ABSTRACT

Title of dissertation: DISCOVERING CREDIBLE EVENTS IN NEAR REAL TIME FROM SOCIAL MEDIA STREAMS

Cody Buntain, Doctor of Philosophy, 2016

Dissertation directed by: Professor Jennifer Golbeck
School of Information

Recent reliance on social media platforms as major sources of news and information, both for journalists and the larger population and especially during times of crisis, motivate the need for better methods of identifying and tracking high-impact events in these social media streams. Social media's volume, velocity, and democratization of information (leading to limited quality controls) complicate rapid discovery of these events and one's ability to trust the content posted about these events. This dissertation addresses these complications in four stages, using Twitter as a model social platform. The first stage analyzes Twitter's response to major crises, specifically terrorist attacks in Western countries, showing these high-impact events do not significantly impact message or user volume. Instead, these events drive changes in Twitter's topic distribution, with conversation, retweets, and hashtags relevant to these events experiencing significant, rapid, and short-lived bursts in frequency. Furthermore, conversation participants tend to prefer information from local authorities/organizations/media over national or international sources, with accounts for local police or local newspapers often emerging as central in the networks of inter-

action. Building on these results, the second stage in this dissertation presents and evaluates a set of features that capture these topical bursts associated with crises by modeling bursts in frequency for individual tokens in the Twitter stream. The resulting streaming algorithm is capable of discovering notable moments across a series of major sports competitions using Twitter's public stream without relying on domain- or language-specific information or models. Furthermore, results demonstrate models trained on sporting competition data perform well when transferred to earthquake identification. This streaming algorithm is then extended in this dissertation's third stage to support real-time event tracking and summarization. This real-time algorithm leverages new distributed processing technology to operate at scale and is evaluated against a collection of other community-developed information retrieval systems, where it performs comparably. Further experiments also show this real-time burst detection algorithm can be integrated with these other information retrieval systems to increase overall performance. The final stage then investigates automated methods for evaluating credibility in social media streams by leveraging two existing data sets. These two data sets measure different types of credibility (veracity versus perception), and results show veracity is negatively correlated with the amount of disagreement in and length of a conversation, and perceptions of credibility are influenced by the amount of links to other pages, shared media about the event, and the number of verified users participating in the discussion. Contributions made across these four stages are then usable in the relatively new fields of computational journalism and crisis informatics, which seek to improve news gathering and crisis response by leveraging new technologies and data sources like machine

learning and social media.

DISCOVERING CREDIBLE EVENTS IN NEAR REAL TIME FROM SOCIAL MEDIA STREAMS

by

Cody Buntain

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2016

Advisory Committee:
Professor Jennifer Golbeck, Chair/Advisor
Professor Jimmy Lin
Professor Nicholas Diakopoulos
Professor Hector Corrada Bravo
Professor V.S. Subrahmanian

ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 – 1346

Dedication

To the two most important people in my life: Leslie and Leigh. Their support, encouragement, and patience cannot be adequately thanked.

Acknowledgments

This work simply would not exist without the encouragement of my advisor, Jennifer Golbeck. Her support and trust allowed me to find my own path as a scientist, and she was always there with a light hand to guide me when I became too lost or discouraged. Jen has been and will continue to be a role model for me in my academic career.

I have benefitted from the insight of several other mentors over the course of this work as well. This research owes a great debt to Jimmy Lin, without whose assistance would have made for a more difficult, slow, and expensive effort. His allowances for my access to large stores of data, state-of-the-art processing infrastructure, and a community of researchers contributed significantly to the investigations detailed herein and spurred my interest in large-scale data science.

A special thanks is also owed to the remaining members of my committee: Nick Diakopoulos, Hector Corrada Bravo, and V.S. Subrahmanian. Their excellent questions, insights, and suggestions contributed greatly to this document's final form.

Given this research's focus on social networks, I would be remiss not to thank my own social network, whose role as an intellectual sounding board and emotional support cannot be understated. Thanks to Phil, Robin, Sana, both Steves, Jay, Matt, Brenna, Kris, both Amandas, Adam, Greg, and Christine. Likewise, the trust and collaboration I received from Sandy and Michael Ring and my previous colleagues at Pikewerks contributed a great deal to my growth as a scientist and

researcher.

