (Figure 5.2(b)).

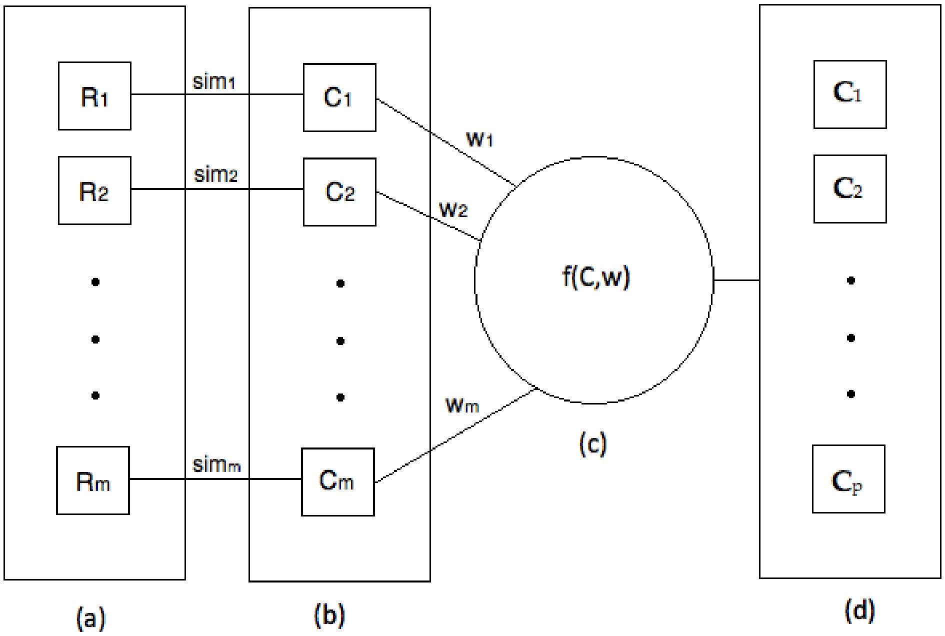


Figure 5.2: A conceptual diagram of an ensemble clustering process.

The clustering quality metrics described in Section 5.1.2 serve two important purposes in our ensemble approach. The first, as previously mentioned, is to select the most suitable threshold setting for each clusterer. The second is to assign a weight to each clusterer, indicating our confidence in its predictions. The weights are assigned during a supervised training phase, and used to determine each clusterer's influence on the overall ensemble similarity assignment. By assigning a weight to a clusterer, we indicate how successful the clusterer was in capturing document similarity on a training set, and, therefore, how likely it is to correctly indicate the similarity of unseen document pairs.

Once we select the best performing thresholds for all clusterers $C_1, \ldots, C_m$, we set their weights $w_1, \ldots, w_m$ to equal their respective combined NMI and B-Cubed scores (see Section

5.1.2), and then normalize the ensemble weights such that $\sum_{i=1}^{m} w_i = 1$. In the conclusion of the ensemble training phase, we have learned an optimal threshold for each clusterer, as well as a quality measure that will be used to weigh its decisions. With this information, we can proceed in two distinct ways: the first is to combine individual clusterer partitions as in the traditional ensemble clustering setting (Section 5.1.3.2), and the second is to use the learned weights and thresholds as a model for the similarity metric, without further influence from the individual clusterers (Section 5.1.3.3). We elaborate on these approaches next.

5.1.3.2 Combining Individual Partitions

The first ensemble-based approach for learning a similarity metric follows the traditional cluster ensemble framework [SGC02] that utilizes individual clusterers' similarity judgments on document pairs. Given a set of documents, we use each clusterer with its learned threshold to generate a clustering partition. Our challenge is to develop a consensus mechanism for combining these individual partitions into one clustering solution ($\mathcal{C}_1, \ldots, \mathcal{C}_p$ in Figure 5.2(d)). The consensus that our algorithm reaches using the clusterers' similarity judgments is translated into a similarity metric σ that can be used in our general clustering framework (Section 4.1).

Intuitively, each clusterer can be regarded as providing an expert vote on whether two documents belong in the same cluster. The consensus function we use is a weighted binary vote: for a pair of documents (d_i, d_j) and clusterer C, we define a prediction function $P_C(d_i, d_j)$ as equal to 1, if d_i and d_j are in the same cluster, or 0 otherwise[2]. Then, we compute the consensus score for d_i and d_j as $\sum_C P_C(d_i, d_j) \cdot w_C$, where w_C is the weight of clusterer C. For example, consider a simple ensemble with three clusterers $C_{time}, C_{location}$, and C_{tags}, whose weights are $0.25, 0.35$, and 0.4, respectively. To determine whether two documents d_i and d_j belong in the same cluster, we compute their prediction $P_{C_i}(d_i, d_j)$, for $i = time, location$, and $tags$. Suppose that C_{time} and $C_{location}$ cluster d_i and d_j together, but C_{tags} does not. The consensus score for d_i and d_j is then 0.25+0.35=0.6.

Note that our general single-pass incremental clustering algorithm has to compare each

[2] Similarly, we can use the raw similarity score.

document to existing *clusters* at every step. However, in the cluster ensemble formulation we can only obtain the clusterers' similarity judgments for *document* pairs. Therefore, in order to measure the similarity of a document to a cluster, we compare the document against all documents in the cluster using the ensemble consensus function, and use the average consensus score as our similarity metric σ for this document-cluster pair.

Learning a similarity metric using this ensemble approach yields a simple model, which uses a weighted combination of the data partitions obtained by clustering according to each feature and corresponding similarity metric from Section 5.1.1. While this approach provides an intuitive solution that models the contribution of each feature-specific similarity in a clustering context, one of its main drawbacks is its best-case quadratic running time in the size of the dataset. Therefore, in the next section we consider a modified approach that still uses the knowledge from the ensemble training phase to combine the similarity metrics, while at the same time improves efficiency with a centroid-based similarity technique.

5.1.3.3 Combining Individual Similarities

The second ensemble-based technique for learning a similarity metric uses the threshold and weight assignment learned in the ensemble training phase (Section 5.1.3.1) as the only input from the clusterers. Instead of computing the consensus score using the clusterers' predictions, we now compute the documents' feature-specific similarity metrics directly for documents and cluster centroids. The advantages of this modification to the ensemble similarity learning technique include improved efficiency via the use of centroids, providing for a more direct similarity metric computation.

To compute a similarity between a document d_i and a cluster centroid c_j, we repeat the same decision procedure for the similarity of document pairs, described above, using the weight and threshold that we learned for each individual feature. For similarity metric σ_C, threshold μ_C, and weight w_C associated with a clusterer C, we define $P_C(d_i, c_j) = 1$ if $\sigma_C(d_i, c_j) > \mu_C$, and 0 otherwise, and compute the combined similarity metric $\sum_C P_C(d_i, c_j) \cdot w_C$. Note that while this formulation of the similarity function uses a weighted binary vote for each feature, we could alternatively use the raw similarity score, as we suggest in the next section.

Note that we can now use the one-pass incremental clustering algorithm with centroid similarity. Depending on the document representation, the centroid is either the average *tf-idf* score per term (for textual features such as title, description, tags), the average time in minutes (for time/date), or the geographic mid-point (for location). Centroids can be updated and maintained with little cost using the general framework described in Section 4.1.

5.1.4 Classification-based Similarity

In this section, we use classification models to learn document similarity functions for social media, as an alternative to the ensemble-based approach. In other words, we use a classifier with similarity scores as features to predict whether a pair of documents belongs to the same event. Formally, given a pair of social media documents d_i and d_j, we compute the raw similarity scores $\sigma_1(d_i, d_j), \ldots, \sigma_m(d_i, d_j)$, corresponding to the document features and individual similarity metrics defined in Section 5.1.1. Using this formulation of the problem, we are able to utilize a variety of state-of-the-art classification algorithms for learning the combined similarity metric σ for our general clustering framework.

Before we can train a similarity metric classifier, we must decide whether to model similarity between document pairs, or documents-centroid pairs. Although we are interested in learning a similarity metric that would indicate when social media documents correspond to the same event, in our clustering framework we compare documents to cluster centroids. Therefore, we consider the alternative of training the classifiers on document-centroid pairs, which more closely resembles the data that the classifier will be predicting on.

Intuitively, modeling the similarity between documents and centroids would be more robust than modeling similarities between document pairs. For example, consider a pair of documents that does not share any tag keywords, yet relates to the same event. Having this pair as a positive example (i.e., the documents are about the same event) provides a false indication that tag keywords do not contribute towards a positive prediction. For centroids, since we aggregate and average the *tf-idf* values of multiple documents, there exists a better chance to capture some overlapping tag vocabulary and, therefore, to more accurately gauge the contribution of tag keywords to the overall similarity metric.

One key challenge for the classification-based approach involves the selection of training examples from which to learn the similarity classifiers. Ideally, we want our model to correctly predict the similarity of every document to every other document (or every centroid, based on the modeling choice described above) in the dataset. However, creating a training example for each document (or document-centroid) pair results in a skewed label distribution, since a large majority of pairs in the training dataset do not belong to the same event. Using a classifier trained with a skewed label distribution as a similarity metric for clustering yields poor clustering solutions since this classifier is much more likely to predict that two items do not belong in the same cluster, thus splitting single events across many clusters.

With this in mind, we can outline two sampling strategies to balance the label distribution. The first strategy is to take the first n documents in the training set according to their upload time, and compare them to every other document in that set. In the case of document-centroid similarities, we compare each document against all centroids, which are computed in advance for each event. To handle the skewed label distribution, we produce a random subsample of this data such that the number of positive and negative examples is balanced. We empirically found that generating a subsample that is 10% of the original sample size, with a balanced label distribution, yields a more accurate similarity metric classifier than other sampling techniques that we experimented with.

The second strategy is to select documents at random, pairing each document with one positive example, randomly selected from the set of documents that share the same event, and one negative example, randomly selected from the set of documents related to different events. For document-centroid pairs, we only have one choice for the positive example per document, but we randomly select among different event centroids for the negative document-centroid pair.

For this family of similarity metric learning techniques, we consider a variety of state-of-the-art classification algorithms, and train them using the datasets discussed in this section. We elaborate on our choice of classifiers and the training process in the next section.

5.1.5 Experiments

We evaluated our work on a large dataset of real world data from popular social media sites, with these goals:

- Examine which sampling and modeling methods, and what classification algorithms perform well for the classification-based approach.
- Determine which similarity metrics and techniques perform best for the event identification task.
- Gain insight about these approaches by analyzing the weights that the similarity metric learning approaches assign to each feature-specific similarity.

We report on the dataset and experimental settings, then turn to the results of our experiments.

5.1.5.1 Experimental Settings

Data: For our experiments, we collected two datasets of labeled event photographs from Flickr, a popular photo-sharing service, using the site's API[3]. The *Upcoming* dataset consists of all photographs that were manually tagged by users with an event id corresponding to an event from the Upcoming event database[4]. These Upcoming tags provide the "ground truth" for our clustering experiments. Each photograph corresponds to a single event, and each event is self-contained and independent of other events in the dataset. The *Upcoming* dataset contains 9,515 unique events, with an average of 28.42 photographs per event, for a total of 270,425 photographs, taken between January 1, 2006, and December 31, 2008.

Our second dataset is the *Last.fm* dataset, which consists of all Flickr photographs that were manually tagged by users with an id corresponding to an event from the Last.fm music event catalog[5]. The *Last.fm* dataset contains 24,958 unique events, with an average of 23.84

[3] http://www.flickr.com/services/api

[4] http://www.upcoming.org

[5] http://www.last.fm/events

photographs per event, for a total of 594,946 photographs, taken between January 1, 2006, and December 31, 2008.

The context features associated with each photograph include the title, description, tags, time/date of capture, and location. On average, 32.2% of the photos include location information in the form of geo-coordinates. On this subset of the data, we perform reverse geo-coding using the Flickr API, to obtain a textual representation of the location of each photo, which we use for the *all-text* feature.

Training Methodology: We train our clustering algorithms on data from the *Upcoming* dataset, and test them on unseen *Upcoming* data, as well as *Last.fm* data. We order the photographs in the *Upcoming* dataset according to their upload time, and then divide them into three equal parts. We use the earliest two thirds of the data as training and validation sets. We use the *training set* to tune the clusterer thresholds for the ensemble-based techniques and train classifiers for the classification-based techniques. We use the *validation set* to learn the weights for the ensemble and tune the threshold for the general single-pass incremental clustering algorithm. The last third of the *Upcoming* data and all of the *Last.fm* data are used as *test sets*, on which we report our results. We chose a time-based split since it best emulates real-world scenarios, where we only have access to past data with which we can train models to cluster future data. We train our similarity metrics once and for all, without adapting them as we observe more data. Dynamically modifying the similarity metrics as new documents arrive is reserved for future work.

Document Representations: The Lemur Toolkit[6] is used to index our documents according to each textual representation discussed in Section 5.1.1. These representations include *Title*, *Tags*, *Description*, and *All-Text*. We use all possible settings of stemming and stop-word elimination for each document representation, and create a separate index for every possible combination. We use the index to compute *tf-idf* vectors for each textual document representation. Finally, we create additional document representations using numeric time/date (*Time/Date-Proximity*) and location coordinates (*Location-Proximity*) as described in Section 5.1.1. If a document representation cannot be created due to missing data (e.g., an unspecified location), we assign it a similarity value of 0 to any other document

[6] http://www.lemurproject.org

for this representation.

Weighing Clusterers: For the ensemble-based approaches, we use Lemur's single-pass incremental clustering implementation to cluster the training data according to each document representation and corresponding similarity metric from Section 5.1.1. We tune the clustering threshold for each clusterer using the training set, considering thresholds in the range $[0, 1]$, with 0.05 increments. For time and location features, we apply log scaling to the similarity metric in order to perform the selection of thresholds with a finer granularity, as appropriate to those metrics. For each document representation, we select the threshold that yields the highest combined NMI and B-Cubed score (Section 5.1.2). For textual document representations, we select one threshold setting per feature and associated parameter settings (stemming and stop-word elimination). We use the best-performing setting for each textual representation when creating future document representations for that feature. The best settings for *Title* and *Description* were no stemming or stop-word elimination, while *Tags* benefited from stemming and *All-Text* from stop-word elimination.

We proceed to cluster the validation set according to each document representation and corresponding similarity metric, using the selected threshold setting for each clusterer. To determine the weight of each clusterer, we compute its combined NMI and B-Cubed scores on the validation set. Finally, we run the ensemble algorithm on the validation set using the selected clusterers, and tune the clustering threshold for the ensemble approach using NMI and B-Cubed.

Training Classifiers: To train similarity classification models (Section 5.1.4), we used the training set to construct four training samples according to the modeling and sampling strategies that we discussed in Section 5.1.4:

- TIME-DD: all possible document-document pairs from the first 500 documents ordered according to their time of creation.
- RANDOM-DD: 10,000 document-document pairs chosen randomly from all possible pairings between documents.
- TIME-DC: all possible document-centroid pairs from the first 500 documents, ordered according to their time of creation, and their corresponding centroids.

- RANDOM-DC: 10,000 document-centroid pairs chosen randomly from all possible pairings between documents and centroids.

For the document-centroid modeling approach, we computed all event centroids based on the ground truth labels.

We used the Weka toolkit [WF05] to build classifiers for all of the above training sets. We explored a variety of classifier types and selected two techniques that yielded the best overall performance in preliminary tests using the training set, although differences were not substantial. We selected support vector machines (Weka's sequential minimal optimization implementation), and logistic regression.

Comparing Techniques: We consider all individual clusterers as baseline approaches, namely, *All-Text*, *Title*, *Description*, *Tags*, *Time/Date-Proximity*, and *Location-Proximity*. We compared them against our clustering approaches using four different similarity metric learning techniques:

- ENS-PART: Ensemble-based approach, combining partitions (Section 5.1.3.2).
- ENS-SIM: Ensemble-based approach, combining similarity scores (Section 5.1.3.3).
- CLASS-SVM: Similarity classifier, using Support Vector Machines (Section 5.1.4).
- CLASS-LR: Similarity classifier, using Logistic Regression (Section 5.1.4).

To evaluate the clustering solutions of these different techniques, we use the clustering quality metrics of Section 5.1.2, namely, NMI and B-Cubed.

5.1.5.2 Experimental Results

We begin with the task of finding the best modeling and sampling strategies for the classification-based techniques, which is of course critical for the performance of these approaches. We trained a classifier using support vector machines and logistic regression for the different sampling and modeling strategies, and tested the quality of clustering results for each classifier and sampling method. The results are shown in Table 5.1, indicating that time-based sampling is consistently superior to random sampling according to both NMI and B-Cubed. Similarly, the document-centroid modeling techniques yield higher-quality

Algorithm	Sample	NMI	B-Cubed
CLASS-SVM	TIME-DC	**0.9492**	**0.8226**
CLASS-SVM	TIME-DD	0.9396	0.7868
CLASS-SVM	RANDOM-DC	0.9082	0.6954
CLASS-SVM	RANDOM-DD	0.8227	0.4180
CLASS-LR	TIME-DC	**0.9508**	**0.8258**
CLASS-LR	TIME-DD	0.9360	0.7743
CLASS-LR	RANDOM-DC	0.8991	0.6483
CLASS-LR	RANDOM-DD	0.8257	0.4360

Table 5.1: Performance of classification-based techniques using different sampling strategies over the validation set.

Algorithm	NMI	B-Cubed
All-Text	0.9240	0.7697
Tags	0.9229	0.7676
ENS-PART	0.9296	0.7819
ENS-SIM	0.9322	0.7861
CLASS-SVM	0.9425	0.8095
CLASS-LR	**0.9444**	**0.8155**

Table 5.2: Performance of all similarity metric learning techniques and the best individual clustering techniques over the *Upcoming* test set.

clustering solutions than techniques that model similarity between document pairs. Therefore, we proceed to test our classification-based techniques using classifiers trained on the time-based document-centroid training sample (TIME-DC).

Next, we compared our similarity metric learning techniques against each other, as well as against the top performing individual clusterers, on the *Upcoming* test set. Table 5.2 presents the clustering performance of all similarity metric learning techniques, as well as the *All-Text* and *Tags* clusterers, in terms of NMI and B-Cubed. Not surprisingly, the top performing *individual* clusterer is *All-Text*.

Title	Date	Location	#Docs
Street Art Photowalk	7/14/08	London	411
Cherry Blossom Festival	4/12/08	San Francisco	269
American Music Union	8/8/08	Pittsburgh	209
How Weird Street Fair	5/4/08	San Francisco	52

Table 5.3: Some events identified by CLASS-LR.

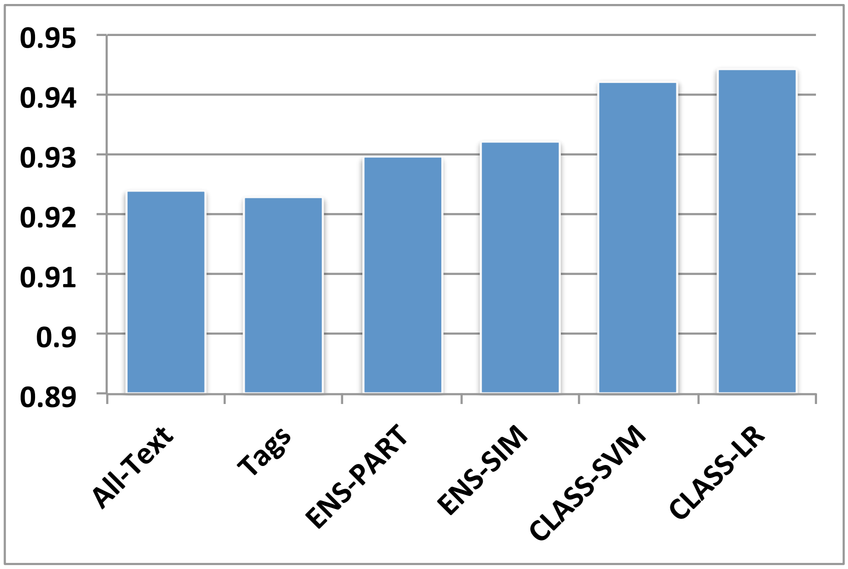


Figure 5.3: NMI scores on the *Upcoming* test dataset.

More importantly, the similarity metric combination approaches that we consider in this work outperform all individual clusterers, including *All-Text* (which also considers all document features, but with a single text-based similarity metric). Among the similarity metric learning techniques, the classification-based techniques CLASS-SVM and CLASS-LR outperform the ensemble-based techniques ENS-PART and ENS-SIM. CLASS-LR is the best overall technique in terms of both NMI and B-Cubed. The least successful of our techniques is ENS-PART, implying that learning the similarity metric directly is more effective than combining individual feature-based clustering partitions. Some events identified by CLASS-LR are shown in Table 5.3.

We also compared our techniques using the *Last.fm* dataset as an independent test

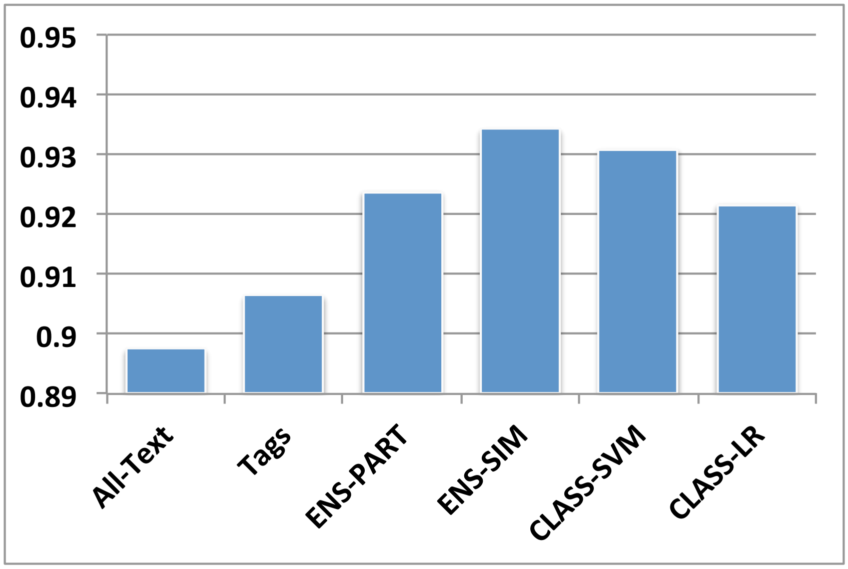


Figure 5.4: NMI scores on the *Last.fm* test dataset.