Finally, I want to thank my family: my mother, my sister, and Leigh for their continued encouragement and patience. Without my mother's incredible efforts to raise my sister and I as a single mother, none of this would be possible. Thank you.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

AMT	Amazon Mechanical Turk
AP	Associated Press
API	Application Programming Interface
AUC	Area Under the Curve
ELG	Expected Latency Gain
FIFA	Fédération Internationale de Football Association
HCI	Human-Computer Interaction
LDA	Latent Dirichlet Allocation
LSH	Locality-Sensitive Hashing
MAD	Median Absolute Deviation
MLB	Major League Baseball
nCG	Normalized Cumulative Gain
nDCG	Normalized Discounted Cumulative Gain
NFL	National Football League
NHL	National Hockey League
NIST	National Institute for Standards and Technology
PET	Popular Event Tracking
RBF	Radial Basis Function
RF	Random Forest
ROC	Receiver Operating Characteristic
SVM	Support Vector Machine
TDT	Topic Detection and Tracking
TREC	TExt Retrieval Conference
UTC	Coordinated Universal Time

Chapter 1: Introduction

Social media's ubiquity has had a profound effect on the ways in which ordinary people share experiences and information, from mundane photographs of last night's dinner to breaking news of earthquakes, terrorist attacks, and real-time coverage of protests or mass demonstrations. While a significant portion of this content might discuss today's outfit or the pinnacle of Justin Bieber's artistry, social media coverage of high-impact events like disasters and social unrest has become an important resource for Internet users. Social media's impact is evidenced not just by this popularity though; entire nations have restricted or completely severed access to platforms like Twitter as a means to control social unrest (as has occurred in both Egypt and Turkey). Despite social media's potential, it is difficult to identify these high-impact events as they occur, follow new developments as events unfold, and differentiate between credible and dubious content.

Events surrounding the Boston Marathon bombings in April of 2013 provide a microcosmic view of social media's potential for social good while simultaneously hindering accurate news consumption through noise and the spread of rumorous information. For social good, Cassa et al. showed a significant number of Twitter messages, or tweets, posted immediately after the explosions contained valuable

information on severity and location that could guide first responders [1]. A Pew research study found nearly a quarter of all Americans (and more than half of Americans between the ages of 18 and 29) followed coverage of the Boston Marathon bombing on social media platforms like Twitter or Facebook [2]. The Boston Police Department (BPD) was lauded for its use of Twitter to interact with this massive audience "to keep the public informed about the status of the investigation, to calm nerves and request assistance, to correct mistaken information reported by the press, and to ask for public restraint in the tweeting of information from police scanners" [3]. At the same time, the shear volume of social media activity made it difficult to follow new developments, as celebrities, their hordes of followers, and other outsiders shared their sympathies [4].

To make matters worse, no guarantees exist about the validity or truth of information posted to social media, and a portion of the information concerning the Boston Marathon bombing in both social and traditional media was incorrect, ranging from the number of bombs found in Boston to whether suspects had been arrested [5]. Even if one only consumed information from trusted social media sources like the Associated Press (AP), event these highly-qualified sources may report inaccurate information, as many examples in Boston showed. Furthermore, two weeks after the Boston Marathon bombing, hackers compromised the AP's Twitter account and posted a message saying the White House had been attacked, and the president was injured. Not only was this tweet shared by over 4,000 users, automated financial systems that process Twitter data picked up the information and caused a "flash crash" in financial markets, resulting in a loss of over one billion

dollars in assets [6]. It is therefore dangerous to assume all (or even many) posts on social media provide true and accurate information, and yet many users and organizations have little recourse to verify this information and often take it as true. The Boston Marathon bombing is also an outlier in this area in that many journalists and media organizations were already present to cover the Boston Marathon; in many cases, journalistic resources are not already present when a disaster strikes, further complicating reliable coverage.

Given the rapid dissemination and democratization of information via social networks *and* the growing population who rely on these sources for up-to-date news, a clear need exists for methods that distill or extract newsworthy and credible information from social media streams. This dissertation presents research to address these needs, covering social media's response to high-impact events, algorithms for streaming and real-time event detection/summarization, and methods for assessing credibility in topical threads of social media messages. Therefore, research contained herein shows:

> **By integrating machine learning and high-volume streams from social media, one can detect high-impact events, describe those events, and evaluate credibility of those occurrences in near real time.**

1.1 Contributions

Researchers have tried to address these needs through a variety of approaches over the past several decades, but social media's volume, velocity, and limited accountability present a slew of advantages and unique challenges. For event detection, researchers have either used retrospective systems that look into the past (of limited utility for up-to-date news) or have tracked predefined keyword frequencies. While such systems are limited by the assumption that one can know the type, location, or language of an event of interest a priori. Similarly, researchers have sought to address credibility issues in social media by anchoring trust in journalists or accounts with large numbers of followers [7–11], but many useful pieces of information are posted by non-authoritative users [12]. Assessing credibility or trustworthiness in or near real time, however, still represents a gap in the literature as many of these efforts assume either known event information and/or retrospective analysis. Social media content comes with a huge variety of features that could support credibility analysis, and while some of these indicators may already have been explored in these works, latitude exists for integrating machine learning techniques with features like network structure and multimedia. These gaps combined with preliminary work in streaming event detection and trust analysis suggest constructing systems for detecting credible events rapidly is feasible.