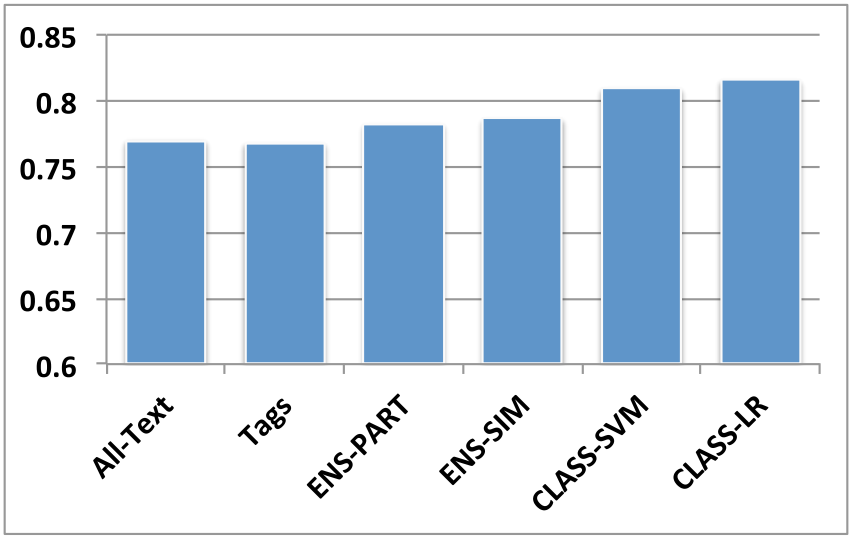


Figure 5.5: B-Cubed scores on the *Upcoming* test dataset.

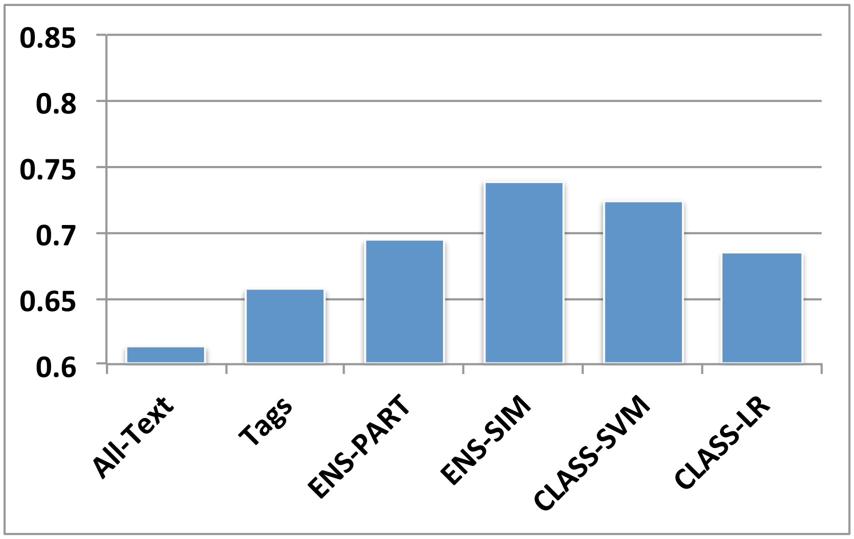


Figure 5.6: B-Cubed scores on the *Last.fm* test dataset.

set (with the training and validation set from the *Upcoming* dataset). As Figures 5.3-5.6 show, the test on the *Last.fm* dataset resulted in similar, albeit not identical, outcomes. In that test, all similarity metric learning techniques still outperform the baselines, but the top-performing technique is now ENS-SIM. Recall that the analysis of our techniques is performed over data from Flickr, with one dataset containing content annotated with events from Upcoming, and the other from Last.fm. Different properties of Last.fm events compared to Upcoming events could be the source of these relative performance differences (e.g., *Tags* similarity is better than *All-Text* for the *Last.fm* dataset), in which case ENS-SIM may be most robust in the face of these differences. Interestingly, the strong results for all methods over *Last.fm* are encouraging, as some real-world scenarios will require training on datasets different than the eventual data to be analyzed.

To determine if our results are statistically significant, we executed a set of tests by partitioning the *Upcoming* test dataset into 10 equal subsets according to document upload time, and ran each clustering technique on every subset. We discuss detailed results only for the NMI metric (while trends for B-Cubed were equivalent to trends observed for NMI, the differences between approaches as measured by B-Cubed were not as significant). We used the Friedman test [Dem06], a non-parametric statistical test for comparing a set of

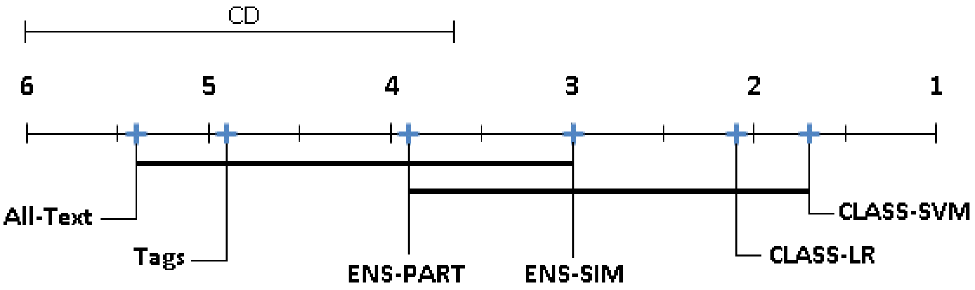


Figure 5.7: Comparison of all techniques using the Nemenyi test. Groups of techniques connected by a line are *not* significantly different at $p < 0.05$.

alternative models. The Friedman test's null hypothesis states that all the approaches have similar performance. The results of the test comparing the 10 runs show that we can reject this null hypothesis with $p < 0.05$, meaning that the performance of some approaches is significantly different.

A post-hoc statistical test is required to expose the relationship between the individual techniques. Figure 5.7 shows the results of the post-hoc analysis of our data using the Nemenyi test and the graphical representation as proposed by Demšar to visualize the relationships between the techniques [Dem06]. Techniques are plotted according to their average rank for the test datasets, and a line spans each group of techniques that is not different in a statistically significant manner. The figure demonstrates that, for the 10 tests, while CLASS-SVM and CLASS-LR are significantly better than both baseline approaches, they are not significantly different from each other, or the other similarity metric learning techniques, at the $p < 0.05$ level. For $p < 0.1$, we can claim that CLASS-SVM is also significantly better than ENS-PART.

Since NMI can be intuitively interpreted as the harmonic mean of the clusters' homogeneity and completeness scores (Appendix A), we also examine our competing approaches using these metrics. Figure 5.8 shows the homogeneity scores of our similarity metric learning techniques and baselines on the *Upcoming* test set. Interestingly, even though CLASS-LR has a higher NMI score than CLASS-SVM, CLASS-SVM produced clusters that are

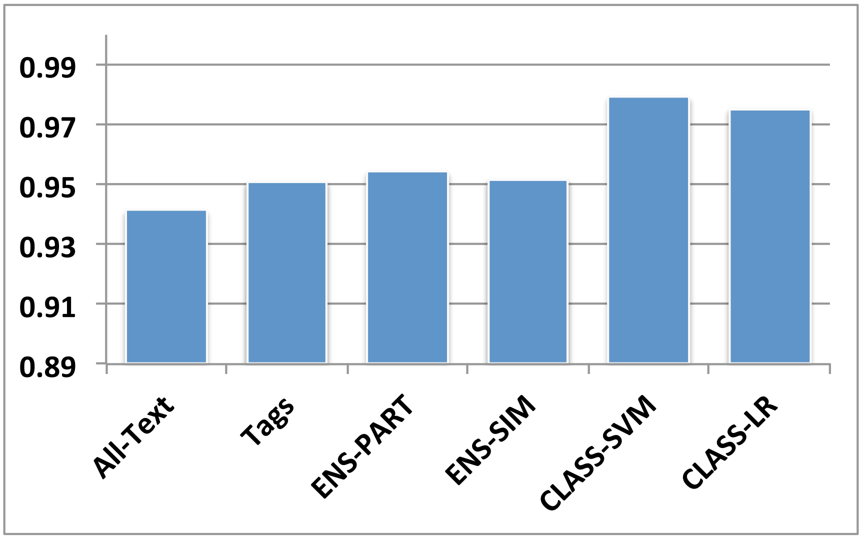


Figure 5.8: Homogeneity scores on the *Upcoming* test dataset.

more homogenous, with a larger proportion of documents in each cluster corresponding to the same event, compared to CLASS-LR. Another interesting insight is that *Tags* has a higher homogeneity score than *All-Text*, which implies that the *Tags* similarity metric is more precise but less inclusive than the *All-Text* similarity.

To complement the homogeneity results, we report the completeness scores of our similarity metric learning techniques and baselines on the *Upcoming* test set. Here, we see that the completeness score of CLASS-SVM is lower than that of CLASS-LR, which explains why the NMI score, which balances homogeneity and completeness, indicated that CLASS-LR is the best approach. Recall from our discussion of cluster quality metrics (Section 5.1.2) that homogeneity and completeness are the two properties of the clustering solution that we aim to optimize. Therefore, while CLASS-SVM may be better than CLASS-LR according to homogeneity, CLASS-LR is better in terms of completeness, and, importantly, strikes a better balance between these two properties. This is also the case for *Tags*, whose completeness score is the lowest overall, again supporting our claim that tag-based similarity is very conservative and, therefore, spreads event documents across more clusters than *All-Text*.

To gain more insight into the results of the various techniques, we analyzed the similarity

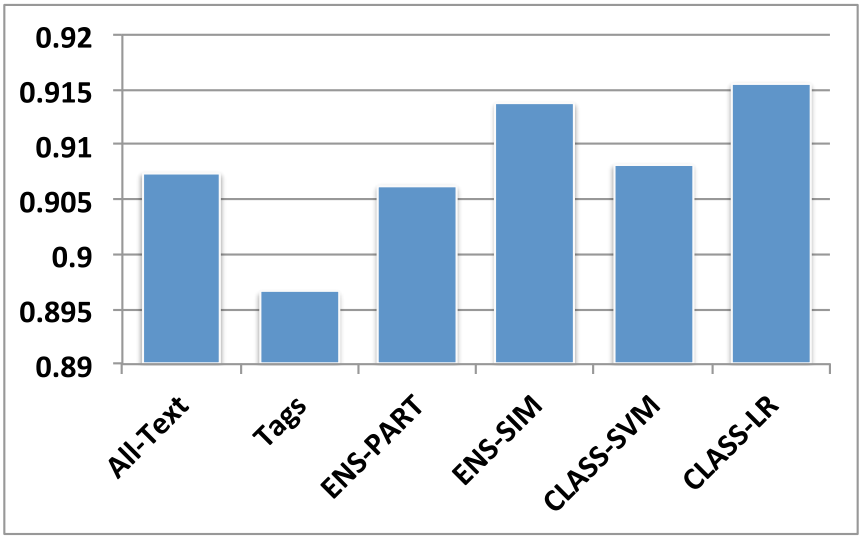


Figure 5.9: Completeness scores on the *Upcoming* test dataset.

metric models. Since the techniques use different modeling assumptions, we examined their differences in terms of the weight coefficients that they assign to each similarity feature. These coefficients, while not comparable in absolute terms, hint at the relative contribution of each similarity feature towards the model's final similarity prediction. CLASS-LR considers *All-Text* as the most important feature, followed by *Time/Date-Proximity*. CLASS-SVM, on the other hand, considers *Title*, followed by *All-Text* as the top two features. A surprising result is that both classifiers agree that, in the presence of all other features, *Location-Proximity* is an indication of document dissimilarity. In contrast, our ensemble model gives the lowest weights to *Title* and *Time/Date-Proximity*, and *Location-Proximity* has the third highest weight (after *Tags* and *All-Text*). These observations can form the basis of a more detailed analysis in the future.

5.2 Exploiting Social Links

In the previous section, we focused on learning social media document similarity metrics, and used them in conjunction with a scalable clustering algorithm. As we discussed, ideally each cluster corresponds to an event and includes the social media documents associated with the event. While this work significantly outperformed the appropriate baselines, it did not exploit the variety of social links available in social media sites. We expect social links to be useful in situations where we cannot determine if documents are similar based on their context features alone (e.g., often documents have missing location information). Therefore, we explore ways to judiciously leverage social links for event identification, to complement the similarity learning models identified in the previous section.

5.2.1 Link-based Similarity

Links such as social network connections, comments, and shared group memberships provide important cues for document similarity in social media. To understand the potential benefits of using (inherently noisy) social links for our event identification task, we analyzed the network of author comments associated with photographs in (a 90,288-document subset of) a large-scale Flickr data set (see Section 5.2.2). Out of the distinct document authors in the data set, 45% commented on some other author's document. Interestingly, 44% of authors who made such comments did so purely within one event (i.e., these authors created documents for an event and only commented on documents for that event, not others). Furthermore, 80% of authors made more comments on documents inside events on which they have published content than on documents for other events in the data set. These exploratory statistics hint that social links (e.g., based on author-comment relationships) might help in identifying event content.

We consider different ways to incorporate social links into a document similarity metric. One way is to use different types of link-based similarities on context features (e.g., author) as features for a similarity metric learning model. These similarities may be binary indicators of the authors' social network connections, shared group memberships, and so forth. In isolation, these features are not very revealing, but combined with other similarity metrics (e.g., based on the documents' context features) they may prove helpful for capturing document similarity. However, incorporating social links into the similarity models is a challenging task: for ensemble-based models (Section 5.1.3), clusterers using just link-based similarities will likely group together many documents relating to different events, and create a large number of singleton clusters where no links exist; for classification-based models, the true contribution of link-based similarity features might be difficult to capture, since social links are often sparse.

While social links between document pairs may be too weak to capture similarity, links between *clusters* of documents may be more revealing. We observed that when our clustering algorithm (Section 4.1) incorrectly splits an event across multiple clusters, it is often due to insufficient similarity between the event's documents rather than due to a strong similarity to documents from other events. As a result, many "pure" clusters, where all documents in the cluster belong to the same event, are created. In a preliminary analysis performed over clusters created by applying our algorithm to (a 90,288-document subset of) the Flickr data set (Section 5.2.2), we found that 24% of the events were split across multiple clusters, and half of these were split into "pure" clusters exclusively. These "pure" clusters represent a simple case where strong evidence of social links between two clusters could help us detect that they belong to the same event.

Therefore, we proceed to explore this alternative direction, using links between social media document clusters to learn whether they should be merged. We analyzed social links in the form of author comments on social media documents within Flickr. Out of the set of distinct authors of documents in our data, 45% commented on some other author's document. Interestingly, 44% of authors who made such comments did so purely inside events (i.e., commented on documents relating to the same events as documents they posted) and 80% of authors made more comments on documents inside events than on other event documents in the dataset. Figure 5.10 shows the authors' comments graph, where each node corresponds to an author, and a directed link from one author to another indicates that the author commented on the other's document. The documents are grouped together according to their cluster assignment. The clusters in Figure 5.10 are two "pure" clusters corresponding to the same event. The strong links between these clusters provide encouraging evidence that social links may be useful to detect social media document similarity where feature-based metrics alone are unable to.

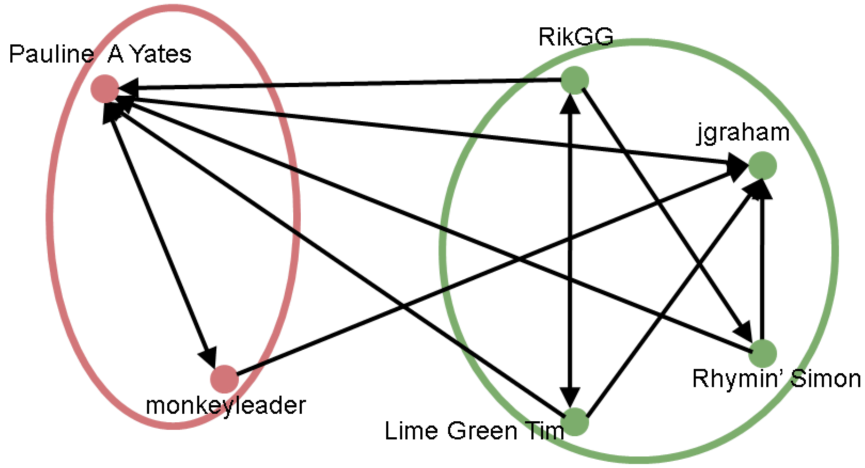


Figure 5.10: Comments between authors in two event clusters.

5.2.2 Exploratory Experiments

We describe our exploratory experiments using author-comment links associated with Flickr photographs to improve the clustering results of a classification-based similarity model based on logistic regression (CLASS-LR). We perform our experiments over the *Upcoming* dataset described in Section 5.1.5.1. Specifically, we develop our social link-based cluster merging strategies on the validation set and then report the results of the entire procedure (clustering and merging) on the test set.

To decide whether to merge any pair of clusters, we train a learning model using the comments associated with each document across "pure" cluster pairs. We consider a variety of link-based features, including the total number of comments between authors in the clusters, the number of mutual comments (i.e., author A_1 from cluster C_1 commented on the document of author A_2 in cluster C_2 and vice versa), and the percentage of shared comments out of all comments associated with each cluster in the pair.

In our learning scenario, we would like to avoid false positives: a false positive corresponds to merging clusters for different events and hence hurts the quality of the initial clustering solution on which we build. Therefore, we use a cost-sensitive classification approach, training a model that assigns the highest cost to false positive errors. We experimented with different classification models and cost values using the Weka toolkit [WF05],

Model	NMI	B-Cubed
CLASS-LR	0.9508	0.8155
Merge-ALL	0.8689	0.6765
Merge-FCFS	0.9404	0.7923
Merge-FCFS-θ	**0.9514**	**0.8226**

Table 5.4: Clustering results for the baseline and alternative merging methods.

and found a Multilayer Perceptron to be the best performing model, keeping the number of false positives to just 425 for 397,386 cluster pairs.

Using this classifier, we can predict whether any pair of clusters should be merged. However, we have to address the special case where the classifier's predictions disagree. For example, consider clusters C_1, C_2, and C_3 where the classifier predicts that (C_1, C_2) and (C_1, C_3) should be merged, but (C_2, C_3) should not. We can either merge C_1, C_2, and C_3 into a single cluster (*Merge-ALL*), or merge C_1 and C_2 but not C_3, where C_1 and C_2 appear earlier in the data set (*Merge-FCFS*). For the latter approach, we add a confidence threshold θ, to ensure that only high-probability merge predictions would be used (*Merge-FCFS-θ*). We experimented with different threshold settings on the validation set and used the conservative setting that yielded the best performance (θ=0.995) for experiments over the test set.

To evaluate our approach, we used the clustering quality metrics discussed in Section 5.1, namely, NMI and B-Cubed. Table 5.4 shows the clustering quality over the test set using the original logistic regression similarity model CLASS-LR and our alternative merging strategies. CLASS-LR is a strong baseline, outperforming both merging strategies Merge-ALL and Merge-FCFS. However, Merge-FCFS-θ provides a small improvement over this baseline, according to both NMI and B-Cubed.

To analyze these results, we deconstruct the B-Cubed score into its precision and recall components, which intuitively explain the observed performance of our alternative strategies. Not surprisingly, all merging strategies hurt the B-Cubed precision, which corresponds to the average proportion of items in every document's cluster that belong to the same event. Similarly, all merging strategies improve the B-Cubed recall, which reflects a decrease in

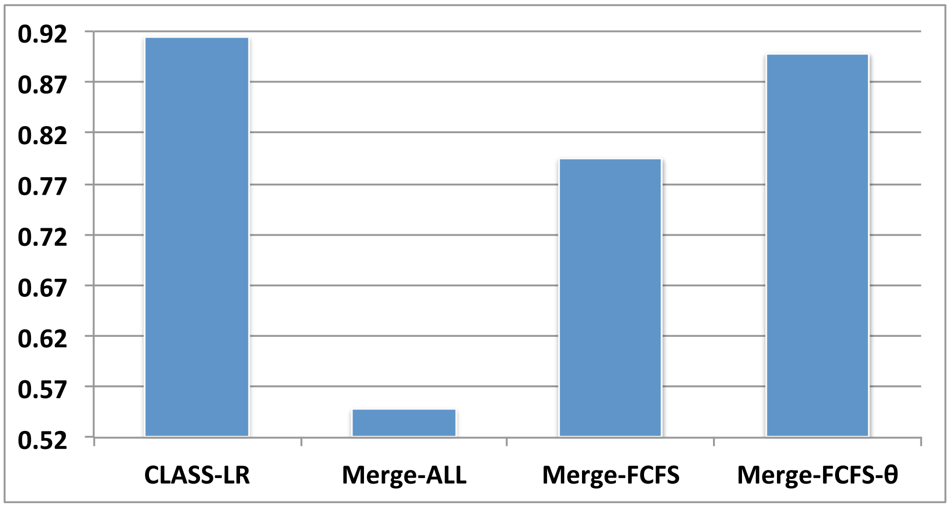


Figure 5.11: B-Cubed Precision scores on the *Upcoming* test dataset.

the number of clusters that each event is spread across. However, the only strategy that creates a better balance between the two, measured by the combined B-Cubed score, is the least aggressive strategy, *Merge-FCFS-θ*. While the overall performance improvement offered by this merging strategy is modest, it serves as an indication that social links can be useful for event identification in social media. Further improvements may be obtained by considering additional types of social links, which we intend to explore in future work.

5.3 Conclusions

In the previous chapter, we introduced a clustering framework for unknown event identification in social media, and showed how we can use it with a post-clustering classification step to distinguish between event and non-event clusters. In this chapter, we focused on learning a similarity metric for the clustering framework, specifically tailored to social media event documents that contain a variety of context features. We discussed and experimented with ensemble-based and classification-based techniques for combining a set of similarity metrics to predict when social media documents correspond to the same event. Our experiments suggest that our similarity metric learning techniques yield better performance than the baselines on which we build. In particular, our classification-based techniques show

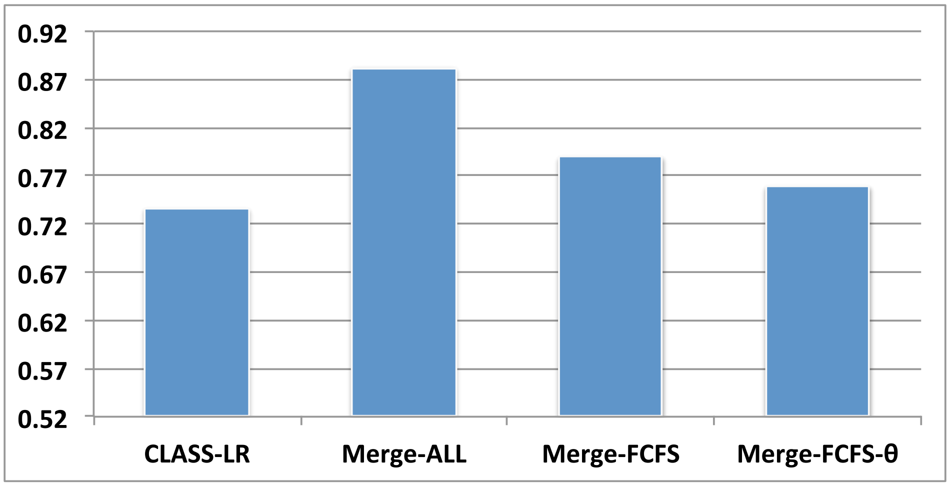


Figure 5.12: B-Cubed Recall scores on the *Upcoming* test dataset.

significant improvement over traditional approaches that use text-based similarity.