Performing this research relies on access to large volumes of social media data, and given the increasing value of this data to advertisers, other commercial entities, and governments, such large data sets can be difficult to obtain. To this end, this

work focuses primarily on the Twitter platform as a model system. While Twitter is an order of magnitude smaller than the largest social media platform, i.e., Facebook, Twitter is one of the most popular social media platforms, maintains a population of over three hundred million monthly active users, and averages over four hundred million posts per day [13]. Most importantly, however, Twitter provides access to a 1%-sample stream of its data to developers for free, making it an appealing and accessible data source.

With these goals and constraints in mind, this research makes the following contributions:

Twitter Response to Terrorist Attacks in Western Countries

As a first step in developing methods for detecting high-impact events in Twitter, one should understand how Twitter responds to these crises. Recent seminal work on crisis informatics in social media investigates the distributions of various types and sources of content following crises (e.g., messages about sympathy, advice/caution, affected individuals, or donations posted by eyewitnesses, government organizations, or media) [4]. Twitter's structural and behavioral responses to these events remains an open question, however, with limited research into Twitter activities like posting, sharing, or following and behaviors like use of hashtags, mentions, and sharing media. This dissertation clarifies these uncertainties through the following contributions:

- Analyzing structural and behavioral responses to terrorist attacks in the gen-

eral Twitter population,

- Comparing Twitter's general response with changes in Twitter activity relevant to the target event (i.e., activity across all of Twitter versus activity mentioning the target event), and
- Identifying trends in Twitter accounts that emerge as central in the social network activity surrounding these events.

Language- and Domain-Agnostic Event Discovery in Social Media Streams

Many research efforts into identifying events in social media presuppose knowledge of event types and other relevant information. For example, if one wanted to track soccer goals in the FIFA World Cup via social media, the typical approach is to track frequencies of a known keyword like "goal" [14]. Such methods work but are inflexible in that they only detect events described by the human-generated seed keywords, a limitation with three important consequences: 1) The system yields limited insight into event context (e.g., who scored the goal or where an event occurred) as event-specific contextual keywords are often omitted from the keyword list to ensure generalizability. 2) Such systems miss *unanticipated* events because the human responsible for seed keywords did not know to include relevant keywords. 3) Lastly, these approaches miss events or information conveyed in languages other than those in the keyword set, implicitly restricting event detection to *only* those languages represented in the seed list. Research introduced in this work addresses

these consequences with three contributions:

- Introducing a streaming algorithm, called LABurst, and feature set for discovering and describing key moments in Twitter's public sample stream without requiring manually defined keywords,
- Evaluating LABurst's performance against a pair of baselines, and
- Demonstrating event detection models trained on sporting events are transferable to disaster response with a case study on identifying earthquakes.

Implementation for Real-Time Event Detection

Of the solutions for detecting events in social media, few are designed to process data streams, even fewer attempt this detection in real time, and of these real-time systems, many suffer from the same domain-specific weaknesses discussed above. To cover this gap in the literature, this work contributes:

- An extension of LABurst, called RTTBurst, to the real-time context,
- The design and implementation of an end-to-end system for ingesting descriptions of user interests and discovering new and relevant tweets using a simple model for identifying bursts in token usage, and
- A set of experiments that integrate this real-time language- and domain-agnostic method with classical information retrieval systems.

Algorithms for Evaluating Social Media Credibility in Near Real Time

As mentioned, it is highly plausible that data taken from social media could be false, misleading, or actively malicious. While research has attempted to assess this lack of credibility in social media, few solutions exist for evaluation in the streaming context. This work extends credibility analysis for social media with the following contributions:

- Leveraging two data sets of credibility and rumor in social media to construct models that can differentiate between credible and non-credible threads of Twitter conversation, and
- An analysis of features for automatically differentiating between credible and non-credible events across multiple data sets.

1.2 Dissertation Roadmap

This dissertation's principal goal is to support a system for identifying and describing high-impact events in social media and assessing their credibility in real time. To that end, this work is constructed in sequential layers, each of which moves closer to this goal.