In addition, we described an exploratory direction for leveraging information from social links to improve the similarity learning models we introduced in this chapter. Our initial statistics and experiments suggest that these links may be useful similarity cues, especially when context features of social media documents are not sufficient for inferring similarity. In the next chapter, we turn to the known event identification scenario, where we use known features of planned events to identify event documents across different social media sites.

Chapter 6

Identification of Content for Known Events

In the previous chapters, we presented techniques for identifying events and their associated social media documents in the unknown identification scenario, where no information is available on the events that exist in a document stream. However, for planned events (e.g., concerts, parades, conferences), there is revealing and structured information (e.g., title, description, time, location) that is often explicitly available on user-contributed event aggregation platforms (e.g., Last.fm events, EventBrite, Facebook events). For such events, we explore approaches to automatically identify diverse social media content, under the known event identification scenario. Importantly, we address the challenge of automatically identifying user-contributed content for these planned events across different social media sites.

Automatically identifying social media content associated with planned events is a challenging problem due to the heterogenous and noisy nature of the data. These properties of the data present a double challenge in our setting, where both the planned event information and its associated social media content tend to exhibit missing or ambiguous information, and often include short, ungrammatical textual features. In our "Celebrate Brooklyn!" example, event features (e.g., title, description, location) are supplied by a Last.fm user; therefore, these features may consist of generic titles (e.g., "Opening Night

Concert"), missing descriptions, or insufficient venue information (e.g., "Prospect Park," with no exact address). Similarly, social media content associated with this event may be ambiguous (e.g., a YouTube video titled "Bird singing at the opening night gala") or not have a clear connection to the event (e.g., a tweet stating "#CB! starts next week, very excited!").

To identify content for planned events, we leverage explicitly provided event features such as title (e.g., "Celebrate Brooklyn! Opening Gala"), description (e.g., "Singer/songwriter Andrew Bird will open the 2011 Celebrate Brooklyn! season"), time/date (e.g., June 10, 2011), location (e.g., Brooklyn, NY), and venue (e.g., "Prospect Park") to *automatically* formulate queries used to retrieve related social media content from *multiple* social media sites. Importantly, we propose a two-step query generation approach: the first step combines planned event features into several queries aimed at retrieving *high-precision* results; the second step uses these high-precision results along with text processing techniques such as term extraction and frequency analysis to build additional queries, aimed at improving recall. We experiment with formulating queries for each social media site individually, and also explore ways to use retrieved content from one site to improve the identification process on another site. In summary, the contributions of this chapter are as follows:

- We pose the problem of identifying social media content for planned events as a query generation and retrieval task (Section 6.1)

- We develop precision-oriented query generation strategies using planned event features (Section 6.2)

- We develop recall-oriented query generation strategies to improve the often low recall of the precision-oriented strategies (Section 6.3)

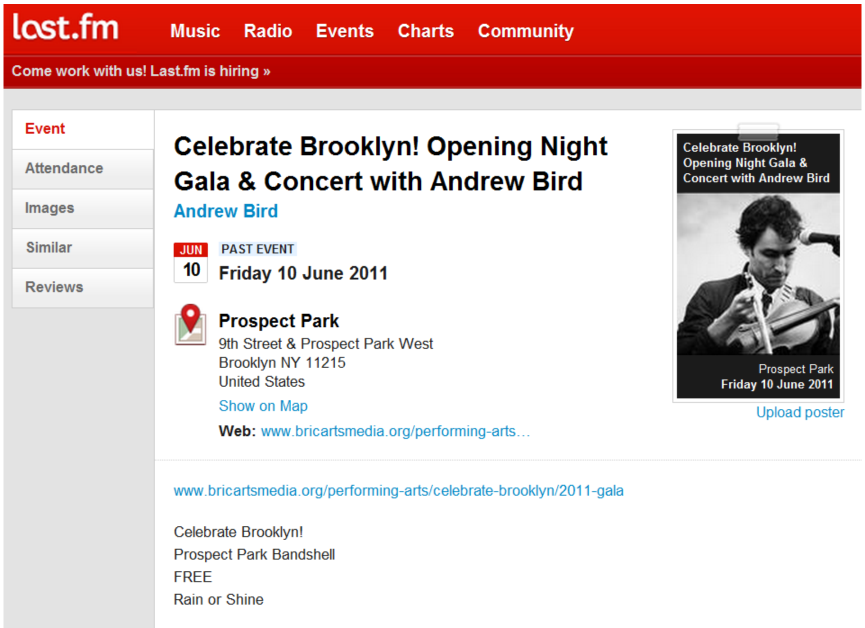
- We demonstrate how query generation strategies developed for one social media site can be used to inform the event content identification process on other social media sites (Section 6.4)

We evaluate our proposed query generation techniques on a set of planned events from several sources and corresponding social media content from Twitter, Flickr, and YouTube

(Section 6.5). We also present an interactive proof-of-concept system that uses our query generation techniques with two alternative user interfaces to retrieve social media documents for planned events (Section 6.6). Finally, we conclude with a discussion of our findings (Section 6.7). The bulk of this paper appeared in [BING11; BCI+11].

6.1 Motivation and Approach

The problem that we address in this chapter is how to identify social media documents across sites for a given planned event with known features (e.g., title, description, time/date, location). Records of planned events—including the event features on which we rely—abound on the Web, on platforms such as Last.fm events, EventBrite, and Facebook events. Figure 6.1 shows a snapshot of such a planned-event record on Last.fm.



Come work with us! Last.fm is hiring »

Celebrate Brooklyn! Opening Night Gala & Concert with Andrew Bird

Andrew Bird

Friday 10 June 2011

Prospect Park
9th Street & Prospect Park West
Brooklyn NY 11215
United States

Web: www.bricartsmedia.org/performing-arts...

www.bricartsmedia.org/performing-arts/celebrate-brooklyn/2011-gala

Celebrate Brooklyn!
Prospect Park Bandshell
FREE
Rain or Shine

Figure 6.1: A Last.fm event record for the "Celebrate Brooklyn!" opening night gala and concert.

As we discussed in Chapter 2, we regard a social media document as relevant to an event if it provides a reflection on the event before, during, or after the event occurs. Consider the "Celebrate Brooklyn!" opening gala concert example (see Figure 6.1). This event's related documents can reflect anticipation of the event (e.g., a tweet stating "I'm so excited for this year's Celebrate Brooklyn! and the FREE opening concert!"), participation in the event (e.g., a video of Andrew Bird singing at the opening gala), and post-event reflections (e.g., a photo of Prospect Park after the concert titled "Andrew Bird really knows how to put on a show"). All of these documents may be relevant to a user seeking information about this event at different times.

For the known event identification scenario that we address in this chapter, the events that we aim to identify are *planned events*, as defined in Chapter 2. Operationally, a planned event is any record posted to one of the public event aggregation platforms available on the Web (e.g., Last.fm events, EventBrite). Unfortunately, not all user-contributed records on these sites are complete and coherent, and while we expect our approaches to handle some missing data, a small subset of these records lack critical features that would make them difficult to interpret by our system and humans alike. Therefore, we do not include in our analysis records that are potentially noisy and incomplete. Specifically, we ignore:

- Records that are missing both start time/date and end time/date
- Records that do not have any location information
- Records with non-English title or description
- Records for endogenous events (Chapter 3) (i.e., events that do not correspond to any real-world occurrence, such as "profile picture change," a Facebook-specific phenomenon with no real-world counterpart)

Regardless of the platform on which they are posted, user-contributed event records generally share a core set of *context features* that describe the event along different dimensions. These features include (see Figure 6.1): title, with the name of the event (e.g., "Celebrate Brooklyn! Opening Night Gala & Concert with Andrew Bird"); description, with a short paragraph outlining specific event details (e.g., "... Celebrate Brooklyn! Prospect Park Bandshell FREE Rain or Shine"); time/date, with the time and date of the event (e.g., Friday 10 June 2011); venue, with the site at which the event is held (e.g., Prospect Park); location, with the address of the event (e.g., Brooklyn, NY). These context features, collectively, will prove helpful for constructing queries that can retrieve different types of social media documents associated with the event.

Problem Definition 3 *Consider any planned-event record posted on an event aggregation platform. Our goal is to retrieve relevant social media documents for this event on multiple social media sites, and identify the top-k such documents from each site, according to given site-specific scoring functions.*

We define the problem of identifying social media documents for planned events as a query generation and retrieval task. Specifically, we design query generation strategies using multi-dimensional features of events on the Web (e.g., textual description, time, location). For each event we generate a variety of queries, which we use *collectively* to retrieve matching social media documents from multiple sites. Since each event could potentially have many associated social media documents, we further narrow down the set of documents we present to a user to the top-k most similar documents, using given site-specific scoring functions (e.g., the multi-feature function in Chapter 5). The similarity metrics that we use, and which are not the focus of this chapter, might differ slightly across social media sites since documents from different sites vary in terms of their associated context features (e.g., documents from Flickr and YouTube have titles and descriptions whereas documents from Twitter do not).

Addressing the above problem presents multiple challenges. Notably, event records are generally informative, but they are also far from perfect. Specifically, sometimes context features are missing altogether, as is occasionally the case with the description feature. Some other times, features have incomplete values (e.g., missing state in the location feature), or values at varying granularities (e.g., "Celebrate Brooklyn! Opening Night Gala & Concert with Andrew Bird" vs. "Opening Night Concert" for the title feature). Still, these event platforms are a rich source of planned event information, so our goal is to develop a variety of query generation strategies, designed to overcome the various challenges that this data presents. Also, social media sites contain highly heterogeneous documents, and the vast majority of these documents are, needless to say, not relevant to an event of interest. Designing queries to target these documents across sites is, therefore, a challenging proposition.

Our approach for associating social media documents with planned events consists of two steps. First, we define precision-oriented queries for an event using its known context features (Section 6.2). These precision-oriented queries aim to collectively retrieve a set of social media documents with high-precision results. Then, to improve the (generally low) recall achieved in the first step, we use term extraction and frequency analysis techniques on the high-precision results to generate recall-oriented queries and retrieve additional documents for the event (Section 6.3). Figure 6.2 presents an overview of our query generation approach.

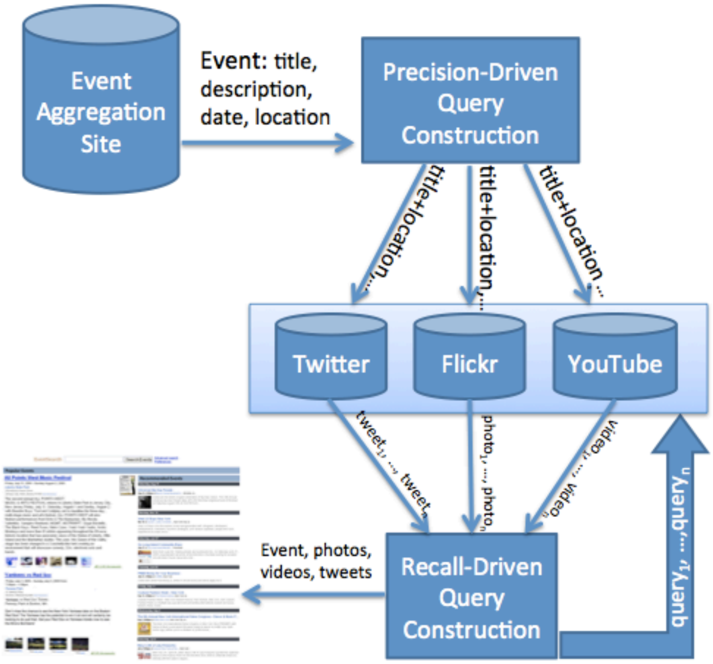

Event Aggregation Site
Event: title, description, date, location
Precision-Driven Query Construction
title+location,...
title+location,...
title+location,...
Twitter
Flickr
YouTube
tweet$_1$, ..., tweet$_n$
photo$_1$, ..., photo$_n$
video$_1$, ..., video$_n$
query$_1$, ..., query$_n$
Event, photos, videos, tweets
Recall-Driven Query Construction

Figure 6.2: Our query-generation approach.

6.2 Precision-Oriented Query Building Strategies

Our first step towards retrieving social media documents for planned events consists of simple query generation strategies that are aimed at achieving high-precision results. These strategies form queries that touch on various aspects of an event (e.g., time/date and venue), following the intuition that these highly restrictive queries should only result in messages that relate to the intended event. We consider a variety of query generation strategies for this step, involving different combinations of the context features, namely, title, time/date, and location, of each event.

The precision-oriented queries for an event consist of combinations of one or more event features. One intuitive feature that we include in all strategies is a restriction on the time at which the retrieved social media documents are posted. In a study of trends on Twitter, Kwak et al. [KLPM10] discovered that most trends last for one week once they become "active" (i.e., once their associated Twitter messages are generated). Since our (planned) events can be anticipated, unlike the trends in [KLPM10], we follow a similar intuition and set the time period T_e that is associated with the event (see Chapter 2) to start a week prior to the event's start time/date and to end a week after the event's end time/date. Note that this time period setting is not the same as the one we used in Chapter 3 (i.e., 72-hours before and after the trend's peak time), because the documents that we collect serve a different purpose from the documents considered in Chapter 3. Specifically, we aim to capture many documents that might be associated with an event, while in Chapter 3 we aim to capture the distribution of documents produced immediately before and after an event. For documents that contain digital media items (e.g., photos, videos), we only consider them if their associated media was created during or after the event's start time. This step, while potentially eliminating a few relevant documents, is aimed at improving precision since we do not expect many digital media items associated with an event to be captured prior to the start of the event. We experimented with more restrictive time windows but observed that relevant documents that contain digital media are generally posted within a week of the event, possibly due to a high barrier to post (e.g., having to upload photos from a camera that does not connect directly to the Internet).

In addition to restricting by time, we always include the title of the event in our precision-oriented strategies, as it often provides a precise notion of the subject of the event. As discussed in Section 6.1, title values exhibit substantial variations in specificity across event records. Some event titles might be too specific (e.g., "Celebrate Brooklyn! Opening Night Gala & Concert with Andrew Bird"); for any such specific title, any social media documents matching it exactly will likely be relevant to the corresponding event. If the titles are too specific, however, no matching documents might be available, which motivates the recall-oriented techniques described in the next section. In contrast, other event titles might be too general (e.g., "Opening Night Concert"). To automatically accommodate these variations in title values, we consider different query generation options for the title feature. Specifically, we generate queries with the original title as a phrase, to capture content for events with detailed titles. We also generate queries with the original title as a phrase augmented with (portions of[1]) the event location, to capture content for events with broad titles, for which the location helps narrow down the matching documents. Finally, we consider alternative query generation techniques that include the title keywords as a list of terms—rather than as a phrase—for flexibility, as well as variations of the non-phrase version that eliminate stop words from the queries.

The intuition for the precision-oriented strategies we define is motivated by the informal results of these strategies over planned events from a pilot system. Our system (Section 6.6) has a customizable interface that allows a user to select among different retrieval strategies. We selected precision-oriented strategies that include three variations of the title (i.e., phrase, list of terms, and list of terms with removed stop words), optionally augmented with either the city or venue portion of the location. We use these precision-oriented strategies to retrieve social media documents for a set of planned events, and verify that they indeed return high-precision results (Section 6.5). The final set of selected precision-oriented strategies is listed in Table 6.1. These strategies, by design, generally offer high precision, though often at the expense of recall.

6.3 Recall-Oriented Query Building Strategies

While the strategies outlined in Section 6.2 often return high-precision social media documents for an event, the number of these high-precision documents is generally low. To improve recall, we develop several strategies for constructing queries using term-frequency

[1] We observed that social media documents usually mention a single, broad aspect of the event's location, such as city or venue, rather than a full address.

Strategy	Example
[“title”+“city”]	[“Celebrate Brooklyn! Opening Night Gala & Concert with Andrew Bird” “Brooklyn”]
[title+“city”]	[Celebrate Brooklyn! Opening Night Gala & Concert with Andrew Bird “Brooklyn”]
[title-stopwords+“city”]	[Celebrate Brooklyn! Opening Night Gala Concert Andrew Bird “Brooklyn”]
[“title”+“venue”]	[“Celebrate Brooklyn! Opening Night Gala & Concert with Andrew Bird” “Prospect Park”]
[title+“venue”]	[Celebrate Brooklyn! Opening Night Gala & Concert with Andrew Bird “Prospect Park”]
[“title”]	[“Celebrate Brooklyn! Opening Night Gala & Concert with Andrew Bird”]
[title]	[Celebrate Brooklyn! Opening Night Gala & Concert with Andrew Bird]
[title-stopwords]	[Celebrate Brooklyn! Opening Night Gala Concert Andrew Bird]

Table 6.1: Our selected precision-oriented strategies.

analysis. Specifically, we treat an event's title, description, and any retrieved results from the precision-oriented techniques as "ground-truth" data for the event. We consider using the precision-oriented results from each social media site individually, and also from all social media sites collectively (Section 6.4).

Using the ground-truth data for each event, we design query formulation techniques to capture terms that uniquely identify each event. These terms should ideally appear in any social media document associated with the event but also be broad enough to match a larger set of documents than possible with the precision-oriented queries. We select these recall-oriented queries in two steps. First, we generate a large set of candidate queries for each event using two different *term analysis and extraction techniques*. Then, to select the most promising queries out of a potentially large set of candidates, we explore a variety of *query ranking strategies* and identify the top queries according to each strategy.

Frequency Analysis: The first query candidate generation technique aims to extract the most frequently used terms, while weighing down terms that are naturally common in the English language. The idea is based on the traditional *tf-idf* approach [MRS08] commonly used in information retrieval. To select these terms, we compute term frequencies over the ground-truth data for word unigrams, bigrams, and trigrams. We then eliminate stop words and remove infrequent n-grams (determined automatically based on the size of the ground-truth corpus). We also eliminate any term that appears in the top 100,000 most frequent words indexed by Microsoft's Bing search engine as of April 2010[2], with the assumption that any of these queries would be too general to describe any event.

To normalize the n-gram term frequency scores, we use a language model built from a large corpus of Web documents (see Section 6.5). With this language model, we compute log probability values for any candidate n-gram term. The probability of a term in the language model provides an indication of its frequency on the Web and can be used to normalize the term's computed frequency. We sort the n-grams extracted for each event according to their normalized term frequency values, and select the top 100 n-grams as candidate queries for the event.

Term Extraction: The second query candidate generation technique aims to identify

[2] http://web-ngram.research.microsoft.com/info/

meaningful event-related concepts in the ground-truth data using an external reference corpus. For this, we use a Web-based term extractor over our available textual event data. This term extractor leverages a large collection of Web documents and query logs to construct an entity dictionary, and uses it along with statistical and linguistic analysis methodologies to find a list of significant terms [KMC05]. The extracted terms for each event serve as additional recall-oriented query candidates, along with the term-frequency query candidates described above.

Each of the techniques we describe could potentially generate a large set of candidate queries. However, many of these queries could be noisy (e.g., [@birdfan], with the name of a user that posts many updates about the event), too general (e.g., [concert tonight]), or describing a specific or non-central aspect of the event (e.g., [Fitz and the Dizzyspells], the name of an Andrew Bird song from the concert). Issuing hundreds of queries for each event is not scalable and could potentially introduce substantial noise, so we need to further reduce the set of queries to the most promising candidates. We explore a variety of strategies for selecting the top candidate queries out of all possible queries that we construct for each event. We consider two important criteria for ordering the event queries: specificity and temporal profile.

Specificity: Specificity assures that we rank long, detailed queries higher than broad, general ones. Since we use conjunctive query semantics, longer queries consisting of multiple terms (e.g. [a,b]), are more restrictive than shorter queries consisting of fewer terms (e.g., [a]). Particularly, since we use term n-gram shingles with n=1, 2, and 3 to construct the recall-oriented queries, our set of candidate queries often includes bigram queries that are subsets of trigram queries (e.g., [bird concert] and [andrew bird concert]). If both such candidates are present in the set, we favor the longer, more detailed version, as we observed that this level of specificity generally helps improve precision and yet is not restrictive enough to hurt recall.

Temporal Profile: The historical temporal profile of a query is another criterion we use to select among the candidate queries for an event. A local spike in document frequency around the time of the event might serve as an indication that the query is associated with the event. As we discussed in Chapters 3 and 4, this type of bursty temporal behavior is a useful, distinguishing characteristic of terms and phrases associated with trending events. Even for planned events that are not trending, the bulk of associated documents is generally posted or captured in close temporal proximity to the time of the event, so queries that are useful for identifying these event documents will also exhibit some increase in document frequency around that time. We keep a record of the number of documents retrieved by each query during the week before and the week after the event, and compare this number to the query's document volume during shorter time periods (one or two days) around the event's time span.

Figure 6.3 shows a document volume histogram over Twitter documents for two recall-oriented queries retrieved around the week of Andrew Bird's concert at "Celebrate Brooklyn!" We can see that the volume of a general query such as [state farm insurance] is consistent over time, whereas the volume of [andrew bird concert], while lower, increases around the time of the event. While this temporal analysis is promising for some social media sites (e.g., Twitter) where the time of the messages generally coincides with the time of the event, it may be problematic for other sites (e.g., YouTube, Flickr) that tend to exhibit a delay between an event's time and upload time of the associated digital media documents for the event, because of the nature of these sites. Therefore, for sites containing digital media, we use the content creation time rather than the upload time, if possible. (This feature is, unfortunately, often noisy or missing, especially for YouTube videos.)

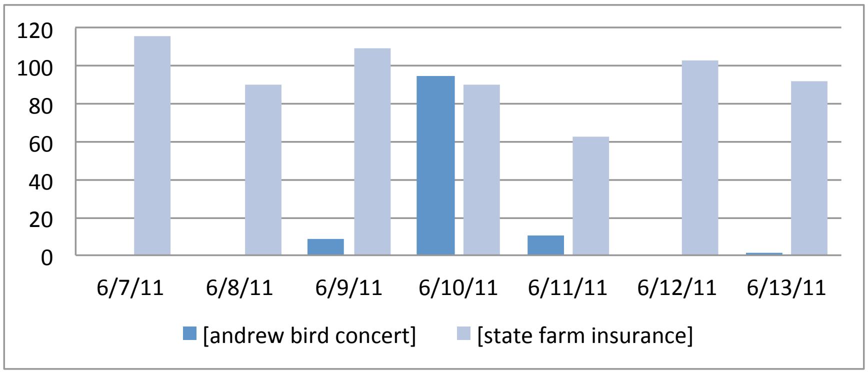


Figure 6.3: Histogram of Twitter document volume over time for two queries around the week of Andrew Bird's Celebrate Brooklyn! concert.

We consider using each of these query selection strategies individually, and also explore ways of combining them, to identify the top candidate queries for any given event. With these queries, we can retrieve associated social media documents from a variety of social media sites. Interestingly, these social media sites can be used as complementary signals for the recall-oriented query generation and retrieval process, as we will see next.

6.4 Leveraging Cross-Site Content

We query for event-related documents on multiple social media sites in order to provide a holistic perspective on the event, complete with digital media and a variety of user perspectives. For the "Celebrate Brooklyn!" opening concert, for instance, we can use these different social media sites to learn about the event (e.g., via a Twitter message "Celebrate Brooklyn kicks off TONIGHT with Andrew Bird concert in Prospect Park!"), watch a video of a song performed at the event (e.g., "Andrew Bird - Effigy (Live) - Prospect Park - Brooklyn, NY" on YouTube), and see up-close photos of Andrew Bird on stage during the event (e.g., "Andrew Bird: Prospect Park Bandshell" photo set on Flickr).

We can leverage event content from one social media site to help retrieve event documents from another social media site in different ways, following the query generation strategies proposed in the previous sections. One simple way, of course, is to generate recall-oriented queries for each site individually and use these queries across sites. Specifically, we can use the high-precision results obtained from an individual site to formulate recall-oriented queries as described in Section 6.3. We can then use these site-specific recall-oriented queries to obtain additional results from other social media sites. This is especially useful when the precision-oriented strategies do not retrieve results from all sites. This is the case for our "Celebrate Brooklyn!" example: since the event title is too specific, the precision-oriented queries fail to retrieve any documents from YouTube, and, therefore, we cannot generate recall-oriented queries for this site. Fortunately, as is often the case, Twitter has a wealth of results for the precision-oriented queries for the event, and the resulting recall-oriented queries (e.g., [andrew bird concert], [brooklyn celebrate]) retrieve relevant videos from YouTube. In short, we manage to extract useful YouTube content through queries derived based on Twitter content.

An alternative way to leverage multi-site social media content is to generate recall-oriented queries using the high-precision results returned from all social media sites collectively. Whenever we obtain precision-oriented results from multiple sites, this approach yields a larger "ground-truth" corpus for the recall-oriented query generation than the ones obtained from each site individually, which may be helpful for identifying salient event terms that appear frequently across sites. At the same time, the results may be dominated by content from one site, possibly obscuring useful content from another site. This approach may also introduce noise or irrelevant content that is often present in some sites and not others (e.g., content-free titles of Flickr photos, Twitter username mentions).

Although content from different social media sites provides promising opportunities, it also presents challenges for our techniques. First, site-specific notations and conventions often introduce noise or inhibit recall. For example, Flickr users often tag photos with their camera settings (e.g., "canoneos5dmarkii"), which may be mistakenly identified as an important event term by the term frequency analysis, especially if the ground-truth corpus for the event is small. In addition, each Flickr tag must consist of a single term, so users often resort to very long multi-word tags (e.g., "greatcanadiancheesefestival"). In contrast, YouTube tags may each consist of several terms, so querying for such long multi-word tags on YouTube rarely yields results. We experimentally evaluate the merits of these alternative multi-site approaches in Section 6.5.

6.5 Experiments

We evaluated our query selection and retrieval techniques using a large dataset of real-world events from several event aggregation sites. For each event, we used our query generation strategies to collect related documents from popular social media sites. We performed three different sets of experiments:

- Comparison of the automatically generated queries against human-produced queries for the events
- Evaluation by human judges of the automatically generated queries

- Evaluation of the quality of the documents retrieved by the automatically generated queries

We report on the dataset and experimental settings, then turn to the results of our experiments.

6.5.1 Experimental Settings

Planned Event Dataset: We assembled a dataset of event records posted between May 13, 2011 and June 11, 2011 on four different event aggregation platforms: Last.fm events, EventBrite, LinkedIn events, and Facebook events. We used the Last.fm API with a location parameter set to "United States"[3] to collect a set of musical performance events. Additionally, we filtered any returned events that did not fall into our specified date range. To collect events from EventBrite, we simply used the EventBrite API with the date parameters set to our specified date range. For LinkedIn events, where an API was not available, we retrieved and parsed event search pages in HTML format, using HTTP GET parameters to specify the date range.

Facebook events deserve special attention due to the difficulty of collecting such data via the site's API. Facebook events can only be retrieved in response to a specific search query or event id. To search for events, we used the most common event terms found in event titles collected by our event tracking system (Section 6.6). This list includes terms that describe specific types of events (e.g., [concert]) and also general terms commonly found in event titles (e.g., [national], [international]). We removed any returned event records that had no location or time information, and events that listed a virtual location (e.g., "everywhere") in their location or venue fields. Unfortunately, after filtering for these required fields we were left with very few events that matched our criteria. Still, we included these events in our experiments as they add diversity to our dataset.

To ensure that we collected events that would potentially have associated social media documents, we filtered out obscure events by requiring a minimum number of event attendees. We tuned this minimum threshold for each site given the observed distribution of

[3] This was the only way to retrieve a set of events from Last.fm without issuing specific queries.

attendees over all collected events. At the end of the process, we collected a total of 393 events, with 90 events from Last.fm, 94 events from EventBrite, 130 events from LinkedIn, and 25 events from Facebook. The above events constitute the test set over which we report our results. For the training and tuning of the strategies, we used a separate set of 329 event records, collected between April 26 and May 11, 2011.

Social Media Documents: We collected social media documents for the events in our dataset from three social media sites: Twitter, YouTube, and Flickr. Specifically, we used each site's respective search API to issue precision-oriented (Section 6.2) and recall-oriented (Section 6.3) queries. From the retrieved results, we eliminated any document that did not exactly match the search query since some site search engines (e.g., for Twitter) search for the query in any content that is linked from the document, and return matching documents as relevant results.

Note that part of our evaluation considers the quality of the top-k documents retrieved by the automatically generated queries (see problem definition in Section 6.1). The ranking of *documents* for an event is not the focus of this chapter. While we do explore techniques for selecting relevant and useful documents for an event (see Chapter 7), the task of learning a ranking function for event documents is reserved for future work, as we discuss in Chapter 9. For our evaluation, we rank the documents retrieved for an event by computing their similarity to the event record using (an adaptation of) the multi-feature similarity function in Chapter 5. As one additional component of the similarity, not present in Chapter 5, we consider the percentage of queries that retrieve a given document when we compute the score for the document and an event. Intuitively, we have observed that documents that are retrieved by several of our queries for an event should be preferred over documents that are retrieved by one such query.

Precision-Oriented Query Generation: For each event, we generate precision-oriented queries as defined in Section 6.2 using the event's context features, namely, title, time/date, city and venue. As an exception, we do not generate queries using the three title-only strategies for Last.fm events since we observed that many of the event titles on Last.fm consist of the name of a performer without any other context. Even though we restrict the social media documents that we retrieve to a specific time period around the event, it is often difficult in the Last.fm case to distinguish between two events held by the same performer in close time proximity. By forcing the location (i.e., city or venue) as part of the query for such events, we ensure that our precision-oriented queries produce results from the intended performance. For event records from the rest of the sites we use all precision-oriented queries from Section 6.2.

Recall-Oriented Query Generation: For each event, we generate recall-oriented queries as described in Section 6.3. To perform the frequency analysis, we index the documents using Lucene[4], with term n-grams, for n=$1, 2$, and 3. To normalize n-gram term frequency scores, we use the Microsoft Web n-gram Service[5], which provides n-gram log probability values. This service returns the joint probability of n-gram terms using a language model created from documents indexed by Microsoft's Bing search engine.

We extract meaningful queries from the high-precision results using the Yahoo! Term Extraction Web Service[6], which returns a list of significant terms or phrases given a segment of text. This term extractor leverages a large collection of documents and query logs to construct an entity dictionary and uses it along with a statistical and linguistic analysis [KMC05] to process the given textual event data. This term extraction service has shown promising results on preliminary experiments with training data, to complement the first term frequency analysis technique above. It has also been successfully used in prior work for similar tasks [DI08; KMC05].

Query Generation and Ranking Techniques: Our experiments consider a subset of the (potentially many) queries generated using the precision- and recall-oriented strategies above. Different techniques will vary on how these subsets are selected. We consider two basic options to rank the queries for selection, namely, using (1) the "specificity" of the queries, as determined by the n-gram score on the Microsoft Web document corpus, or (2) variations of a "temporal" profile of the queries, determined by analyzing the volume of matching documents for the queries over time. Each alternative technique selects the top-10 queries according to the associated ranking criterion, as follows:

[4] http://lucene.apache.org/

[5] http://research.microsoft.com/web-ngram

[6] http://developer.yahoo.com/search/content/V1/termExtraction.html

- MS n-gram Score (MS): n-gram score of the query from the Microsoft Web n-gram Service
- Time Ratio (TR): ratio of the number of documents created in the 48 hours before and after the event to the number of documents created in the week before and after the event
- Restricted Time Ratio (RTR): ratio of the number of documents created in the 24 hours before and after the event to the number of documents created in the week before and after the event
- MS n-gram Score and Time Ratio (MS-TR): MS score multiplied by TR score
- MS n-gram Score and Restricted Time Ratio (MS-RTR): MS score multiplied by RTR score

We apply these techniques to documents from Twitter, YouTube, and Flickr individually and also to documents from all three sites collectively. We use the site's name or "All," along with the strategy name (e.g., Twitter-MS, All-TR) to distinguish among these alternatives. We also compare the above techniques, which include both precision- and recall-oriented queries, against a technique that selects all precision-oriented queries. We refer to this technique as Precision.

Evaluation and Metrics: To evaluate our strategies, we collected annotations for a random sample of 60 events in our dataset. For each event, we used two annotators for three different tasks: comparison against human-produced queries, human evaluation of generated queries, and evaluation of document retrieval results. To compare our automatically generated queries against human-produced queries, we asked each annotator to provide 5 different queries that would be useful for collecting social media documents for each event. We use the Jaccard coefficient to measure the similarity of the set of automatically generated queries G to the set of human-produced queries H for each event. Specifically, for each query $q_g \in G$ and each query $q_h \in H$ we compute $J(q_g, q_h) = (q_g \cap q_h)/(q_g \cup q_h)$, with set operations performed over query terms. The Jaccard value that we report for G is then $\sum_{q_g \in G} max_{q_h \in H}(J(q_g, q_h))/|G|$.

For the human evaluation of the automatically generated queries, we asked two annotators to label 2,037 queries selected by our strategies for each event on a scale of 1-5, based on their relevance to the event. Here, we aim to gauge the potential of each query to retrieve results related to the event. For our "Celebrate Brooklyn!" example, the queries [celebrate], [celebrate brooklyn], and [andrew bird celebrate brooklyn] would receive scores of 1, 3, and 5, respectively. In cases of disagreement we use the average rating. For two events in this set, our annotators were unable to provide queries due to ambiguous content (e.g., "ready film" as the title, without description), and content in a foreign language (e.g., queries in Italian for "FashionCamp," despite setting our API parameters for English-only content). These events were removed from the analysis.

Finally, for the evaluation of the quality of the documents retrieved by the automatically generated queries, we used Amazon's Mechanical Turk[7] to collect relevance judgments for the top-20 documents retrieved from Twitter, YouTube, and Flickr for each of our query selection techniques above. We collected two binary relevance judgments for each document, and an optional third judgment in cases of annotator disagreement. To evaluate the retrieved documents, we use a standard metric, namely, normalized discounted cumulative gain, or *NDCG* [CMS09], which captures the quality of ranked lists with focus on the top results. We use the binary version of NDCG [CMS09], to measure how well our approach ranks the top documents relative to their ideal ranking.

6.5.2 Experimental Results

We begin by comparing the similarity of the automatically generated queries and human-produced queries for our events. Table 6.2 shows the results of our query generation methods using documents from Twitter, Flickr, and YouTube separately, and documents from all sites collectively. Across all strategies, queries generated using Flickr or YouTube documents were less similar to the human-produced queries compared to queries generated using Twitter documents. For Flickr, this result can be explained by the common use of long multi-word tags, which were often selected as the top queries by our strategies (e.g., [20110603musichallofwilliamsburgbrooklynny]). While these queries may not reflect

[7] `https://www.mturk.com`

Strategy	Twitter	Flickr	YouTube	All
MS	0.571	0.216	0.181	0.272
TR	0.524	0.254	0.097	0.277
RTR	0.517	0.253	0.094	0.317
MS-TR	0.531	0.209	0.141	0.244
MS-RTR	0.523	0.209	0.141	0.263

Table 6.2: Jaccard coefficient for automatically generated queries and human-produced queries.

human behavior, they could still be useful for retrieving event content, as we will see. In contrast, Precision had the highest Jaccard value at 0.705, indicating that the human-produced queries were most similar to the precision-oriented queries we defined in Section 6.2. Interestingly, using documents from all sites collectively did not improve the similarity, possibly due to the presence of Flickr tags among the selected queries for this strategy.

For the next step in our analysis, Figure 6.4 shows the average annotator rating for our alternative query generation approaches. Not surprisingly, Precision achieved the best average rating since, by design, it produced very detailed queries that are expected to return relevant results for their associated events. The query generation techniques that used Twitter documents, especially Twitter-MS, were again the most successful set of techniques. Based on our annotation guidelines, the score of Twitter-MS indicates that, on average, queries generated by this strategy are expected to retrieve some results for their associated event. The query generation techniques that used YouTube documents received the lowest scores in this evaluation. One possible explanation is that the query-generation strategies may not be effective when formulated using YouTube data alone, which may be related to the lack of reliable temporal information for YouTube documents, as we discussed in Section 6.3.

For our third set of experiments, we examined the relevance of documents retrieved by our query generation strategies to their associated events. Figure 6.5 shows the NDCG scores for the top $5, 10, 15$, and 20 Twitter documents retrieved by Precision, Twitter-MS, and Twitter-RTR (Twitter-TR, Twitter-MS-TR, and Twitter-MS-RTR produced similar

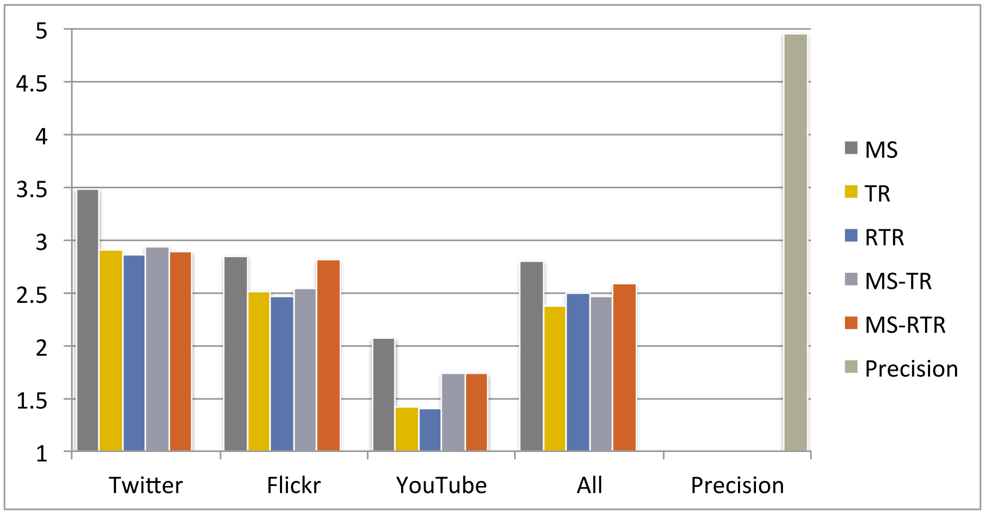


Figure 6.4: Average annotator rating of our automatically generated queries.

Strategy	5 Docs	10 Docs	15 Docs	20 Docs
Twitter-MS	0.759	0.724	0.690	0.690
Twitter-RTR	0.828	0.793	0.759	0.759
Precision	0.414	0.293	0.241	0.224

Table 6.3: Percentage of events with Twitter results at different recall levels for alternative query strategies.

results to Twitter-MS and Twitter-RTR, and were, therefore, omitted). Validating our earlier observation (Section 6.2), Precision retrieved highly relevant results. Both Twitter-MS and Twitter-RTR also produced good results, demonstrating their effectiveness at retrieving Twitter documents for planned events.

It is important to note that the NDCG scores at each level of recall were averaged over the set of events that had some returned results for each strategy. Table 6.3 reports the percentage of events in our dataset for which each strategy returned results at various levels of recall. As expected, Precision returned results for a small fraction of the events. Interestingly, Twitter-RTR returned results for a larger proportion of the events than Twitter-MS. This can be explained by the way these alternative strategies select their top queries. Specifically, all queries selected for Twitter-RTR must have some matching documents, since we

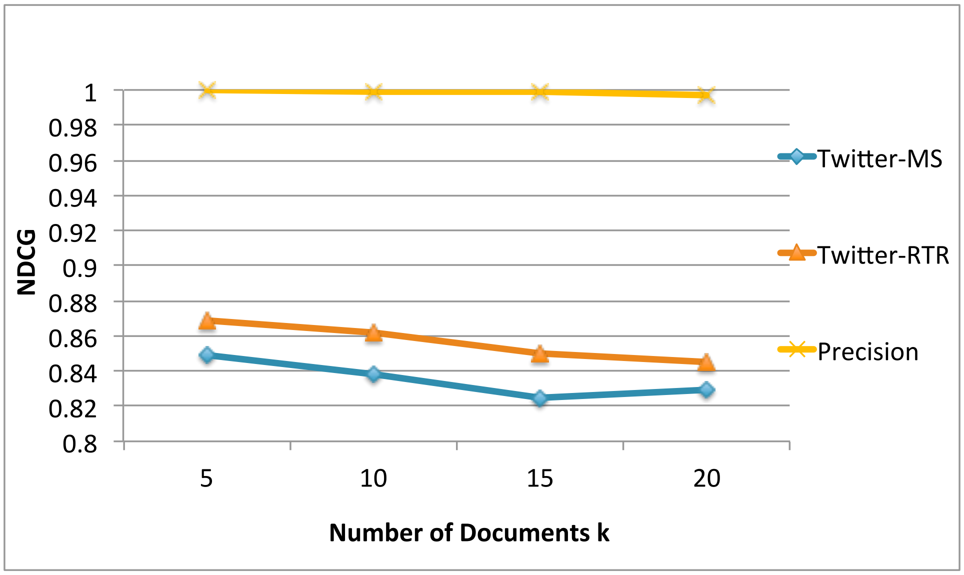


Figure 6.5: NDCG scores for top-k Twitter documents retrieved by our query strategies.

consider each query's document volume over time as the selection criterion. In contrast, Twitter-MS is biased towards rare terms (i.e., terms with lower probability scores), making it the second most precise among the strategies, following Precision.

Our next set of results examines the effectiveness of our approaches for retrieving event documents across social media sites. Given our observations from the query-based evaluations, we evaluated the relevance of documents retrieved by the best performing query generation approach, namely, MS-Twitter, from both YouTube and Flickr. Figure 6.6 shows the NDCG scores of Precision, Twitter-MS, and YouTube-MS for the top-k YouTube documents, averaged over all events. In addition, the size of each point reflects the number of events that had at least k documents retrieved by the strategy. As we can see, Twitter-MS performed better and retrieved results for more events than YouTube-MS, indicating that Twitter documents can be potentially used to improve both precision and recall of YouTube documents for planned events.

We performed a similar evaluation over documents from Flickr, using Precision, Twitter-MS, and Flickr-MS. Precision, expectedly, retrieved relevant results for a small number of events. Interestingly, unlike YouTube-MS, Flickr-MS achieved higher NDCG scores than

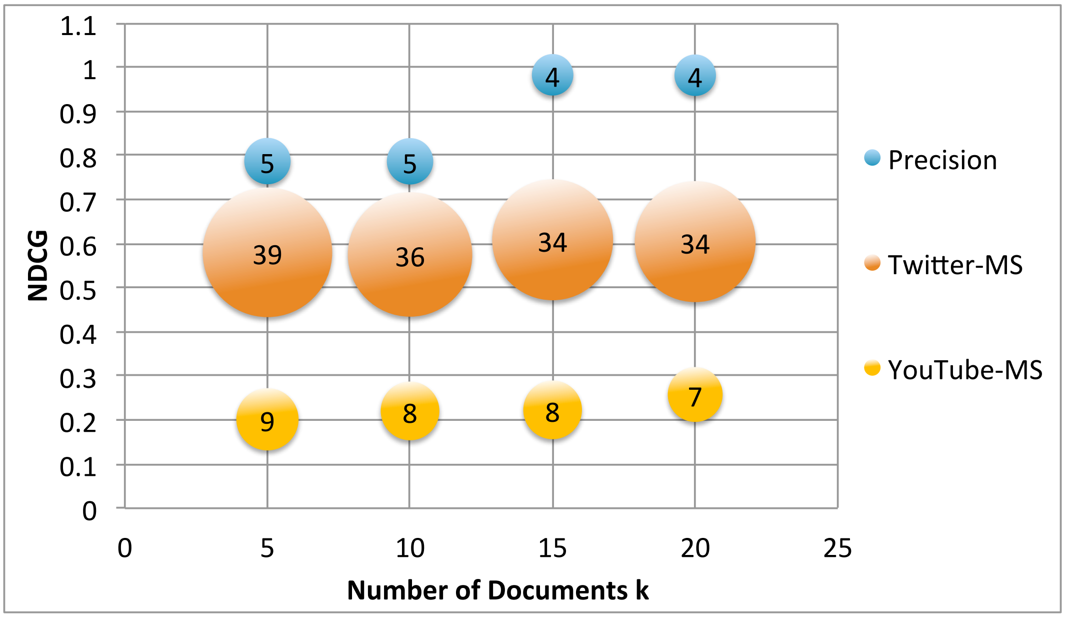


Figure 6.6: NDCG scores for top-k YouTube documents retrieved by our query strategies.

Twitter-MS. However, the number of events covered by Flickr-MS is smaller than the number of events covered by Twitter-MS, showing that Twitter-MS can still retrieve relevant Flickr documents and can be particularly useful in cases where Flickr-MS returns no results.

Overall, our evaluation showed that our query generation approaches can effectively retrieve relevant social media documents for planned events on multiple social media sites. In addition, we demonstrated that we can leverage social media documents on Twitter to generate a query strategy (i.e., Twitter-MS) that can retrieve relevant event documents on YouTube and Flickr.