Chapter 2 establishes the foundations of this work with a discussion of related and prior research into event detection and its applications to social media. Special interest is paid to social media as it pertains to crises (both natural and man-made). The chapter ends with an overview of current and ongoing research into credibility

analysis and rumor detection on Twitter.

Chapter 3 then explores the structural and behavioral responses on Twitter to a collection of terror events in countries with high densities of Twitter users: the 2013 Boston Marathon Bombing in the United States, the 2014 Sydney Hostage Crisis in Australia, the 2015 Charlie Hebdo Shooting and November 2015 Paris attacks in France, and the March 2016 Brussels transit attacks in Belgium. Results in this chapter show, while these events do not significantly impact general Twitter usage, those users who are discussing the event behave in predictable ways across events. These results inform research into crisis-level event detection by identifying potential signals of a crisis-level event and demonstrating the need for event detection mechanisms that operate at the topical level.

Chapter 4 builds on the prior chapter's results and develops LABurst, an algorithm for detecting and describing high-impact, crisis-level events in social media streams based on changes in individual token frequency. Owing to the relative rarity of crisis events, this chapter starts by identifying high-impact moments across three major sporting competitions: the 2013 World Series, the 2014 Super Bowl, and the 2014 World Cup. This task includes a set of features one can use to identify these events and evaluates the power of each feature. Results then show LABurst outperforms a time series analysis baseline and is competitive with a domain-specific baseline despite LABurst lacking any domain knowledge. These outcomes are then connected back to crisis informatics by transferring LABurst's models learned in the sports domain to the task of identifying earthquakes in Japan.

A direct extension of LABurst, called RTTBurst, is then presented in Chap-

ter 5, which leverages LABurst's results and a subset of features to identify novel and topical events in real time. This chapter focuses on the implementation of an end-to-end system for real-time topic tracking using burst detection and distributed computing platforms. Evaluation results in a large-scale competition on real-time tracking and summarization in Twitter are then presented to demonstrate this approach's potential. Chapter 5 closes with an experiment integrating this real-time language- and domain-agnostic method with classical information retrieval systems, showing RTTBurst's burst detection generally increases performance.

Once these mechanisms are established for real-time event detection, research direction shifts in chapter 6 to establishing credibility in this data. To this end, Chapter 6 explores two data sets on credibility in Twitter, one that captures journalist-backed veracity and the other capturing crowd-sourced perceptions of credibility. Results show veracity is negatively correlated with the amount of disagreement in and length of a conversation, and perceptions of credibility are influenced by the amount of links to other pages, shared media about the event, and the number of verified users participating in the discussion. Differences in these two types of credibility prevent models trained on one data set from performing well when applied to the other. Despite such differences, one can build classifiers with these features to differentiate between credible and non-credible events for each data set.

Finally, chapter 7 concludes this dissertation with an overview of results, a discussion of their utility to the larger field, and early efforts and ideas for future work. Taken together, the research and results described in this dissertation provide a foundation for deeper exploration into social media's use as a reliable and

up-to-date source of information. With the beginnings of an implemented system already developed and tested, future work can expand on these results to develop support tools for journalists, first-responders, and consumers in areas under-served by traditional media platforms or in areas rife with political conflict and propaganda.

Chapter 2: Relevant Work on Social Media Analysis

Since Milgram's work on the value of weak social ties, researchers have increasingly studied social networks, their evolution, and what they can reveal about society and human behavior [15]. As electronic communities emerged with Usenet and online bulletin board systems in the late 1970s/early 1980s and chatrooms in the 1990s, scholars gained a new perspective from which to analyze these social networks directly. With the introduction of social media following the Internet's proliferation and early blogging platforms in the 2000s, the volume and velocity of social networks and media data grew rapidly. Today, social media platforms like Facebook and Twitter boast monthly user populations rivaling the size of countries, with Facebook connecting over 1.6 billion monthly active users alone in 2016 (larger than the population of China) [13].

Researchers have shown great interest in these massive and rapidly expanding data sets, with new research about insights garnered from such data published constantly. The democratization of information platforms like Facebook, Twitter, YouTube and others is supporting new and powerful uses across a range of topics. Research has shown correlations between sentiment on Twitter and fluctuations in major stock markets, leading the financial industry to leverage social networks

in buying and selling stocks and other assets [16]. In 2010, Sakaki et al. demonstrated Twitter could be used as an early warning system for earthquakes in Japan. Users shared news of earthquakes on Twitter almost instantaneously, and the radio waves containing this information propagated through wires faster than earthquakes propagated through the Earth's crust [17]. Other research has tracked evolutions in social movements and activism using social media [18–20], and more still have explored social media as a news source [21–24].

This dissertation builds on this foundation of social media analysis, with this chapter focusing on the most relevant research areas. First, this chapter discusses social media's use during and after disasters (both natural and man-made) in the recent field of crisis informatics. These disaster scenarios provide use cases for event and news detection in social media, leading this chapter into a discussion of event detection on the Internet and in social media.

Until the past few years, much of this research was retrospective (e.g., identifying newsworthy events over the past few days, weeks, or months) or dealt with small data sets, but as data generated by social media grew, near real-time event detection became more interesting and feasible. At the same time, the large size of social media data necessitated new methods to analyze at scale and perform these real-time event/news detection tasks, which this chapter covers as well. As motivated in Chapter 1, however, real-time event detection is only part of the problem: social media coverage of these events has little guarantee of veracity and is riddled with rumor. This chapter therefore closes with a review of recent research into issues of rumor detection and veracity in social media.

2.1 Social Media, Terrorism, and Crisis Informatics

A great deal of research has explored the role of social media, especially Twitter, in disaster response and critical information communication [25–28]. Much of the initial work on Twitter and crisis informatics focused on natural disasters, with examples like early work by Hughes and Palen on Twitter and emergency events [29]. This research demonstrated Twitter populations responded to these events by sharing more links to other web pages and fewer "person-specific reply tweets" [29]. Hughes and Palen acknowledged Twitter behaviors grow and evolve over time, and that these patterns may change. Despite these uncertainties, Hughes and Palen claimed emergency management organizations could leverage these resources for public communication, and government agencies have been performing their own investigations into this potential use [30].

Effective communication is important for managing any crisis response but is especially important during terrorist attacks, where the panic and fear that comes from poor communication is part of the attacker's objective. Analyses into Twitter use as it relates to terror attacks emerged as a research topic in 2011 when Oh, Agrawal, and Rao discussed ways in which terrorists in Mumbai in 2008 used information posted to Twitter to increase their effectiveness [31]. There results showed tweets contained informative situational awareness content often available on Twitter before major media networks. Similarly, Sullivan's 2014 paper also showed terrorists and sympathizers used social networks like Twitter to amplify their message and reshape the narrative surrounding the events [32].

These results suggest social networks like Twitter can disseminate information faster than governments or major news networks, making them valuable resources during crisis response [31]. At the same time, however, Twitter users are especially anxious and more prone to accept and transmit rumors during crises, and especially during about terrorist attacks [33,34]. While dangers do exist in unrestricted social media streams that cover terror attacks [31,32], works by Gupta and Kumaraguru, Faustino et al., Vieweg et al, and Olteanu et al. have all shown social media contains important and useful information about these events. This information can enhance public understanding and situational awareness of the event while also calming and supporting those affected [4, 12, 27]. Recent work by Olteanu et al. revealed the majority of content shared during 26 crises in 2012-2013 came from traditional media sources and contained (in order of prevalence) useful information, information about people affected by the crisis, and expressions of sympathy and emotional support [4]. While this work provided unique insights into the sources and types of content shared in response to these unexpected events, open questions remain concerning Twitter's structural and behavioral responses; e.g., it is unknown whether these events drive more users to social media and affect post or retweet frequency. Work in this dissertation builds upon these foundation by investigating these responses to a specific type of event (terror attacks) in a specific context (industrialized, Western countries). While others have explored social media and terror, this effort extends the state of the art by generalizing across a set of terror events.

2.2 Event Detection

Detecting events by leveraging digital media has fascinated researchers for over twenty years, with new methods, breakthroughs, and technologies emerging every few years. This subfield has evolved to integrate the latest available techniques and data sources, starting from early digital newsprint to blogs and now social media. Early stages of this research started in the mid-nineties with the Topic Detection and Tracking (TDT) initiative. These programs demonstrated feasibility in detecting new topics from traditional media sources, but as Allan, Papka, and Lavrenko discussed in 1998, these approaches required additional work to see real success [35]. Even in that early work, Allan et al. were already discussing the tradeoffs of using pre-defined keyword sets and event classes when detecting new events, an issue researchers are still addressing today. It is also important to note that, at this nascent stage, topic detection and event detection were relatively synonymous.