6.6 Event Tracking System

To enable interaction with our various query formulation strategies described in this chapter, we created a proof-of-concept system that, given an event record and a query formulation strategy, builds the appropriate queries as outlined by the strategy and returns all matching event documents. The current implementation uses records from the Upcoming event database and retrieves event content from Twitter. We describe two applications that build

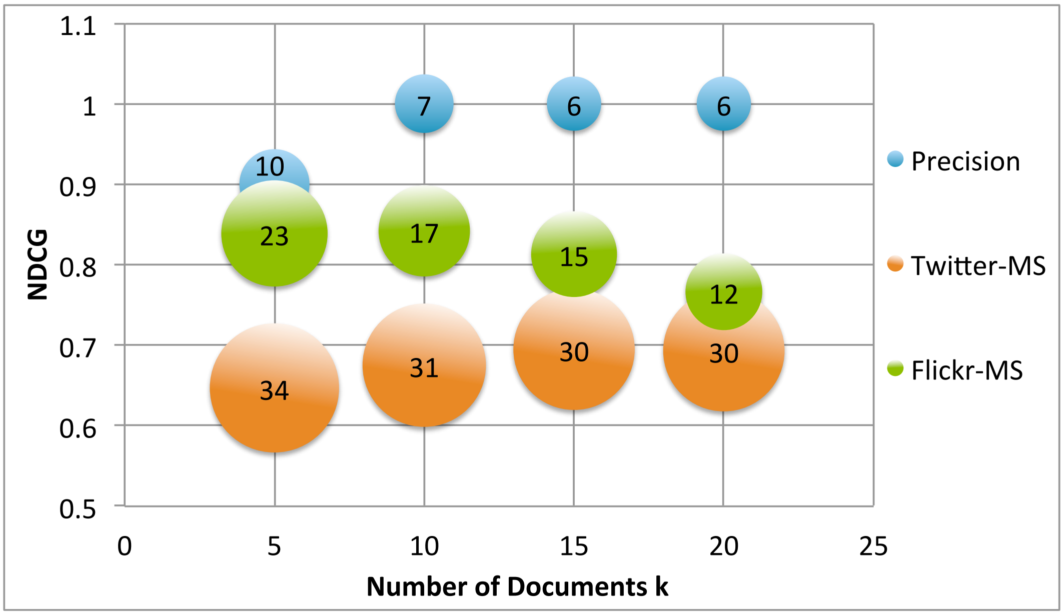


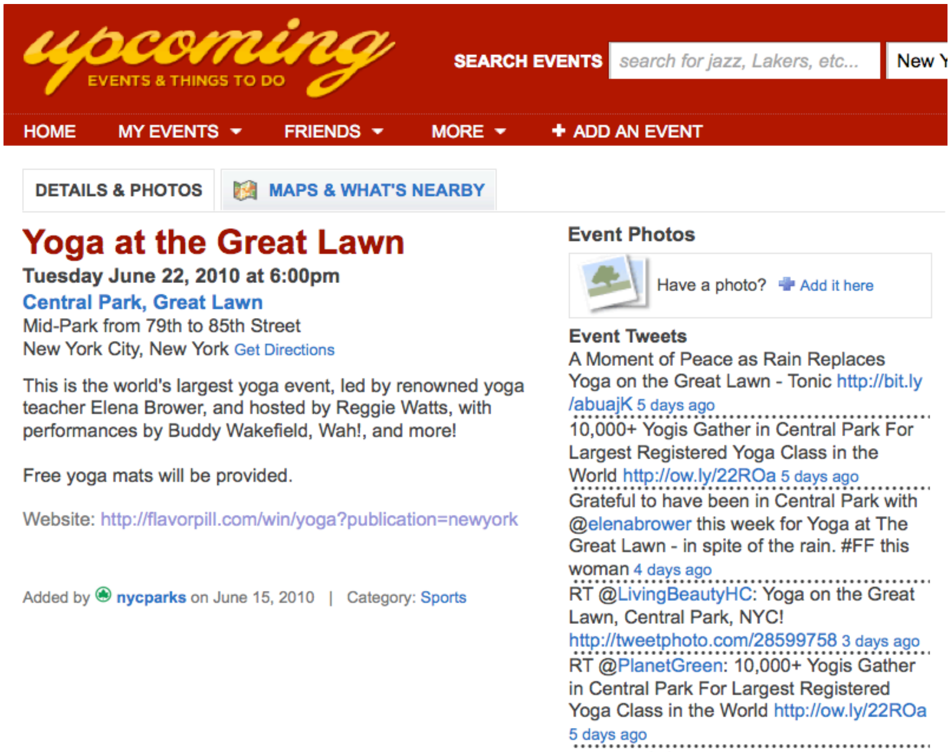
Figure 6.7: NDCG scores for top-k Flickr documents retrieved by our query strategies.

on this query formulation system to create two user experiences for interacting with social media content for planned events.

6.6.1 Browser Plug-In

Our first sample application uses a browser plug-in script that enables seamless embedding of social media documents related to planned events on Upcoming (Figure 6.8). This plug-in script calls a query formulation engine with the set of precision-oriented querying strategies described in Section 6.2. These strategies may be modified via plug-in settings, or dynamically selected based on event type. When a user navigates to an Upcoming event page, the plug-in script collects the event ID and associated event features, and sends them to the query formulation engine, along with the selected query building strategies. The query formulation engine automatically constructs appropriate event-specific queries using the event's features, as required by each query building strategy. The engine then collects all of the matching social media documents, and finally sends the results to the plug-in script.

For efficiency, we issue asynchronous requests to the query formulation engine, where



Yoga at the Great Lawn

Tuesday June 22, 2010 at 6:00pm

Central Park, Great Lawn

Mid-Park from 79th to 85th Street
New York City, New York Get Directions

This is the world's largest yoga event, led by renowned yoga teacher Elena Brower, and hosted by Reggie Watts, with performances by Buddy Wakefield, Wah!, and more!

Free yoga mats will be provided.

Website: http://flavorpill.com/win/yoga?publication=newyork

Added by nycparks on June 15, 2010 | Category: Sports

Event Photos

Have a photo? Add it here

Event Tweets

A Moment of Peace as Rain Replaces Yoga on the Great Lawn - Tonic http://bit.ly/abuajK 5 days ago

10,000+ Yogis Gather in Central Park For Largest Registered Yoga Class in the World http://ow.ly/22ROa 5 days ago

Grateful to have been in Central Park with @elenabrower this week for Yoga at The Great Lawn - in spite of the rain. #FF this woman 4 days ago

RT @LivingBeautyHC: Yoga on the Great Lawn, Central Park, NYC! http://tweetphoto.com/28599758 3 days ago

RT @PlanetGreen: 10,000+ Yogis Gather in Central Park For Largest Registered Yoga Class in the World http://ow.ly/22ROa 5 days ago

Figure 6.8: Browser plug-in.

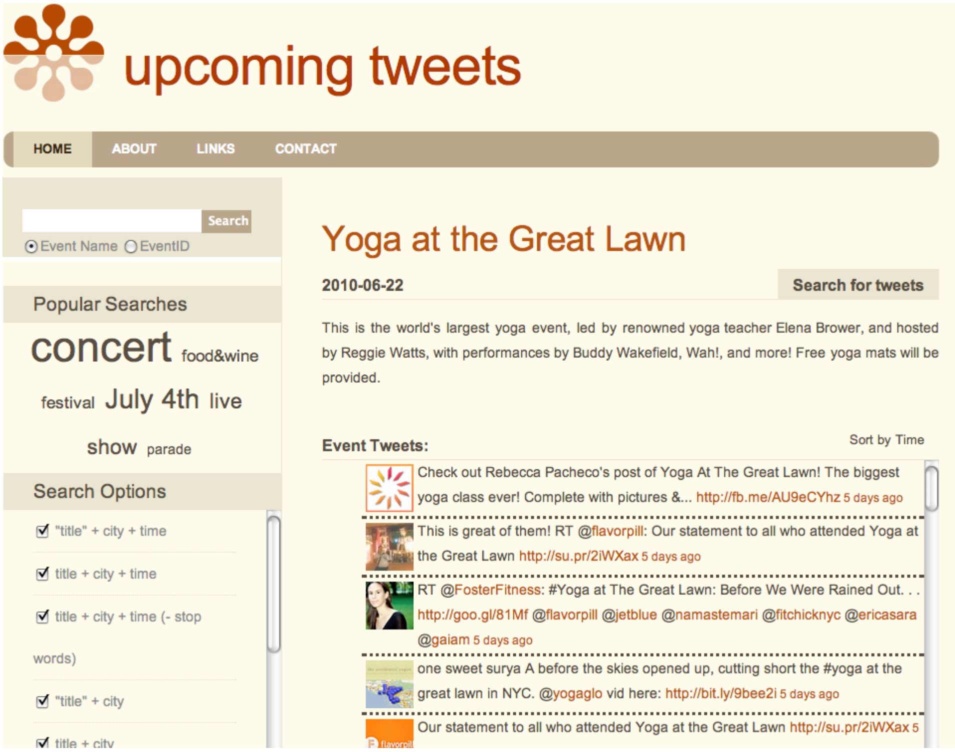
each strategy corresponds to one request, and post the documents in the order in which they are returned. We use a hash-map to keep track of all documents that are already displayed on the page, in order to avoid displaying duplicate documents, which may be returned by the different strategies. Additional performance improvement is gained from issuing a request for any locally cached documents that were previously retrieved for this event by any user of our system. We dynamically append the resulting documents to the Upcoming page, alongside the event description.

6.6.2 Customizable Web-based Interface

Our customizable interface enables users to select specific strategies for automatically retrieving documents for any given event record. Through this interface, users can either search for events or select from a list of recent events (Figure 6.9). On the sidebar, we display the list of query formulation strategies that, if checked, will be used in the retrieval process. When a user selects the "search for tweets" link for an event, our interface issues simultaneous, asynchronous calls to our query formulation engine, to retrieve documents for this event according to each selected strategy.

We display the documents dynamically, as soon as they are retrieved, for efficiency reasons. However, we also include options for ranking the documents according to various criteria. One such ranking criterion is to order the documents according to the time at which they were posted, in case a user is interested in the most up-to-date information about an event. Another ranking option orders documents according to the number of strategies for which they were retrieved. In the future, we plan to experiment with richer ranking functions (Chapter 9).

This interface provides flexibility by allowing users to dynamically modify the set of retrieval strategies, which is particularly useful when the high-precision strategies have insufficient recall. Another feature is a user-driven option to remove generally high-precision strategies when they are expected to introduce noise for a specific event: for example, this scenario might happen with a "title-only" query strategy when an event title is ambiguous (e.g., "4th of July Celebration," referring to an event in Charleston, SC). While the customization feature may be useful in some situations, the automatic query formulation



Search Options

☑ "title" + city + time

☑ title + city + time

☑ title + city + time (- stop words)

☑ "title" + city

☑ title + city

Yoga at the Great Lawn

2010-06-22

This is the world's largest yoga event, led by renowned yoga teacher Elena Brower, and hosted by Reggie Watts, with performances by Buddy Wakefield, Wah!, and more! Free yoga mats will be provided.

Event Tweets:

Check out Rebecca Pacheco's post of Yoga At The Great Lawn! The biggest yoga class ever! Complete with pictures &... http://fb.me/AU9eCYhz 5 days ago

This is great of them! RT @flavorpill: Our statement to all who attended Yoga at the Great Lawn http://su.pr/2iWXax 5 days ago

RT @FosterFitness: #Yoga at The Great Lawn: Before We Were Rained Out. . . http://goo.gl/81Mf @flavorpill @jetblue @namastemari @fitchicknyc @ericasara @gaiam 5 days ago

one sweet surya A before the skies opened up, cutting short the #yoga at the great lawn in NYC. @yogaglo vid here: http://bit.ly/9bee2i 5 days ago

Our statement to all who attended Yoga at the Great Lawn http://su.pr/2iWXax 5

Figure 6.9: Customizable interface.

strategies proposed in this chapter provide a viable alternative for such iterative user supervision, ideally resulting in a satisfactory set of documents without manual intervention.

6.7 Conclusions

While our previous event identification efforts (Chapters 4 and 5) focused on identifying unknown events, in this chapter we explored the known identification scenario, where we have information about the events for which social media content is to be identified. In this setting, we presented a query-oriented solution for retrieving social media documents for planned events across different social media sites. Using a combination of precision-oriented and recall-oriented query generation techniques, we showed how to automatically and effectively associate social media documents with planned events from various sources. Importantly, in accordance with our breadth goal, we demonstrated how social media documents from one social media site can be used to enhance document retrieval on another social media site, thus contributing to the diversity of information that we can collect for planned events. Overall, our techniques help unveil important information related to planned events, presenting diverse, multi-faceted content from the points of view of users who participate in and reflect on these events. As we will discuss, often the number of documents that we identify for an event in social media, both in the known event identification scenario presented in this chapter and in the unknown scenario described in the previous chapters, exceeds the number of documents that could be reasonably consumed by a human looking for information about the event. Therefore, in the next chapter, we consider techniques for selecting among the documents that we identify for each event, so that we can focus on high-quality, relevant, and useful documents for each event.

Chapter 7

Selection of Event Content

Events in social media often have vast amounts of associated content. For example, President Obama's inauguration has over 30,000 associated YouTube videos as of January 2011. A 2010 live broadcast of a U2 concert on YouTube drew over 130,000 posts on Twitter. Even smaller events often feature dozens to hundreds of different documents. In the previous chapters, we described a variety of techniques to identify different types of events and their associated social media documents across several social media sites. Important applications such as event browsing and search could greatly benefit from such identified event content, but they need to prioritize and select from this content to avoid overwhelming their users with too much information. In this chapter, we address this problem of selecting social media content for an event.

Selecting the most salient social media content for an event is a challenging task, due to the heterogeneity and varied quality of the data. For instance, seemingly related social media documents with good textual quality might not be truly relevant to the event (e.g., a Flickr photo titled "Bill cares about his health" for the United States health care reform bill passage). At the same time, relevant, high-quality documents might not be useful (e.g., a Twitter message stating "I can't stop thinking about the health care reform bill passage") as they do not provide much information about the event in question. This chapter examines several approaches for finding high-quality documents that are relevant and contain useful information for each event. For any event, given its associated social media documents, we aim to select documents that best represent the event. We use centrality-based techniques to select documents that have high textual quality, strong relevance to the event, and, importantly, are useful to people looking for information about the event.

In summary, the contributions of this chapter are as follows:

- We suggest and define content selection goals for event content in social media (Section 7.1.1)
- We propose techniques for selecting the top documents for each event according to our defined content selection goals (Section 7.1.2)
- We evaluate our proposed content selection techniques using a large-scale dataset from Twitter (Section 7.2)

Finally, we discuss the implications of our findings and conclude (Section 7.3). The bulk of this chapter appeared in [BNG11b].

7.1 Identifying Event Content

In this chapter, we consider several strategies for selecting social media documents for any event, with focus on textual quality, relevance, and usefulness (Section 7.1.1). Formally, we define the content selection problem that we address in this chapter as follows:

Problem Definition 4 *Given an event e and its associated social media documents D_e, our goal is to select a set of documents from D_e that exhibit high textual quality and strong relevance to the event, and which include highly useful details for people who seek information about the event.*

For this content selection problem, we assume that we are given an event and a corresponding set of social media documents associated with the event. As we discussed in the previous chapters, there are a variety of ways by which we might arrive at this scenario. In this chapter, we make a couple of assumptions about our given events and their associated social media documents. First, we do not know any information about the given event. With this assumption we can handle any type of event, including events discovered under the unknown identification scenario (Chapters 4 and 5) and, of course, any event identified under the known identification scenario (Chapter 6). Second, we assume that the social media documents that we select consist primarily of textual content. With this assumption, our techniques can handle content from Twitter, which generally consists of simple, short textual messages. However, our centrality-based techniques can be easily extended to handle the variety of document representations and corresponding similarity metrics discussed in Chapter 5.

Regardless of the mode of identification or type of event, once we have an identified event and its associated social media documents, we address the problem of selecting a subset of these documents for presentation (Section 7.1.2). We describe our content selection goals and approaches next.

7.1.1 Content Selection Goals

We select documents for each identified event with three desired attributes: *quality*, *relevance*, and *usefulness*. *Quality* refers to the textual quality of the documents, which reflects how well they can be understood by a human. As previously discussed, the quality of documents posted on social media sites varies widely. High-quality documents contain crisp, clear, and effective text that is easy to understand (e.g., a Twitter message stating that "The Superbowl is playing on channel 4 right now"). Low-quality documents, on the other hand, contain incomprehensible text, heavy use of short-hand notation, spelling and grammatical errors, and typos (e.g., a YouTube video titled "obv maj fail lol"). Interestingly, the quality of a document is largely independent of its associated event.

Relevance in our context refers to how well a social media document reflects information related to its associated event. Highly relevant documents clearly refer to or describe their associated event (e.g., a YouTube video, with description: "The steelers' touchdown was amazing - I wish they'd do it again" for the Super Bowl 2010 event). Documents are not relevant to an event if they do not refer to the event in any way (e.g., a Twitter message, stating "good morning, what are people doing today?" for the Super Bowl 2010 event). In between these two extremes are documents that are somewhat relevant to an event, where the event is not the main subject (e.g., A Flickr photo, with description: "I can't believe I'm stuck at work, I'd rather be watching the superbowl" for the Super Bowl 2010 event) or documents that are barely relevant and only obscurely refer to the event (e.g., a Twitter messages, stating that "this game is so boring, but watching the commercials is mildly entertaining" for the Super Bowl 2010 event).

Usefulness refers to the potential value of a social media document for someone who is interested in learning details about an event. Useful documents should provide some insight about the event, beyond simply demonstrating that the event occurred (e.g., via a statement on Twitter, a photo of the event on Flickr). The level of usefulness of social media documents varies. Documents that are clearly useful provide potentially interesting details about the event (e.g., a Twitter message announcing that "The Packers and Steelers are playing in this year's Superbowl" for the Super Bowl 2010 event). Documents that are clearly not useful provide no context or information about the event (e.g., a message by a Twitter user, stating "super bowl!!! that's all folks" for the Super Bowl 2010 event). Other documents may reflect a user's opinion about the event, where somewhat useful event information is directly stated or can be inferred (e.g., a Flickr photo titled "the best superbowl game ever" for the Super Bowl 2010 event).

We use these three attributes, namely, quality, relevance, and usefulness as absolute measures of user satisfaction with the selected event content, and as relative measures of the success of our alternative content selection approaches, which we describe next.

7.1.2 Content Selection Approaches

With our content selection goals in mind, we now propose alternative approaches for selecting a subset of social media documents associated with a given event. These approaches rely on the observation that the most topically central documents in a cluster of event documents are likely to reflect key aspects of the event better than other, less central cluster documents. This notion of centrality in a cluster of social media documents can be defined in a variety of ways:

Centroid: The centroid similarity approach computes the cosine similarity of the *tf-idf* representation (as defined by Kumaran and Allan [KA04]) of each document to its associated event cluster *centroid*, where each cluster term is associated with its average weight across all cluster documents. It then selects the documents with the highest similarity value. Since a cluster's centroid highlights important terms used to describe the event (e.g., for Tiger Woods's famous apology speech, centroid terms with high weight might include "tiger," "woods," "apology," and "elin"), documents with high similarity to these key terms are likely to reflect key aspects of the event, as desired by the relevance and usefulness goals. In addition, since centroid term weights are based on frequency across all documents, they tend to be high for quality terms (e.g., without typos or spelling errors), addressing our quality selection goal.

Degree: An alternative view of centrality involves document similarity across all documents in an event cluster. In this alternative approach, we represent each document in the cluster as a node in a graph, and any pair of nodes whose cosine similarity exceeds a predetermined threshold is connected by an edge. Using this graph formulation, the degree method selects nodes with the highest degree centrality, defined as the degree of each node, weighted by the number of nodes in the graph. Using degree centrality enables us to select documents that contain important terms that may not have been captured by the centroid due to low support in the cluster documents (e.g., a small but highly connected subset of documents might also include the word "mistress" when describing the Tiger Woods apology). In this method, highly connected documents are also likely to include key event terms, a desirable property for content selection.

The degree centrality method treats each edge as an equal vote for its adjacent nodes' centrality. However, it is often beneficial to associate a weight with each edge, based on the similarity value of the nodes it connects. In fact, this idea has been considered for the task of extractive summarization [ER04], a related task where sentences from multiple documents are selected to form a summary. Our third approach, *LexRank*, is based on a state-of-the-art technique by the same name used to select document sentences for summarization [ER04].

LexRank: The LexRank approach [ER04] defines centrality based on the idea that central nodes are connected to other central nodes. In other words, every node has a centrality value, which it distributes to its neighbors. This idea can be represented using the formula $p(m) = \sum_{n \in adj[m]}(p(n)/deg(n))$, where $p(n)$ is the centrality of node n, $adj[m]$ is the set of nodes adjacent to node m, and $deg(n)$ is the degree of node n. The value of $p(m)$ for each cluster document can be computed using the power method [ER04], which estimates the stationary probability distribution resulting from a random walk on the document graph. We select the top documents in the cluster according to their LexRank value.

In addition to these centrality-based approaches, we considered baseline content selection techniques such as selecting the most recent documents added to a cluster or selecting documents from popular users (i.e., users with many followers). Unfortunately, when used in isolation, these techniques suffer from serious drawbacks (e.g., inability to reduce selection of noisy, irrelevant content) so we eliminated them from consideration after running experiments on training data. These potentially useful signals could instead be incorporated with our centrality based approaches in a disciplined way (e.g., using a trained ranking function), a task that we reserve for future work (Chapter 9).

7.2 Experiments

We evaluated our content selection strategies on a large dataset of Twitter messages. We describe this dataset and report the experimental settings (Section 7.2.1), and then turn to the results of our experiments (Section 7.2.2).

7.2.1 Experimental Settings

Data: We used the Twitter API to collect over 2,600,000 Twitter messages, or *tweets*, posted during February 2010 by New York City users (i.e., by Twitter users whose location, as entered by the users, is in the New York City area). This dataset was collected as part of our work on unknown event identification described in Chapter 4, and is location-centric for this reason. However, we believe that this characteristic of the data does not introduce any bias in our evaluation since our techniques currently do not consider the tweets' location in the selection process.

We cluster our entire dataset in an online fashion as described in Section 4.1. We used the data from the first week in February to calibrate the parameters of the clustering algorithm, and then used the second week of February for the development of our centrality-based approaches (and to rule out poorly performing alternatives such as time-based selection). Finally, we report our results on test data selected from the latter half of February (i.e., Weeks 3 and 4).

Annotations: To test the content selection approaches, we selected 50 event clusters, with an average of 412 messages per cluster, from our test set (the presence of event content in the cluster was determined by two annotators, with substantial agreement, with Cohen's kappa coefficient κ=0.79). For each event cluster we selected the top-5 messages according to each content selection approach. We used two annotators to label each message according to our desired attributes: quality, relevance, and usefulness. The annotators labeled each message on a scale of 1 to 4 for each attribute, where a score of 4 signifies high quality, strong relevance, and clear usefulness, and a score of 1 signifies low quality, no relevance, and no usefulness. Agreement between annotators on low (1, 2) and high (3, 4) ratings for each attribute was substantial to high, with kappa coefficient values $\kappa = 0.92, 0.89, 0.61$ for quality, relevance, and usefulness, respectively. In our evaluation, we use the average score for each message to compare the algorithmic results.