Though this early research focused primarily on topic detection from newsprint and traditional media sources, work by Kleinberg in 2002 altered the landscape by applying topic detection to non-traditional data sources like personal email archives and by introducing one of the first real treatments of burst analysis in "document streams that arrive continuously over time" rather than static collections [36]. Despite introducing the streaming context, Kleinberg cast topic detection as a retrospective, state-based optimization problem. Kleinberg then leveraged hidden Markov models to find sequences of high usage keywords, or bursts, from which he could detect events (and construct complex nested states to develop event hier-

archies). Kleinberg's examination laid the foundation for research into topic bursts that characterized much of the proceeding work in this area.

Following from Kleinberg's work and the increasing size of digital content on the Internet, several new approaches to topic detection emerged. Notably, topic detection divided into two distinct tasks: identifying topics in data via algorithms like Latent Dirichlet Allocation (LDA) [37] and detecting events from text. Research in this dissertation focuses on events rather than topics, so early event detection work like that from Fung et al. in 2005 is especially interesting [38]. Fung et al. extended Kleinberg's burst detection scheme by identifying bursty keywords from digital newspapers and clustering these keywords into groups to identify bursty events, which displayed success in identifying trending events in an English-language newspaper from Hong Kong.

Along with new interest in burst-centric research, scientists also began exploring additional data sources beyond traditional newsprint, especially blog content. Blogs offered direct insight into the social consciousness in a way previously unavailable via traditional media since blogs include a great deal of social information regarding the author. Zhao et al. took advantage of this additional social information in their 2007 work on flow-based event detection [39]. By integrating this social data with the textual and temporal techniques described above, Zhao et al. were able to identify events with high accuracy in two social data sets: the Enron email data set and the Dailykos dataset. Similarly, Bansal's group from the University of Toronto developed the Blogscope project to identify trending and bursty keywords by location as well as time across the entire "blogosphere" [40, 41].

Soon after, the research community began experimenting with alternative media sources like blogs, but real gains came when microblogging platforms began their rise in popularity. These microblogging platforms include Twitter and Sina Weibo and are characterized by constrained post sizes (e.g., Twitter constrains user posts to 140 characters) and broadcasting public information. A great deal of research explores how data posted to these networks can be leveraged to detect events of various kinds. For example, Becker, Naaman, and Gravano analyzed Twitter to separate tweets into those about "real-world events" versus non-event messages [42,43]. Diao et al. also developed a retrospective technique to separate tweets into global, event-related topics and personal topics [44]. This retrospective event detection research is valuable in understanding well-known events, but rapidly detecting events in streams of social media data is more difficult.

Many researchers have explored motivations for using platforms like Twitter and have shown interesting dynamics in user behavior around high-impact events. Lehmann et al.'s 2012 work on collective attention on Twitter explored hashtags and the different classes of activity around their use [45]. Their work included a class for activity surrounding unexpected, exogenous events, characterized by a burst in hashtag usage with little activity leading up to the event. Examples of work on burst detection includes several domain-specific research efforts that use sporting events for evaluation [46–48]. Lanagan and Smeaton's work relied almost solely on detecting bursts in Twitter's per-second message volume, which inspired a baseline method discussed in Chapter 4. Though naive, this frequency approach was able to detect large bursts on Twitter in high-impact events without complex linguist

analysis and performs well in streaming contexts as little information must be kept in memory. Detecting such bursts provide evidence of an event, but it is difficult to gain insight into that event without additional processing.

One of the most well-known efforts in detecting events from these microblog streams is the previously mentioned work by Sakaki, Okazaki, and Matsuo on detecting earthquakes in Japan using Twitter [17]. This work demonstrated one can detect earthquakes using Twitter simply by tracking frequencies of earthquake-related tokens. Surprisingly, this approach outperformed geological earthquake detection tools since digital data propagated faster through wires than tremor waves in the Earth's crust. Though this research was limited in that it required pre-specified tokens and was highly domain- and location-specific (Japan has a high density of Twitter users, so earthquake detection may perform less well in areas with fewer Twitter users), it demonstrated a significant use case and the potential of such applications.

Along with Sakaki et al., 2010 saw two other relevant papers: Lin et al.'s construction of PET, a probabilistic popular event tracker [49], and Petrović, Osborne, and Lavrenko's application of locality-sensitive hashing (LSH) for detecting first-story tweets from Twitter streams [50]. The PET model by Lin et al. circumvented the need for language model-based stop word lists by using probabilistic models to discriminate between common and informational tokens. Secondly, integrating social and structural features into the event detection task demonstrated significant performance enhancements could be gained through non-textual features. Thirdly, their paper built on Kleinberg's initial work by subsuming his state machine approach as a degenerate case of the PET model. Like the majority of its

contemporary systems, however, PET required seeding with a pre-specified list of tokens to guide its event detection.