Techniques for comparison: We evaluate and compare our three content selection approaches, namely, *Centroid*, *Degree*, and *LexRank*. To compute the degree centrality, we set the similarity threshold for connecting two message nodes to 0.05. For the *LexRank* approach we used the Mead toolkit [ER04] with the LexRank feature option, which produces a ranked list of messages according to their LexRank score.

7.2.2 Experimental Results

We evaluated our three competing approaches according to user-perceived quality, relevance, and usefulness with respect to a specific event. Figure 7.1 summarizes the average performance of these approaches across all 50 test events. All three approaches received high scores for quality (where a score of 4 implies excellent quality). *Degree* and *Centroid*, on average, selected messages that are either somewhat relevant or highly relevant. However, *Centroid* is the only approach that received a high score for usefulness, indicating that, on average, its selected messages were either somewhat or clearly useful with respect to the associated events.

To test for significant differences between the approaches, we also compared them against each other in terms of the number of events that each approach was preferred for. Table

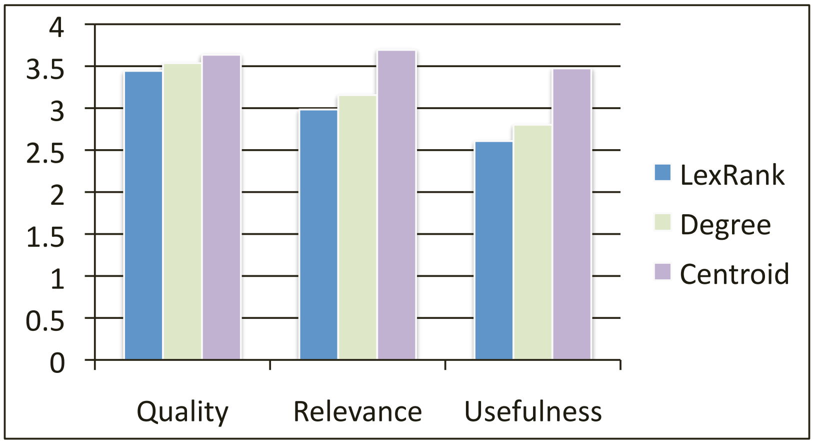


Figure 7.1: Comparison of content selection techniques.

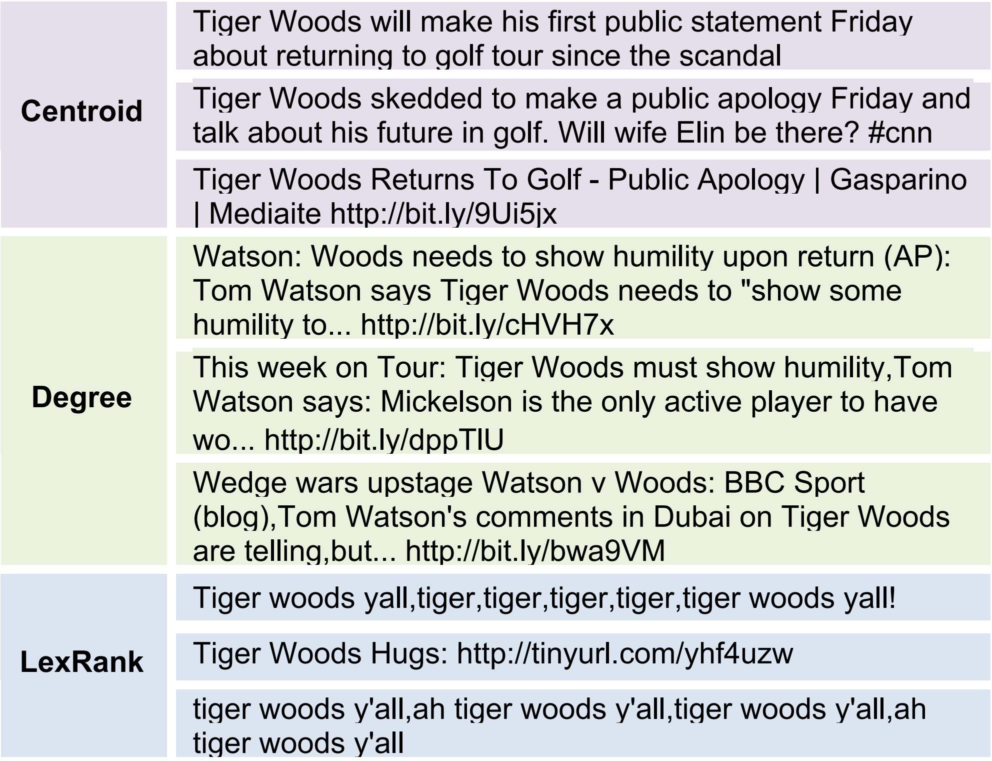
Method	Quality	Relevance	Usefulness
LexRank	2.23	2.4	2.4
Degree	2.02	2.09	2.08
Centroid	**1.75**	**1.51**	**1.52**

Table 7.1: Preference rank of content selection approaches, averaged over 50 test events.

7.1 shows the average rank of each approach according to the three desired attributes. We performed a statistical significance analysis based on these ranked preferences using the Friedman test [Dem06], a non-parametric statistical test for comparing a set of alternative models. According to this test, there are significant differences between the approaches ($p < 0.01$) in terms of relevance and usefulness. Post-hoc analysis of our data using the Nemenyi test [Dem06] determined that *Centroid* is significantly better than the other approaches in terms of both relevance and usefulness. Significant differences between *Degree* and *LexRank* could not be determined. Additionally, we could not reject the null hypothesis of the Friedman test (i.e., that all approaches have similar performance) in terms of quality.

7.3 Conclusions

A single event might sometimes attract hundreds or thousands of social media content items, so being able to rank and filter event content is a requirement for a variety of applications



Centroid	Tiger Woods will make his first public statement Friday about returning to golf tour since the scandal
	Tiger Woods skedded to make a public apology Friday and talk about his future in golf. Will wife Elin be there? #cnn
	Tiger Woods Returns To Golf - Public Apology | Gasparino | Mediaite http://bit.ly/9Ui5jx
Degree	Watson: Woods needs to show humility upon return (AP): Tom Watson says Tiger Woods needs to "show some humility to... http://bit.ly/cHVH7x
	This week on Tour: Tiger Woods must show humility,Tom Watson says: Mickelson is the only active player to have wo... http://bit.ly/dppTIU
	Wedge wars upstage Watson v Woods: BBC Sport (blog),Tom Watson's comments in Dubai on Tiger Woods are telling,but... http://bit.ly/bwa9VM
LexRank	Tiger woods yall,tiger,tiger,tiger,tiger,tiger woods yall!
	Tiger Woods Hugs: http://tinyurl.com/yhf4uzw
	tiger woods y'all,ah tiger woods y'all,tiger woods y'all,ah tiger woods y'all

Figure 7.2: Sample tweets selected by the different approaches for the "Tiger Woods Apology" event.

that aim to communicate that content effectively. In this chapter, we presented content selection approaches that provide a promising step towards this goal.

Among three centrality-based approaches, *Centroid* emerged as the preferred way to select documents given a cluster of documents related to an event. Based on our observation of the data, we believe that the success of this method is related to its inherent assumption that each cluster revolves around one central topic. *LexRank* and *Degree*, on the other hand, tend to select documents that are strongly similar to one another, but may sometimes diverge from the main topic of the cluster (e.g., see Tom Watson's comments on Tiger Woods, selected by *Degree*, in Figure 7.2).

In addition to the centrality-based approaches described in this chapter, we developed a variety of re-ranking techniques that boost the centrality score of documents with potentially useful features (e.g., URLs, tags). Users can manually adjust these techniques based on their preferences. An exploration of these re-ranking techniques using our Twitter dataset revealed a disagreement among users on what aspects of a document (beyond quality, relevance, and usefulness, as defined in our annotation guidelines) are desirable. Some users tend to prefer Twitter content with accompanying URLs, due to the promise of additional, potentially interesting information, while others see more value in verbose messages, with self-contained information related to the event. We plan to explore this further in future work (Chapter 9).

Chapter 8

Related Work

This chapter reviews the literature that is relevant to this dissertation. Section 8.1 outlines prior research on event identification in textual news documents, which consists of many efforts that predated our own, and, in most cases, the rise of social media. Section 8.2 describes work on trend detection and analysis in social media, related to the study of Twitter trends and trending events presented in Chapter 3. Section 8.3 discusses related efforts on event identification in social media, for both the unknown and known identification scenarios that we addressed in Chapters 4, 5, and 6. Section 8.4 provides an overview of large-scale clustering and alternative metric learning techniques that we considered for the clustering framework used in Chapters 4 and 5, and the similarity metric learning problem in Chapter 5. Finally, Section 8.5 discusses related research on organizing and presenting social media content, including event-driven analytics and social media summarization, which is related to our content selection task in Chapter 7.

8.1 Event Identification in Textual News

The topic detection and tracking (TDT) event detection task [All02] inspired many research efforts that focused on discovering and organizing news events [APL98; KA04; YPC98]. While some efforts focused on *online* event detection in continuous text document streams [APL98; KA04], others explored *retrospective* detection of events and their associated text documents [YPC98]. With an abundance of well-formed text, many of the proposed approaches (e.g., [HGM00; MAMS04; ZZW07]) rely on natural language processing techniques to extract linguistically motivated features. Zhang et al. [ZZW07] extracted named entities and part-of-speech tags from textual news documents, and used them to reweigh *tf-idf* representations of these documents for the new event detection task. Filatova and Hatzivassiloglou [FH03] identified named entities corresponding to participants, locations, and times in text documents, and then used the relationships between certain types of entity pairs to detect event content. Hatzivassiloglou et al. [HGM00] used linguistic features (e.g., noun phrase heads, proper names) and learned a logistic regression model for combining these features into a single similarity value. Makkonen et al. [MAMS04] extracted meaningful semantic features such as names, time references, and locations, and learned a similarity function that combines these metrics into a single clustering solution. They concluded that augmenting documents with semantic terms did not improve performance, and reasoned that inadequate similarity functions were partially to blame. As we previously discussed, social media documents have little textual content, and this content is often noisy, and generally lacks well-established structure and semantics. Therefore, techniques that rely on these properties of text are often not suitable for the social media domain. At the same time, the idea of combining a variety of similarity metrics for event detection [HGM00; MAMS04] was extended by our work and tailored to social media documents (Chapter 5).

Extracting events from text has been the focus of numerous studies as part of the NIST initiative for Automatic Content Extraction (ACE) [Ahn06; JG08]. The ACE program defines event extraction as a supervised task, given a small set of predefined event categories and entities, with the goal of extracting a unified representation of the event from text via attributes (e.g., type, subtype, modality, polarity) and event roles (e.g., person, place, buyer, seller). Ahn [Ahn06] divided the event extraction task into different subtasks, including identification of event keyword triggers (see Chapter 2), and determination of event coreference, and then used machine learning methods to optimize and evaluate the results of each subtask. Ji and Grishman [JG08] proposed techniques for extracting event content from multiple topically similar documents, instead of the traditional approach of extracting events from individual documents in isolation.

In contrast with the predefined templates outlined by ACE, Filatova et al. [FHM06] presented techniques to automatically create templates for event types, referred to as "domains," given a set of domain instances (i.e., documents containing information related to events that belong to the domain). Our goal is to identify events of different types over social media documents. For this, we either operate in an unsupervised manner, or in a supervised manner that does not impose restrictions on the event's domain. For the supervised, known event identification task, we do not use explicit templates but rather guide our identification process using event-specific (as opposed to type- or domain-specific) attributes.

8.2 Trend Analysis in Social Media

The general topic of studying trends in social media has recently received considerable research interest. Research efforts often examined a small number of such trends to produce some descriptive and comparative characteristics of social media trends or popular terms. Cheong and Lee [CL09] looked at four trending topics and two control terms, and a subset of the messages associated with each, commenting on features such as the time-based frequency (i.e., volume of messages) for each term, and the category of users and type of devices used to post the associated messages. Yardi and boyd [Yb10] examined the characteristics of content related to three topics on Twitter, two topics representing geographically local news events, and one control topic. The authors studied the messages posted for each topic (i.e., messages containing terms manually selected by the authors to capture related content), and the users who posted them. Among other findings, the authors suggest that local topics feature denser social connectivity between the posting users. Similarly, Sakaki et al. [SOM10] suggest that the social connectivity for breaking events is lower, but have only examined content related to two manually chosen events. In our study of Twitter trends in Chapter 3, we were not able to verify this hypothesis.

To detect trends in Blogs, Glance et al. [GHT04] used a measure of "burstiness," among other techniques, identifying trending terms and phrases. This measure of "burstiness" is similar to the one we used in our study of trends and trending events on Twitter (Chapter 3). As an additional step, Glance et al. clustered the trending terms and phrases to create trending topics, each consisting of a set of related terms. While we do not cluster terms for our study in Chapter 3, we do use clustering to identify events in our unknown identification scenario (Chapters 4 and 5). However, we cluster documents rather than terms, to identify the events and their associated social media documents simultaneously.

Singh and Jain [SJ10] examined Twitter messages with select hashtags and showed that the content for each such set follows a power-law distribution in terms of popularity, time, and geo-location. Kwak et al. [KLPM10] showed that different trending terms on Twitter have different characteristics in terms of the number of replies, mentions, retweets, and "regular" tweets that appear in the set of tweets for each term, but do not reason about why and how exactly these trends are different. Some of the metrics we used in Chapter 3 for characterizing trends are similar to those used in these studies, but we go further and perform a large-scale analysis of trends according to manual assignments of these trends to distinct categories.

On a slightly larger scale, Kwak et al. [KLPM10] also examined the time series volume data of tweets for each trending term in their dataset, namely, a sample of 4,000 of the trending terms computed and published by Twitter. The authors based their analysis on the findings of Crane and Sornette [CS08], which analyzed time series viewing data for individual YouTube videos. Crane and Sornette observed that YouTube videos fall into two categories, based on their view patterns. When a time series shows an immediate and fast rise in a video's views, Crane and Sornette assert that the rise is likely caused by external factors (i.e., attention was drawn to the video from outside the YouTube community) and, therefore, dub this category of videos "exogenous." When there is no such rise, the authors suggest that a video's popularity is due to "endogenous" factors. Videos are also classified as "critical" or "sub critical," again according to the time series data. Kwak et al. [KLPM10] use these guidelines to classify the Twitter trends in each of these two categories, showing how many trends fit each type of time-series signature. However, the two groups of authors never verified that the trends or videos labeled as exogenous or endogenous indeed matched their labels. In Chapter 3, we used the time series data (amongst other characteristics) while manually coding identified trends as exogenous or endogenous in order to observe whether these categories show different time series effects.

The related problem of information dissemination has also attracted substantial attention. As a notable example, recent work studies the diffusion of information in news and blogs [GGLNT04; LBK09]. Leskovec et al. [LBK09] studied how memes propagate and diffuse across the Web, from mainstream media to blogs and vice versa. Specifically, they observed that the peak of attention for a meme in blogs generally lags 2.5 hours behind the peak of attention for the same meme in mainstream media. As another example, Jansen et al. [JZSC09] study word-of-mouth activity around brands on Twitter. The trends we identified on Twitter (Chapter 3) are, of course, both a product and a generator of information dissemination processes.

8.3 Event Identification in Social Media

While event detection in textual news documents has been studied in depth, the identification of events in social media sites is still in its infancy.

Several related papers explored the unknown event identification scenario in social media. Weng and Lee [WL11] proposed wavelet-based signal detection techniques for identifying "real-life" events on Twitter. These techniques can detect significant bursts or trends in a Twitter data stream but, unlike our work in Chapter 4, they do not filter the vast amount of non-event content that exists on Twitter. This, unfortunately, results in poor performance, with very low precision scores compared with the precision achieved by our methods. Related to our work in Chapters 4 and 5, Sankaranarayanan et al. [SST$^+$09] identified late breaking news events on Twitter using clustering, along with a text-based classifier and a set of news "seeders," which are handpicked users known for publishing news (e.g., news agency feeds). As we discussed, such text-based and seeder-driven filtering of non-event data can be used to generate the event document stream we use in Chapter 5. Petrović et al. [POL10] used locality-sensitive hashing to detect the first tweet associated with an event in a stream of Twitter messages. We use the general text-based classifier suggested in [SST$^+$09] and a method for identifying top events suggested by Petrović et al. [POL10] as baseline approaches in our evaluation of the unknown identification methods of Chapter 4.

While our work in the unknown event identification scenario focuses on timely, online, analysis, several efforts tried to address this task using retrospective analysis. Rattenbury et al. [RGN07] analyzed the temporal usage distribution of tags to identify tags that correspond to events. Chen and Roy [CR09] used the time and location associated with Flickr image tags to discover event-related tags with significant distribution patterns (e.g."bursts") in both of these dimensions.

Recent efforts proposed techniques for known identification of events in social media. Many of these techniques rely on a set of manually selected terms to retrieve event-related documents from a single social media site [SOM10; Yb10]. Sakaki et al. [SOM10] developed techniques for identifying earthquake events on Twitter by monitoring keyword triggers (e.g., "earthquake" or "shaking"). In their setting, the type of event must be known a priori, and should be easily represented using simple keyword queries. Most related to our work in Chapter 6, Benson et al. [BHB11] identified Twitter messages for concert events using statistical models to automatically tag artist and venue terms in Twitter messages. Their approach is novel and fully automatic, but it limits the set of identified messages for concert events to those with explicit artist and venue mentions. Importantly, both of these approaches are tailored to one specific social media site. In contrast, we propose methods for identifying social media documents across *multiple* sites with varying types of documents (e.g., photos, videos, textual messages). Our goal is to automatically retrieve social media documents for any planned event, without any assumption about the textual content of the event or its associated documents. While not exclusively in the social media domain, Tsagkias et al. [TdRW11] extracted named entities and quotations from news articles, as well as explicit links between news and social media documents, to identify social media utterances related to individual news stories. In contrast with their well formed, lengthy textual documents and explicitly linked content, content in our known event identification setting (Chapter 6) is brief and often noisy, and generally does not contain explicit links to social media documents.

8.4 Large-Scale Data Clustering

There are many approaches for clustering large-scale data [Ber02], trading off runtime performance and clustering accuracy. One of the important issues to address when clustering large-scale data is how to compare the data elements against each other, which is hard to perform in a scalable manner as the size of the data grows.

Several solutions were proposed to alleviate this problem. One set of solutions [TBW95; ZRL96] uses statistical properties to represent subsets of the data, thus reducing the total number of comparisons to be made. Hatzivassiloglou et al. [HGM00] and Allan et al. [APL98] showed that an incremental single-pass clustering algorithm can be effectively used for event detection in textual news. Hatzivassiloglou et al. discovered that a single-pass approach can offer good, inexpensive performance for text document clustering, given a carefully selected clustering threshold. For the unknown event identification scenario, we adopt this type of single-pass incremental solution, and represent clusters according to the average value of their elements (i.e., centroid).

Other large-scale clustering solutions involve "blocking" methods [BKM06; HS95; MNU00], which partition elements into several subsets based on a rough measure of similarity. These subsets are then clustered in parallel using traditional clustering algorithms (e.g., K-means, EM [Ber02]) with exact similarities. For our clustering framework (Section 4.1), recent work by Reuter et al. [RCD+11] showed that a time-based blocking function can offer efficiency improvements over the complete-link similarity setting, which compares each incoming document to all previously seen documents at any point in the stream. Alternatively, several approaches [DDGR07; MBN07] use locality-sensitive hashing, a method for finding approximate nearest neighbors, to determine a small set of candidate clusters for each element. This method reduces the number of comparisons that the clustering algorithm must make for each element, thus increasing the algorithm's efficiency.

The choice of clustering similarity metric is critical for obtaining high-quality clustering solutions. In domains where more than one similarity metric is appropriate, several approaches have been proposed for combining multiple similarities using machine learning techniques [BBS05; BM03; CKM09; CR02]. Bilenko et al. [BBS05] used an online perceptron to combine several basis similarity functions, including cosine similarity, string edit distance, and relative difference (for numeric values), in the context of a system for linking related database records. Similar to our work in Chapter 5, Chen et al. [CKM09] used weighted ensemble voting and, in addition, context features of the clusterers in the ensemble, to enable entity resolution in different domains (e.g., personal Websites and publications).

Other metric learning approaches use optimization techniques to learn a similarity metric from labeled examples directly [DKJ$^+$07; XNJR02]. Xing et al. [XNJR02] posed the metric learning problem as a convex optimization problem, and showed how such metric can be learned from pairs of similar items using semidefinite programming. While they argue the efficiency of their metric, they only performed experiments over small-scale datasets. Importantly, the optimized metric is part of a family of distance functions known as Mahalanobis distances, which are a generalized form of the standard Euclidean distance. For social media documents, we are interested in a variety of similarity (or, conversely, distance) metrics that reflect the natural, intuitive similarity for a given document representation (e.g., cosine similarity for textual features, geo-coordinate similarity for location-based features). In Chapter 5, we defined document representations and corresponding similarity metrics, and used them as basis functions for a trained similarity metric with classification-based and ensemble-based techniques.

8.5 Social Media Content Summarization, Topic Discovery, and Analytics

Research on summarizing, discovering, or otherwise presenting social media content has gathered recent attention. The task of social media summarization is related to our content selection task (Chapter 7), but instead of selecting a set of potentially disconnected messages, it aims to construct a coherent summary representation. Several efforts [CP11; SHK10] considered ways to summarize a set of social media documents related to a specific topic. Sharifi et al. [SHK10] proposed approaches for summarizing a set of Twitter messages that were retrieved in response to a keyword query. They used graph-based phrase reinforcement and *tf-idf* techniques to produce very short summaries, which often consist of fewer than 10 words. Chakrabarti and Punera [CP11] proposed techniques for summarizing Twitter messages for *events*. They used Hidden Markov Models to segment the set of messages into sub-events, and then selected key messages from each "interesting" sub-event, to include in the overall summary. This approach, as the authors note, is geared towards structured, long-running events and its effectiveness has not been determined for short events such as concerts or festivals.

Identifying latent, though not necessarily trending, topics on Twitter is the subject of many recent efforts [HD10; OKA10; RDL10; ZJH+11]. Hong and Davidson [HD10] proposed several schemes to train standard LDA, and the Author-Topic LDA models for topic discovery over Twitter data. Ramage et al. [RDL10] used Labeled LDA to map Twitter messages into learned latent and labeled dimensions (e.g., using hashtags, emoticons). Interestingly, they showed how topic models can be used to characterize users by the topics they most commonly use. O'Connor et al. [OKA10] described a system for presentation of Twitter search results, which uses a language modeling approach to identify topic phrases that are most distinctive for a set of retrieved Twitter messages. Search results are then grouped by topic and theme, and ranked based on topical diversity, size, and uniqueness of information. Finally, Zhao et al. [ZJH+11] extracted and ranked topical key phrases on Twitter, using topic models and topic-biased PageRank. They defined relevance and interestingness with respect to a topic as the properties of good key phrases they aim to extract. We considered topic modeling as a potential approach for event identification, despite several drawbacks of this type of solution for our task. We discuss this issue in detail in the following chapter.