Petrović and his colleagues' research on clustering in Twitter avoided the need for seed keywords by instead focusing on the practical considerations of clustering large streams of data quickly. That is, rather than construct a probabilistic mixture model for each token, Petrović focused on methods for clustering tweets that contained similar tokens into topical clusters. While typical clustering algorithms require distance calculations for all pairwise messages, LSH facilitated rapid clustering at the scale necessary to support event detection in Twitter streams by restricting the number of tweet comparisons to only those within some threshold of similarity. Once these clusters were generated, Petrović was able to track their growth over time to determine impact for a given event. This research was unique in that it was one of the early methods that did not require pre-specified seed tokens for detecting events and has been very influential in the field, resulting in a number of additional publications that demonstrate its utility in breaking news and high-impact crisis events [21, 51, 52]. An open issue in Petrović's work, however, is its reliance on semantic similarity between tweets, which limits its ability to operate in mixed-language environments.

In 2011, a new method for detecting events using wavelets that could identify events in Twitter streams without seed keywords was introduced by Weng and Lee [53]. After stringent filtering (removing stop words, common words, and non-English tokens), this method used wavelet analysis to isolate and identify bursts in token frequency along a sliding window. Significant bursts were then converted

into a cross-correlation matrix against which a graph partitioning algorithm was run to construct topical clusters. Besides the heavy filtering of the input data, this approach exhibited notable similarities with the language-agnostic method described in Chapter 4 with its reliance on bursts to detect event-related tokens. The methods described in Weng and Lee's paper, however, was more retrospective, focusing on daily news rather than breaking news detection on which the research herein focuses.

More recently, Xie et al.'s 2013 paper on TopicSketch performed real-time event detection from Twitter streams "without pre-defined topical keywords" by maintaining acceleration features across three levels of granularity: individual token, bigram, and total stream [54]. As with Petrović's use of LSH, Xie et al. leveraged "sketches" and dimensionality reduction to facilitate event detection and also relies on language-specific similarities. Furthermore, Xie et al. focused only on tweets from Singapore rather than the worldwide stream.

Despite this extensive body of research, it is worth asking how event detection on Twitter streams differs from Twitter's own offerings on "Trending Topics," which they make available to all their users. When a user visit's Twitter's website, she is immediately greeted with her personal feed as well as a listing of trending topics for her city, country, worldwide, or nearly any location she chooses. These topics offer insight into the current popular topics on Twitter, but the main differentiating factor is that these popular topics are not necessarily connected to specific events. Rather, popular memetic content like "#MyLovelifeInMoveTitles" often appear on the list of trending topics. Additionally, Twitter monetizes these trending topics as a form of advertising [55]. These trending topics also can be more high-level than

the interesting moments we seek to identify: for instance, during the World Cup, particular matches or the tournament in general were identified as trending topics by Twitter, but individual events like goals or penalty cards in those matches were not. It should be clear then that Twitter's trending topics serves a different purpose than the streaming event detection described herein.

2.3 Credibility and Social Media

Questions of credibility in the digital realm are not new phenomena. After computers' mystique and air of infallibility began to fall away in the mid-1990s, human-computer interaction (HCI) researchers sought terminology to describe and methods to improve credibility in computer systems. Fogg and Tseng's 1999 work in this area argued that, when people spoke of "credibility," they really referred to the perception of "believability," in which the information or output of a computer system is accepted as true or correct [56]. They went on to describe the nuanced differences between "credibility" and "trustworthiness," which they essentially reduced to whether one trusts the system itself ("trustworthiness") or the *information* produced by the system ("credibility"). While their work was more extensive in describing various aspects of computer credibility, the definition they presented is particularly germane to the research laid out in this dissertation because the information extraction and decision support tools herein hinge on the quality and believability of their output, and many researchers following this work use this definition of believability. Furthermore, Fogg and Tseng set the stage for the extensive body of literature in

this area with their finding that users find computers more credible when the user has a pressing or critical information need. This necessity is powerful when making split-second decisions informed by real-time information, so users need to be (but likely are not) particularly aware of possible issues in credibility when time is most critical.