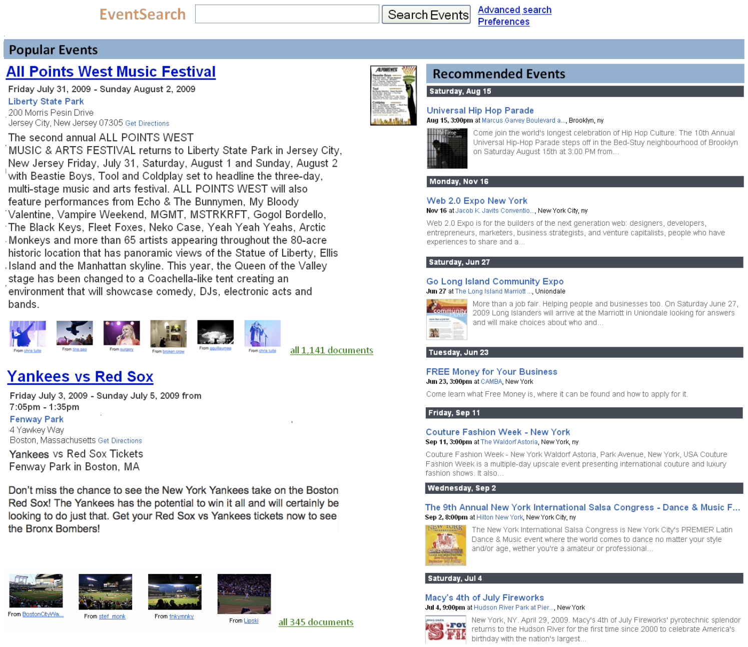
Several recent efforts attempted to provide analytics for events detected or tracked on Twitter. De Longueville et al. [DLSL09] described a method for using Twitter to track forest fires and the response to the fires by Twitter users. Starbird et al. [SPHV10] described the temporal distribution, sources of information, and locations in tweets from the Red River Valley floods of April 2009. Nagarajan et al. [NGS+09] downloaded Twitter data for three events over time, and analyzed the topical, geographic, and temporal importance of descriptors (e.g., different keywords) that can help visualize the event data. O'Connor et al. [OBRS10] performed sentiment analysis on Twitter, and showed a correlation between sentiment measured on Twitter and public opinions derived from polling data for the United States presidential elections in 2008, and presidential job approval in 2009. Finally, Shamma et al. [SKC10], Diakopolous and Shamma [DS10], and Diakopoulos et al. [DNKS10] analyze the tweets corresponding to large-scale media events (e.g., the United States President's annual State of the Union speech) to improve event reasoning, visualization, and analytics. These tasks may all be improved or better automated with the event identification techniques presented in this dissertation.

Chapter 9

Conclusions and Future Work

As users continue to share event-related content through social media channels, identifying and characterizing this content remains critical to enable a variety of applications that build on this useful source of information. Users should be able to interact with event content in a timely manner, and also be exposed to a variety of content from different social media sites, offering a multifaceted view of events, complete with textual content and multimedia. Figure 9.1 illustrates an event browsing and search system, which is one potential application that could benefit from temporally relevant and multifaceted social media event data.

In this dissertation, we presented event identification, characterization, and content selection techniques, each of which serves as an integral part of such applications that interact with events, and their associated documents, in social media. Specifically, we outlined alternative social media identification scenarios, and the types of events that we can identify under each scenario (Chapter 2). For the unknown identification scenario, where events are identified in an unsupervised manner, we studied the different types of trending events and non-event trends that exist in social media (Chapter 3). We then leveraged the conclusions of our study to inform our event classifiers, which we used in conjunction with a proposed clustering framework for unknown identification of events (Chapter 4). To improve our clustering framework, we developed alternative approaches for learning multi-feature similarity metrics, suitable for the social media domain, using the rich family of context features associated with social media documents (Chapter 5). Then, in the known identification scenario, we developed query formulation strategies to identify social media documents for planned events. Importantly, we showed how event content identified on one social media site can be used to identify additional event content on other social media sites (Chapter 6). Finally, we explored approaches for selecting a subset of the documents associated with any event in social media, with focus on content quality, relevance, and usefulness (Chapter 7).



Popular Events

All Points West Music Festival

Friday July 31, 2009 - Sunday August 2, 2009
Liberty State Park
200 Morris Pesin Drive
Jersey City, New Jersey 07305 Get Directions

The second annual ALL POINTS WEST MUSIC & ARTS FESTIVAL returns to Liberty State Park in Jersey City, New Jersey Friday, July 31, Saturday, August 1 and Sunday, August 2 with Beastie Boys, Tool and Coldplay set to headline the three-day, multi-stage music and arts festival. ALL POINTS WEST will also feature performances from Echo & The Bunnymen, My Bloody Valentine, Vampire Weekend, MGMT, MSTRKRFT, Gogol Bordello, The Black Keys, Fleet Foxes, Neko Case, Yeah Yeah Yeahs, Arctic Monkeys and more than 65 artists appearing throughout the 80-acre historic location that has panoramic views of the Statue of Liberty, Ellis Island and the Manhattan skyline. This year, the Queen of the Valley stage has been changed to a Coachella-like tent creating an environment that will showcase comedy, DJs, electronic acts and bands.

From chris tuite From tina.gao From surgery From broken crow From gguillaumee From chris tuite all 1,141 documents

Yankees vs Red Sox

Friday July 3, 2009 - Sunday July 5, 2009 from 7:05pm - 1:35pm
Fenway Park
4 Yawkey Way
Boston, Massachusetts Get Directions

Yankees vs Red Sox Tickets
Fenway Park in Boston, MA

Don't miss the chance to see the New York Yankees take on the Boston Red Sox! The Yankees has the potential to win it all and will certainly be looking to do just that. Get your Red Sox vs Yankees tickets now to see the Bronx Bombers!

From BostonCityWa... From stef_monk From fnkymnky From Lipski all 345 documents

Recommended Events

Saturday, Aug 15

Universal Hip Hop Parade
Aug 15, 3:00pm at Marcus Garvey Boulevard a..., Brooklyn, ny
Come join the world's longest celebration of Hip Hop Culture. The 10th Annual Universal Hip-Hop Parade steps off in the Bed-Stuy neighbourhood of Brooklyn on Saturday August 15th at 3:00 PM from...

Monday, Nov 16

Web 2.0 Expo New York
Nov 16 at Jacob K. Javits Conventio..., New York City, ny
Web 2.0 Expo is for the builders of the next generation web: designers, developers, entrepreneurs, marketers, business strategists, and venture capitalists, people who have experiences to share and a...

Saturday, Jun 27

Go Long Island Community Expo
Jun 27 at The Long Island Marriott ..., Uniondale
More than a job fair. Helping people and businesses too. On Saturday June 27, 2009 Long Islanders will arrive at the Marriott in Uniondale looking for answers and will make choices about who and...

Tuesday, Jun 23

FREE Money for Your Business
Jun 23, 3:00pm at CAMBA, New York
Come learn what Free Money is, where it can be found and how to apply for it.

Friday, Sep 11

Couture Fashion Week - New York
Sep 11, 3:00pm at The Waldorf Astoria, New York, ny
Couture Fashion Week - New York Waldorf Astoria, Park Avenue, New York, USA Couture Fashion Week is a multiple-day upscale event presenting international couture and luxury fashion shows. It also...

Wednesday, Sep 2

The 9th Annual New York International Salsa Congress - Dance & Music F...
Sep 2, 8:00pm at Hilton New York, New York City, ny
The New York International Salsa Congress is New York City's PREMIER Latin Dance & Music event where the world comes to dance no matter your style and/or age, wether you're a amateur or professional...

Saturday, Jul 4

Macy's 4th of July Fireworks
Jul 4, 9:00pm at Hudson River Park at Pier..., New York
New York, NY. April 29, 2009. Macy's 4th of July Fireworks' pyrotechnic splendor returns to the Hudson River for the first time since 2000 to celebrate America's birthday with the nation's largest...

Figure 9.1: Mock-up illustration of an event search and browsing system.

While we developed key techniques for enabling important applications such as event browsing and search, which rely on organized, timely, and multifaceted event data from social media sites, there are many remaining opportunities for future work, to optimize, connect, and extend the event identification and characterization techniques described in this dissertation. We outline some of these directions for future work next.

9.1 Clustering Framework Optimization

The clustering framework developed in this dissertation (Section 4.1) is at the core of our unknown event identification techniques and its results directly affect the events that we are able to discover. For this reason, we studied this framework and its alternatives extensively prior to selecting it as part of our solution. Although we experimented with alternative clustering algorithms and parameter settings while training our techniques (see Section 4.1), several optimizations of the clustering framework could be considered for future work. Such optimizations could potentially improve the scalability and enhance the quality of our clustering results.

There are several properties of a suitable clustering approach for the unknown event identification scenario, as we previously discussed (Section 4.1). Specifically, recall that the clustering approach we use should be able to efficiently process large amounts of data incrementally, and not require a priori knowledge of the number of clusters. Although these requirements eliminate many candidate clustering algorithms (e.g., K-Means, agglomerative hierarchical clustering [Ber02]), our chosen algorithm, as described in Section 4.1, is not the only possible solution. As one alternative, we could consider applying several efficiency optimizations to our current single-pass incremental approach, specifically involving blocking or canopy techniques [BKM06; MNU00].

Recently, Reuter et al. [RCD$^+$11] extended our clustering framework (as presented in [BNG10]) to include a blocking technique, which considers only a subset of the already-clustered documents as candidate "nearest-neighbors" for any newly posted document. The "blocker" (i.e., candidate selection function) that they proposed is simply a sliding time window, which selects a subset of the clustered documents that are within an empirically-tuned temporal proximity to the given document. Not surprisingly, this technique was significantly more efficient than the baseline techniques that compute the similarity between all document pairs. Unfortunately, Reuter et al. did not use a centroid representation of each cluster but rather compared any document to all other seen documents in the stream that fit the blocking criterion. An interesting direction to explore is the tradeoff between efficiency and accuracy of the clustering algorithm, using document-document versus document-centroid similarities with appropriate blocking functions. An additional option for the blocking function is a location-based candidate selection technique, using the location proximity between the documents as a rough indicator that the documents correspond to the same event. This location-based blocker might be more effective for certain types of events (e.g., local events) than others (e.g., global breaking news events).

As an alternative to clustering, we could consider approaches for grouping topically similar social media documents using topic models [BNJ03]. Topic models are a class of unsupervised probabilistic modeling techniques [BNJ03] that have grown in popularity over the past several years and which have been used in a variety of applications. Broadly, these models map a text document collection into sets of distributions over words. These distributions serve as representations for the different topics in the document collection. Recently, Hoffman et al. [HBB10] developed scalable, online topic models, for detecting topics in massive document collections. Their algorithm only requires a single pass over the document collection, making it significantly more efficient than previously-proposed batch topic modeling approaches, and, therefore, more suitable for our event identification problem. Unfortunately, the number of topics must be specified as a parameter for this algorithm, and this number is static, unlike the number of clusters in our framework, which may change over time.

Extending online topic models to handle a variable number of topics is one possible solution that would make this alternative approach suitable for our problem. One obvious drawback of topic models for the social media domain is that they associate documents with topics purely based on textual content. As we showed in Chapter 5, learning similarity metrics using a variety of context features, textual and non-textual, is more effective than text-based approaches at determining when social media documents correspond to the same event. Still, topic modeling is an effective technique for characterizing social media content [RDL10], and may prove to be a useful signal for identifying events and their associated social media documents in our unknown identification scenario.

9.2 Identifying Unknown Events with Learned Similarity Metrics Across Sites

In Chapters 4 and 5 we presented two alternative approaches for identifying events and their associated social media documents in the unknown event identification scenario. In Chapter 4, we first clustered social media documents using our proposed clustering framework, and then employed a post-clustering classification step to determine which clusters correspond to events. In that chapter, we used a simple text-based similarity metric as we were focused on social media sites such as Twitter, whose documents consist of short textual messages. In contrast, in Chapter 5 we used our clustering framework with a stream that consists of event documents exclusively. Here, we assumed that non-event documents were filtered prior to clustering. Importantly, we defined multi-feature similarity metrics to handle documents from social media sites such as Flickr, which include a variety of rich and descriptive context features. A natural extension of the work described in these chapters is to unify the main contributions in each chapter, namely, the clustering and classification approach of Chapter 4, and the similarity metric learning techniques of Chapter 5, into an end-to-end unknown event identification framework with learned similarity metrics.

As a potential improvement to the techniques presented in Chapters 4 and 5, this unknown identification framework should be able to incorporate social media documents across sites, as we do in the known event identification scenario (Chapter 6). For this, we may choose to cluster documents from all social media sites simultaneously, or cluster documents from each social media site in isolation, and then merge similar event clusters across sites. If we choose to cluster documents across sites collectively, we must train a new similarity metric to handle a generic representation of social media documents that would fit documents from most social media sites. Fortunately, most social media documents can be represented using a core set of features (Chapter 5). As a challenge, social media documents from sites such as Twitter consist of short segments of text, frequently with no additional context features. To handle such documents, we could consider core context features such as title or location as "missing," in the context of the generic social media document representation. As an alternative, by clustering documents from each site in isolation, we could have the advantage of tailoring the similarity metric to each specific site and potentially exploiting additional, site-specific document features (e.g., video category on YouTube, "people in photos" feature on Flickr).

9.3 Improving Breadth of Event Content

To address our breadth requirement (Chapter 1), we considered several query-based techniques for identifying event documents from different types of social media sites (Chapter 6). We were able to show that we can successfully use identified event content on one site to retrieve additional event content on other sites. This process of bootstrapping examples from one view of the data (e.g., event documents from Flickr) to inform the learning process in a different, complementary view of the data (e.g., event documents from YouTube) is related to the powerful notion of co-training [BM98]. To extend this idea, we could cast the cross-site event identification problem as a semi-supervised clustering problem [Zhu05], where some events, and their associated social media documents, are available to us as seed event clusters. In this scenario, we can use *learned* multi-feature similarity metrics to identify social media documents that are highly similar to any existing event clusters.

Similar to the process we followed in Chapter 6, we can treat social media documents that are highly similar to known event clusters as “ground truth.” As a challenging extension to our work, we can use the newly identified documents to *augment* the event clusters with additional context features that might not have been available initially (e.g., location, title, tags). These additional features, in turn, could help identify new event documents on complementary social media sites. This process repeats, as newly identified event documents become “ground truth” and are iteratively used to identify more documents on other social media sites.

9.4 Ranking Events for Search and Presentation

The events that we identified and characterized, along with their associated social media documents, present many opportunities for improving and complementing search tools that Web search engines provide. Specifically, as we discussed, we can enable social media-augmented event browsing and search, where users are presented with a select set of relevant events, for the search case in response to a keyword query. State-of-the-art search applications rely on this notion of relevance to build domain-appropriate ranking models. Therefore, one challenging direction for future work would be to learn a ranking model for events that would consider the event's relevance for any user and query based on the available social media context features.

Learning ranking models for information retrieval tasks has been the topic of many research efforts over the past several years [BSR+05; CS01; XLW+08; FISS03; LBW07]. These ranking models use discriminative training over labeled examples to capture document relevance by combining possibly revealing features with respect to a global notion of relevance, or as it relates to specific queries and users. Such features may be generated by traditional information retrieval methods such as language modeling [PC98] and PageRank scoring [BP98]. The challenges that must be addressed in order to learn successful ranking models include feature generation, ranking model selection, and labeled data collection for model training and evaluation.

Keyword searches are often ambiguous and their results are not equally relevant for every user. For example, a user in New York searching for the query [concert in the park] may expect a different set of events than a person searching for the same query in San Francisco ("the park" likely meaning Central Park in the first case and Golden Gate Park in the other). Furthermore, we would expect that a New York Philharmonic's concert in the park would be more relevant than a concert in the park by an unknown artist, based on its overall importance (e.g., as measured using link analysis [BP98]). Therefore, it is important to define features that capture the static relevance of an event, as well as its relevance to different queries and users. We could explore three different types of features:

- *Static relevance features*: capture the (query-independent, user-independent) importance of an event.

- *Query relevance features*: capture the relevance of an event for a specific keyword query.

- *Personalized relevance features*: capture the relevance of an event for a specific user.

We elaborate on each of these types of features next.

Static relevance features: We could examine several possible feature indicators of event importance that would help us gauge an event's static relevance. These features reflect an absolute notion of importance, which is not specific to any user or query. For example, a presidential inauguration should have a higher static rank than a speech by a local politician. Yet, in some cases, the relative importance is not as clear (e.g., the New York City Marathon versus the Macy's Thanksgiving Day Parade), and, therefore, we could analyze different factors that may be used to infer the global importance of an event. Specifically, we could consider features such as the number of social media documents associated with the event, the volume of comments, and the distribution of links between these social media documents (e.g., using measures such as PageRank [BP98]). Understandably, local and community-specific events would likely have fewer associated social media documents and, therefore, be deemed as less important than popular, widely-known events, according to this definition of importance. However, features based on user-specific and query-specific relevance, which could be combined with the static relevance features using a trained ranking model, could help adjust the predicted rank of these events in cases where they may be of relevance to the specific user and query.

Query relevance features: Determining the relevance of an event with respect to a query is a challenging task, with some queries containing ambiguous keywords (e.g., [police concert 2011]), or simply not containing enough information (e.g., [www] referring to the World Wide Web conference). For our query relevance features, we could experiment with the traditional information retrieval models such as Okapi BM25 [RW99], language models [PC98], and others [MRS08]. Additionally, since each event can be represented using a record of values, some textual and some numeric, we could explore techniques for parsing the query to extract and compare these structured features. For instance, for the keyword query [macworld expo SF 2010], we would want to recognize that 2010 is the year, SF is the location, and macworld expo is the title. We can compute the similarity along the event features using the appropriate similarity metrics as described in Chapter 5. Finally, we may consider representing each event using the content from all of its documents simultaneously, or extract the query relevance features for each document separately, and report aggregate statistics as the event-level query relevance features (the latter approach being less scalable but possibly revealing).

Personalized relevance features: The personalized rank features reflect the importance of the event for a particular user. These features could be particularly important in the context of event browsing, where a query is not provided. Our ability to capture this importance would improve with the amount of information that we have about the user. For example, a New York Philharmonic concert in Central Park should be ranked higher for a person who lives in New York than for a person who lives in California. Therefore, it would be useful to include a feature that would indicate the geographical proximity of the user to the event. We could also use signals from a user's social network to compute revealing personalized relevance features. For example, if the user's social network contacts search for or browse event content relating to a Columbia University commencement ceremony, this activity could provide an indication that this event would also be of interest to the user. We could consider features such as the number (or percentage) of social network connections that show interest in the event via browsing, search, or comments on associated social media documents.

To train a ranking model using these features, we could explore different ways of collecting labeled relevance judgments for document-query pairs and document-user pairs using several user profiles. These relevance judgments may be obtained explicitly using human annotators, or implicitly by analyzing user interaction with ranked results through clickthrough behavior [RJ05]. Obtaining labels implicitly using clickthrough behavior is an easy way to generate large amounts of training data, but this data may be noisy and influenced by contextual factors such as the position of the item on the search results page [BMC07]. While employing expert human annotators is often expensive, we could explore the use of services such as Amazon's Mechanical Turk that can efficiently distribute this type of labeling task to many of its workers.

We could consider a variety of techniques for training a ranking model. With so many existing models in the literature, researchers have created benchmarking datasets for competitive evaluation of ranking techniques [LXQ+07]. We could explore suitable ranking models, using these benchmark datasets to identify techniques that optimize quality and scalability. We could train ranking models that combine the three different types of features, namely static, query, and personalized relevance features, to enable event search. Alternatively, for event browsing, where no query is given, we could train a separate ranking model using static and personalized features alone.

In summary, this dissertation presents a variety of useful techniques for event-based organization of social media content, to enable timely exploration and interaction with this rich and valuable source of event information. Specifically, we provide important insights regarding the types of events that exist in social media and the characteristics of their associated content. We develop key methods for identifying different types of events and their associated social media documents under two alternative scenarios, namely, the unknown identification scenario, where we identify trending events and their associated documents using an online clustering framework, and the known identification scenario, where we identify social media documents for planned events using query formulation and retrieval strategies. Since the number of identified social media documents for an event may be large and, therefore, difficult for users to explore in its entirety, we also design techniques for selecting a subset of high-quality, relevant event documents that reflect useful information for each event. Overall, this dissertation provides a framework for studying events in social media, and offers contributions for improving the utility of social media content through event identification, characterization, and content selection. Promising directions for future work could build on these contributions to provide a powerful interface to user-contributed event content, reflecting the rich and diverse points of view of social media users around the world.

Appendix A

Normalized Mutual Information and V-Measure

In our discussion of clustering evaluation metrics (Section 5.1.2) we described Normalized Mutual Information (NMI) [SGC02], a frequently-used measure of quality for clustering results. In this appendix, we present a proof that NMI is equivalent to a recently proposed clustering quality metric called V-measure [RH07], which represents the harmonic mean of two desirable criteria for a clustering solution: homogeneity and completeness. Homogeneity of a clustering solution reflects the degree to which documents in any single cluster correspond to one event. Completeness of a clustering solution reflects the degree to which all documents that correspond to one event are in a single cluster. Homogeneity and completeness directly reflect the clustering properties that we aim to optimize (Section 5.1.2).

Formally, for a set of clusters $C = \{c_1, \ldots, c_J\}$ and events $E = \{e_1, \ldots e_K\}$, homogeneity h is defined as $h = 1 - \frac{H(E|C)}{H(E)}$ and completeness m is defined as $m = 1 - \frac{H(C|E)}{H(C)}$. Therefore, V-measure is defined as $V = \frac{h \cdot m}{h+m}$. Substituting the definitions of h and m, we have:

$$V = \frac{(1 - \frac{H(E|C)}{H(E)}) \cdot (1 - \frac{H(C|E)}{H(C)})}{(1 - \frac{H(E|C)}{H(E)}) + (1 - \frac{H(C|E)}{H(C)})} \tag{A.1}$$

which is equivalent to:

$$V = \frac{(\frac{H(E)-H(E|C)}{H(E)}) \cdot (\frac{H(C)-H(C|E)}{H(C)})}{(\frac{H(E)-H(E|C)}{H(E)}) + (\frac{H(C)-H(C|E)}{H(C)})} \tag{A.2}$$

Since $H(E) - H(E|C) = H(C) - H(C|E)$, we get:

$$V = \frac{\frac{(H(E)-H(E|C))^2}{H(E) \cdot H(C)}}{\frac{(H(E)-H(E|C)) \cdot (H(E)+H(C))}{H(E) \cdot H(C)}} \tag{A.3}$$

and

$$V = \frac{H(E) - H(E|C)}{H(E) + H(C)} \tag{A.4}$$

Finally, since $H(E) - H(E|C) = I(C, E)$, we have:

$$V = \frac{I(C, E)}{H(E) + H(C)} = NMI(C, E) \tag{A.5}$$

By showing that NMI is equivalent to the harmonic mean of homogeneity and completeness, we can reason about the quality of our clustering solution in terms of each component separately, thus making our results more interpretable than when using the combined metric alone. This proof unifies two frequently-used clustering quality metrics, which would hopefully enable a direct comparison between clustering results evaluated according to these metrics.

Bibliography

[ACD$^+$98] James Allan, Jaime Carbonell, George Doddington, Jonathan Yamron, and Yiming Yang. Topic detection and tracking pilot study final report. In *Proceedings of Broadcast News Transcription and Understanding Workshop*, 1998.

[AGAV08] Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information Retrieval*, 2008.

[AH10] Sitaram Asur and Bernardo A. Huberman. Predicting the future with social media. Technical Report HPL-2010-53, HP Laboratories, 2010.

[Ahn06] David Ahn. The stages of event extraction. In *Proceedings of the COLING/ACL 2006 Workshop on Annotating and Reasoning about Time and Events (ARTE '06)*, 2006.

[All02] James Allan, editor. *Topic Detection and Tracking: Event-based Information Organization*. Kluwer Academic Publisher, 2002.

[APL98] James Allan, Ron Papka, and Victor Lavrenko. On-line new event detection and tracking. In *Proceedings of the 21st ACM International Conference on Research and Development in Information Retrieval (SIGIR '98)*, 1998.

[BBS05] Mikhail Bilenko, Sugato Basu, and Mehran Sahami. Adaptive product normalization: Using online learning for record linkage in comparison shopping. In *Proceedings of the Fifth IEEE International Conference on Data Mining (ICDM '05)*, 2005.

[BCI$^+$11] Hila Becker, Feiyang Chen, Dan Iter, Mor Naaman, and Luis Gravano. Automatic identification and presentation of Twitter content for planned events. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM '11)*, 2011.

[Ber02] Pavel Berkhin. Survey of clustering data mining techniques. Technical report, Accrue Software, 2002.

[BHB11] Edward Benson, Aria Haghighi, and Regina Barzilay. Event discovery in social media feeds. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT '11)*, 2011.

[BING11] Hila Becker, Dan Iter, Mor Naaman, and Luis Gravano. Identifying content for planned events across social media sites. Submitted for publication, 2011.

[BKM06] Mikhail Bilenko, Beena Kamath, and Raymond J. Mooney. Adaptive blocking: Learning to scale up record linkage. In *Proceedings of the Sixth IEEE International Conference on Data Mining (ICDM '06)*, 2006.

[BM98] Avrim Blum and Tom Mitchell. Combining labeled and unlabeled data with co-training. In *Proceedings of the 11th Annual Conference on Computational Learning Theory (COLT '98)*, 1998.

[BM03] Mikhail Bilenko and Raymond J. Mooney. Adaptive duplicate detection using learnable string similarity measures. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '03)*, 2003.

[BMC07] Hila Becker, Christopher Meek, and David Maxwell Chickering. Modeling contextual factors of click rates. In *Proceedings of the 22nd AAAI Conference on Artificial Intelligence (AAAI '07)*, 2007.

[BNG10] Hila Becker, Mor Naaman, and Luis Gravano. Learning similarity metrics for event identification in social media. In *Proceedings of the Third ACM International Conference on Web Search and Data Mining (WSDM '10)*, 2010.

[BNG11a] Hila Becker, Mor Naaman, and Luis Gravano. Beyond trending topics: Real-world event identification on Twitter. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM '11)*, 2011.

[BNG11b] Hila Becker, Mor Naaman, and Luis Gravano. Selecting quality Twitter content for events. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM '11)*, 2011.

[BNJ03] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[BP98] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. In *Proceedings of the 7th International World Wide Web Conference (WWW7)*, 1998.

[BSR$^+$05] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd International Conference on Machine Learning (ICML '05)*, 2005.

[BW03] Susanne Boll and Utz Westermann. MediAEther: An event space for context-aware multimedia experiences. In *Proceedings of the 2003 ACM SIGMM Workshop on Experiential Telepresence (ETP '03)*, 2003.

[BXNG10] Hila Becker, Bai Xiao, Mor Naaman, and Luis Gravano. Exploiting social links for event identification in social media. In *Proceedings of the Third Annual Workshop on Search in Social Media (SSM '10)*, 2010.

[CKM09] Zhaoqi Stella Chen, Dmitri V. Kalashnikov, and Sharad Mehrotra. Exploiting context analysis for combining multiple entity resolution systems. In *Pro-*

ceedings of the 2009 ACM International Conference on Management of Data (SIGMOD '09), 2009.

[CL09] Marc Cheong and Vincent Lee. Integrating Web-based intelligence retrieval and decision-making from the Twitter trends knowledge base. In *Proceeding of the Second ACM Workshop on Social Web Search and Mining (SWSM '09)*, 2009.

[CMS09] W. Bruce Croft, Donald Metzler, and Trevor Strohman. *Search Engines: Information Retrieval in Practice*. Addison-Wesley Publishing Company, 2009.

[CP11] Deepayan Chakrabarti and Kunal Punera. Event summarization using tweets. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM '11)*, 2011.

[CR02] William W. Cohen and Jacob Richman. Learning to match and cluster large high-dimensional data sets for data integration. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '02)*, 2002.

[CR09] Ling Chen and Abhishek Roy. Event detection from Flickr data through wavelet-based spatial analysis. In *Proceedings of the 18th ACM International Conference on Information and Knowledge Management (CIKM '09)*, 2009.

[CS01] Koby Crammer and Yoram Singer. Pranking with ranking. In *Advances in Neural Information Processing Systems 14 (NIPS '01)*, 2001.

[CS08] Riley Crane and Didier Sornette. Robust dynamic classes revealed by measuring the response function of a social system. *Proceedings of the National Academy of Sciences*, 105(41):15649–15653, 2008.

[DAR09] Carlotta Domeniconi and Muna Al-Razgan. Weighted cluster ensembles: Methods and analysis. *ACM Transactions on Knowledge Discovery from Data*, 2(4):1–40, 2009.

[DDGR07] Abhinandan S. Das, Mayur Datar, Ashutosh Garg, and Shyam Rajaram. Google news personalization: Scalable online collaborative filtering. In *Proceedings of the 16th International World Wide Web Conference (WWW '07)*, 2007.

[Dem06] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

[DI08] Wisam Dakka and Panagiotis G. Ipeirotis. Automatic extraction of useful facet hierarchies from text databases. In *Proceedings of the IEEE 24th International Conference on Data Engineering (ICDE '08)*, 2008.

[Diw03] Urmila M. Diwekar. *Introduction to Applied Optimization.* Springer, 2003.

[DK92] Daniel Dayan and Elihu Katz. *Media Events: The Live Broadcasting of History.* Harvard University Press, 1992.

[DKJ$^+$07] Jason V. Davis, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S. Dhillon. Information-theoretic metric learning. In *Proceedings of the 24th International Conference on Machine Learning (ICML'07)*, 2007.

[DLSL09] Bertrand De Longueville, Robin S. Smith, and Gianluca Luraschi. "OMG, from here, I can see the flames!": A use case of mining location based social networks to acquire spatio-temporal data on forest fires. In *Proceedings of the 2009 International Workshop on Location Based Social Networks (LBSN '09)*, 2009.

[DNKS10] Nicholas Diakopoulos, Mor Naaman, and Funda Kivran-Swaine. Diamonds in the rough: Social media visual analytics for journalistic inquiry. In *Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST '10)*, 2010.

[DS10] Nicholas Diakopoulos and David A. Shamma. Characterizing debate performance via aggregated Twitter sentiment. In *Proceedings of the 28th International Conference on Human Factors in Computing Systems (CHI '10)*, 2010.

[ER04] Günes Erkan and Dragomir R. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *JAIR*, 22(1):457–479, 2004.

[Eve02] Events, 2002. In Stanford Encyclopedia of Philosophy. Retrieved June 2nd, 2010 from http://plato.stanford.edu/entries/events/.

[FH03] Elena Filatova and Vasileios Hatzivassiloglou. Domain-independent detection, extraction, and labeling of atomic events. In *Proceedings of Recent Advances in Natural Language Processing (RANLP '03)*, 2003.

[FHM06] Elena Filatova, Vasileios Hatzivassiloglou, and Kathleen McKeown. Automatic creation of domain templates. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics (ACL-COLING '06)*, 2006.

[FISS03] Yoav Freund, Raj Iyer, Robert E. Schapire, and Yoram Singer. An efficient boosting algorithm for combining preferences. *Journal of Machine Learning Research*, 4:933–969, 2003.

[GGLNT04] Daniel Gruhl, R. Guha, David Liben-Nowell, and Andrew Tomkins. Information diffusion through blogspace. In *Proceedings of the 13th International World Wide Web Conference (WWW '04)*, 2004.

[GHT04] Natalie S. Glance, Matthew Hurst, and Takashi Tomokiyo. BlogPulse: Automated trend discovery for Weblogs. In *Proceedings of the WWW 2004 Workshop on the Weblogging Ecosystem*, 2004.

[GMT05] Aristides Gionis, Heikki Mannila, and Panayiotis Tsaparas. Clustering aggregation. In *Proceedings of the 21st International Conference on Data Engineering (ICDE '05)*, 2005.

[Gri10] Ralph Grishman. The impact of task and corpus on event extraction systems. In *Proceedings of the Seventh Conference on International Language Resources and Evaluation (LREC '10)*, 2010.

[GSFW94] Deborah J. Gerner, Philip A. Schrodt, Ronald A. Francisco, and Judith L. Weddle. Machine coding of event data using regional and international sources. *International Studies Quarterly*, 38(1):91–119, 1994.

[HBB10] Matthew Hoffman, David Blei, and Francis Bach. Online learning for latent Dirichlet allocation. In *Advances in Neural Information Processing Systems 23 (NIPS '10)*, 2010.

[HCL07] Qi He, Kuiyu Chang, and Ee-Peng Lim. Analyzing feature trajectories for event detection. In *Proceedings of the 30th ACM International Conference on Research and Development in Information Retrieval (SIGIR '07)*, 2007.

[HD10] Liangjie Hong and Brian D. Davison. Empirical study of topic modeling in Twitter. In *Proceedings of the First Workshop on Social Media Analytics (SOMA '10)*, 2010.

[HGM00] Vasileios Hatzivassiloglou, Luis Gravano, and Ankineedu Maganti. An investigation of linguistic features and clustering algorithms for topical document clustering. In *Proceedings of the 23rd ACM International Conference on Research and Development in Information Retrieval (SIGIR '00)*, 2000.

[HHSC11] Brent Hecht, Lichan Hong, Bongwon Suh, and Ed H. Chi. Tweets from Justin Bieber's heart: The dynamics of the "location" field in user profiles. In *Proceedings of the 29th International Conference on Human Factors in Computing Systems (CHI '11)*, 2011.

[HS95] Mauricio A. Hernández and Salvatore J. Stolfo. The merge/purge problem for large databases. In *Proceedings of the 1996 ACM International Conference on Management of Data (SIGMOD '96)*, 1995.

[JG08] Heng Ji and Ralph Grishman. Refining event extraction through cross-document inference. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT '08)*, 2008.

[JZSC09] Bernard J. Jansen, Mimi Zhang, Kate Sobel, and Abdur Chowdury. Twitter power: Tweets as electronic word of mouth. *Journal of the American Society for Information Science and Technology*, 60(11):2169–2188, 2009.

[KA04] Giridhar Kumaran and James Allan. Text classification and named entities for new event detection. In *Proceedings of the 27th ACM International Conference on Research and Development in Information Retrieval (SIGIR '04)*, 2004.

[KAKS97] George Karypis, Rajat Aggarwal, Vipin Kumar, and Shashi Shekhar. Multilevel hypergraph partitioning: Application in VLSI domain. In *Proceedings of the 34th ACM Conference on Design Automation (DAC '97)*, 1997.

[KGP$^+$04] April Kontostathis, Leon Galitsky, William M. Pottenger, Soma Roy, and Daniel J. Phelps. A survey of emerging trend detection in textual data mining. In Michael W. Berry, editor, *A Comprehensive Survey of Text Mining - Clustering, Classification and Retrieval.* Springer, 2004.

[Kle03] Jon Kleinberg. Bursty and hierarchical structure in streams. *Data Mining and Knowledge Discovery*, 7:373–397, 2003.

[KLPM10] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is Twitter, a social network or a news media? In *Proceedings of the 19th International World Wide Web Conference (WWW '10)*, 2010.

[KMC05] Reiner Kraft, Farzin Maghoul, and Chi Chao Chang. Y!Q: Contextual search at the point of inspiration. In *Proceedings of the 14th ACM International Conference on Information and Knowledge Management (CIKM '05)*, 2005.

[LBK09] Jure Leskovec, Lars Backstrom, and Jon M. Kleinberg. Meme-tracking and the dynamics of the news cycle. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '09)*, 2009.

[LBW07] Ping Li, Chris Burges, and Qiang Wu. McRank: Learning to rank using multiple classification and gradient boosting. In *Advances in Neural Information Processing Systems 20 (NIPS '07)*, 2007.

[LPS$^+$08] Sophia Liu, Leysia Palen, Jeannette Sutton, Amanda Hughes, and Sarah Vieweg. In search of the bigger picture: The emergent role of on-line photo-sharing in times of disaster. In *Proceedings of the Information Systems for Crisis Response and Management Conference (ISCRAM '08)*, 2008.

[LS99] M. LeCompte and J. Schensul. *Designing and conducting ethnographic research*. Altamira Press, Walnut Creek, CA, 1999.

[LXQ$^+$07] Tie-Yan Liu, Jun Xu, Tao Qin, Wenying Xiong, and Hang Li. LETOR: Benchmark dataset for research on learning to rank for information retrieval. In *Proceedings of SIGIR 2007 Workshop on Learning to Rank for Information Retrieval*, 2007.

[Mak03] Juha Makkonen. Investigations on event evolution in TDT. In *Proceedings of HLT-NAACL Student Workshop of Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, Edmonton, Canada, 2003.

[Mak09] Juha Makkonen. *Semantic Classes in Topic Detection and Tracking*. PhD thesis, University of Helsinki, Department of Computer Science, 2009.

[MAMS04] Juha Makkonen, Helena Ahonen-Myka, and Marko Salmenkivi. Simple semantics in topic detection and tracking. *Information Retrieval*, 7(3–4):347–368, 2004.

[MBN07] Fabian Moerchen, Klaus Brinker, and Claus Neubauer. Any-time clustering of high frequency news streams. In *Proceedings of the ACM SIGKDD Data Mining Case Studies Workshop (DMCS '07)*, 2007.

[MNU00] Andrew McCallum, Kamal Nigam, and Lyle H. Ungar. Efficient clustering of high-dimensional data sets with application to reference matching. In *Pro-*

ceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '00), 2000.

[MRS08] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval.* Cambridge Univ. Press, 2008.

[NBG11] Mor Naaman, Hila Becker, and Luis Gravano. Hip and trendy: Characterizing emerging trends on Twitter. *Journal of the American Society for Information Science and Technology*, 62(5):902–918, 2011.

[NBL10] Mor Naaman, Jeffrey Boase, and Chih-Hui Lai. Is it really about me?: Message content in social awareness streams. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work (CSCW '10)*, 2010.

[NGS$^+$09] Meenakshi Nagarajan, Karthik Gomadam, Amit P. Sheth, Ajith Ranabahu, Raghava Mutharaju, and Ashutosh Jadhav. Spatio-temporal-thematic analysis of citizen sensor data: Challenges and experiences. In *Proceedings of the 10th International Conference on Web Information Systems Engineering (WISE '09)*, 2009.

[OBRS10] Brendan OConnor, Ramnath Balasubramanyan, Bryan R. Routledge, and Noah A. Smith. From tweets to polls: Linking text sentiment to public opinion time series. In *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media (ICWSM '10)*, 2010.

[OKA10] Brendan OConnor, Michel Krieger, and David Ahn. TweetMotif: Exploratory search and topic summarization for Twitter. In *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media (ICWSM '10)*, 2010.

[Osb02] Jason Osborne. Notes on the use of data transformations. *Practical Assessment, Research and Evaluation*, 8(6), 2002.

[PC98] Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st ACM International Conference on Research and Development in Information Retrieval (SIGIR '98)*, 1998.

[PMS09] Owen Phelan, Kevin McCarthy, and Barry Smyth. Using Twitter to recommend real-time topical news. In *Proceedings of the Third ACM Conference on Recommender Systems (RecSys '09)*, 2009.

[POL10] Saša Petrović, Miles Osborne, and Victor Lavrenko. Streaming first story detection with application to Twitter. In *Proceedings of the 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL '10)*, 2010.

[RCD$^+$11] Timo Reuter, Philipp Cimiano, Lucas Drumond, Krisztian Buza, and Lars Schmidt-Thieme. Scalable event-based clustering of social media via record linkage techniques. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM '11)*, 2011.

[RDL10] Daniel Ramage, Susan Dumais, and Dan Liebling. Characterizing microblogs with topic models. In *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media (ICWSM '10)*, 2010.

[RGN07] Tye Rattenbury, Nathaniel Good, and Mor Naaman. Towards automatic extraction of event and place semantics from Flickr tags. In *Proceedings of the 30th ACM International Conference on Research and Development in Information Retrieval (SIGIR '07)*, 2007.

[RH07] Andrew Rosenberg and Julia Hirschberg. V-Measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP '07)*, 2007.

[RJ05] Filip Radlinski and Thorsten Joachims. Query chains: Learning to rank from implicit feedback. In *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '05)*, 2005.

[RW99] Stephen E. Robertson and Steve Walker. Okapi/Keenbow at TREC-8. In *Proceedings of the 14th Text REtrieval Conference (TREC-8)*, 1999.

[SGC02] Alexander Strehl, Joydeep Ghosh, and Claire Cardie. Cluster ensembles – A knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3:583–617, 2002.

[SHK10] Beaux Sharifi, Mark-Anthony Hutton, and Jugal K. Kalita. Experiments in microblog summarization. In *Proceedings of the 2010 IEEE Second International Conference on Social Computing (SOCIALCOM '10)*, 2010.

[Sin84] Roger W. Sinnott. Virtues of the Haversine. *Sky and Telescope*, 68:159, 1984.

[SJ10] Vivek K. Singh and Ramesh Jain. Structural analysis of the emerging event-web. In *Proceedings of the 19th International World Wide Web Conference (WWW '10)*, 2010.

[SKC10] David A. Shamma, Lyndon Kennedy, and Elizabeth Churchill. Statler: Summarizing media through short-message services. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work (CSCW '10)*, 2010.

[SOM10] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. Earthquake shakes Twitter users: Real-time event detection by social sensors. In *Proceedings of the 19th International World Wide Web Conference (WWW '10)*, 2010.

[SPHV10] Kate Starbird, Leysia Palen, Amanda L. Hughes, and Sarah Vieweg. Chatter on the red: What hazards threat reveals about the social life of microblogged information. In *Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work (CSCW '10)*, 2010.

[SPS08] Irina Shklovski, Leysia Palen, and Jeannette Sutton. Finding community through information and communication technology in disaster response. In *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work (CSCW '08)*, 2008.

[SST$^+$09] Jagan Sankaranarayanan, Hanan Samet, Benjamin E. Teitler, Michael D. Lieberman, and Jon Sperling. Twitterstand: News in tweets. In *Proceedings of the 17th ACM International Conference on Advances in Geographic Information Systems (GIS '09)*, 2009.

[TBW95] Stelios C. A. Thomopoulos, Dimitrios K. Bougoulias, and Chin-Der Wann. Dignet: An unsupervised-learning clustering algorithm for clustering and data fusion. *IEEE Transactions on Aerospace Electronic Systems*, 31:21–38, 1995.

[TdRW11] Manos Tsagkias, Maarten de Rijke, and Wouter Weerkamp. Linking online news and social media. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining (WSDM '11)*, 2011.

[WF94] Stanley Wasserman and Katherine Faust. *Social network analysis: Methods and applications.* Cambridge University Press, 1994.

[WF05] Ian H. Witten and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques.* Morgan Kaufmann, 2005.

[WL11] Jianshu Weng and Francis Lee. Event detection in Twitter. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM '11)*, 2011.

[WZHS07] Xuanhui Wang, ChengXiang Zhai, Xiao Hu, and Richard Sproat. Mining correlated bursty topic patterns from coordinated text streams. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '07)*, 2007.

[XLW$^+$08] Fen Xia, Tie-Yan Liu, Jue Wang, Wensheng Zhang, and Hang Li. Listwise approach to learning to rank: Theorem and algorithm. In *Proceedings of the 25th International Conference on Machine Learning (ICML '08)*, 2008.

[XNJR02] Eric P. Xing, Andrew Y. Ng, Michael I. Jordan, and Stuart Russell. Distance metric learning, with application to clustering with side-information. In *Advances in Neural Information Processing Systems 15 (NIPS '02)*, 2002.

[Yb10] Sarita Yardi and danah boyd. Tweeting from the town square: Measuring geographic local networks. In *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media (ICWSM '10)*, 2010.

[YCB+99] Yiming Yang, Jaime Carbonell, Ralf Brown, Thomas Pierce, Brian T. Archibald, and Xin Liu. Learning approaches for detecting and tracking news events. *IEEE Intelligent Systems Special Issue on Applications of Intelligent Information Retrieval*, 14(4):32 – 43, 1999.

[YPC98] Yiming Yang, Thomas Pierce, and Jaime Carbonell. A study on retrospective and on-line event detection. In *Proceedings of the 21st ACM International Conference on Research and Development in Information Retrieval (SIGIR '98)*, 1998.

[Zhu05] Xiaojin Zhu. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences Department, University of Wisconsin-Madison, 2005.

[ZJH+11] Wayne Xin Zhao, Jing Jiang, Jing He, Yang Song, Palakorn Achananuparp, Ee-Peng Lim, and Xiaoming Li. Topical keyphrase extraction from Twitter. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT '11)*, 2011.

[ZRL96] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. BIRCH: An efficient data clustering method for very large databases. In *Proceedings of the 1996 ACM International Conference on Management of Data (SIGMOD '96)*, 1996.

[ZT01] Jeffrey M. Zacks and Barbara Tversky. Event structure in perception and conception. *Psychological Bulletin*, 127, 2001.

[ZZW07] Kuo Zhang, Juan Zi, and Li Gang Wu. New event detection based on indexing-tree and named entity. In *Proceedings of the 30th ACM International Conference on Research and Development in Information Retrieval (SIGIR '07)*, 2007.