In the year following Fogg and Tseng's deconstruction of credibility, Flanagin and Metzger published a piece on users' perceptions of credibility specific to the Internet [57]. As people increasingly turned to the Internet as an information resource despite the potential for exploitation, misinformation, and bias, Flanagin and Metzger surveyed a significant number of Internet users to explore whether they found web-based information credible and how credible it was in comparison to other media. At this early point in the Internet's life and while respondents' behaviors somewhat influenced their perceptions and habits, users found the information on the Internet mostly as credible as television, radio, and magazines but less credible than newspapers (a trend that would change over the next five years). As with Fogg and Tseng, Flanagin and Metzger also further motivated the need for mechanisms to stimulate, facilitate, or automate users' ability to verify credibility with their finding that "few [users] are rigorously verifying the information obtained via the Internet."

Paralleling the research into event detection, the next five years saw growing use of the Internet and the emergence of blogs and the Blogosphere in web users' habits. During this time, Consumer Reports carried out two studies on trust and credibility in the Web and found that, over time, news websites had become as

trusted and credible as their newspaper counterparts [58, 59]. Blogs and their early social networks on the other hand, while not even mentioned in the 2002 survey, were found almost wholly untrustworthy in 2005 even among the small percentage of respondents who said they had visited blogs.

This sentiment changed rapidly though, and by 2007, the work of Johnson, Kaye, et al. found politically interested web users had begun to seek out political blogs and judged them as moderately credible [60]. Interestingly though, these politically minded web users judged blogs "as more credible than any mainstream media or online source" in terms of depth of analysis but ranked blogs poorly with respect to fairness. Johnson et al. hypothesized these findings were partially the result of an information selection bias in that these politically minded users had actively sought out the blogs, which suggested user intentions had significant bearing on perceptions of credibility.

At about this time, social media platforms like MySpace, Facebook, Yahoo! Answers, and Twitter (which opened in 2007) were rising in prominence (except for perhaps MySpace, which was being supplanted). Soon thereafter, Agichtein et al. began investigating automated methods for identifying high-quality content in such platforms, specifically Yahoo! Answers. Along the same vein as Fisher et al. [61], McCallum et al. [62], and Welser et al. [63], this work sought to integrate not only content-based features but also "non-content information" like structural relationships between answerers, answerer roles, click count, dwell time, and patterns of behavior of highly voted answerers. Agichtein and his colleagues then demonstrated a classification mechanism built with these textual and non-textual features could

discriminate between high- and low-quality Yahoo! Answers content with high accuracy. They theorized that such an approach could be generalized to other social networking platforms that exposed similar structural, non-textual features.

Again following event detection vogue, researchers' fascination with credibility in social networks really became a fascination with Twitter and other social networks, and from 2010 onward, numerous studies were published on credibility in Twitter each year. 2010 in particular saw a pair of studies on Twitter's dark side in political campaigns. Mustafaraj and Metaxas investigated Twitter usage during several elections and how spammers and malicious entities used the platform to attack political candidates [64]. Their exploration focused specifically on how search engines were providing "real-time search results" backed by Twitter with little regard for veracity, which resulted in spammers' first instance of a "Twitterbomb" on Google's search pages. By creating numerous fake accounts with compelling posts, all of which led back to a single web page, spammers were able to entice actual users to retweet the spam content and reach a much broader audience. Mustafaraj and Metaxas's results also revealed the interaction networks between the accounts they flagged as possible spam sources differed significantly from interaction networks between actual users.

The second such study saw the creation of the Truthy system by Ratkiewicz et al. [65]. Truthy's purpose was to detect memetic attacks on political candidates, specifically attacks that used "astroturf" memes to spread misinformation and create a false sense of grassroots support for this misinformation. To support Truthy, Ratkiewicz et al. built the Klatsch framework to process daily batches of Twitter

data, detect memes in that data, and assign a "truthiness score" to each meme. Truthy detected these memes by first filtering based on a curated set of politically relevant keywords and popularity (a threshold of mentions per hour) then analyzed each meme's diffusion characteristics and sentiment. As part of this diffusion analysis, Truthy identified the most prolific broadcasters of a particular meme. Then, using supervised learning, Ratkiewicz et al. developed a "truthiness" classifier to discriminate between "truthy" and "legitimate" memes with an accuracy exceeding 90%. Therefore, Ratkiewicz et al. showed that not only was Twitter a vehicle for both legitimate and malicious information, they also demonstrated feasibility in discovering and removing misinformation.

Perhaps 2010's most compelling exploration of credibility in social media was again from Yahoo! Research with Mendoza et al.'s study of Twitter use during the 2010 Chile earthquake [66]. During this natural disaster, Twitter played a significant role in communication and coordination, but it was also a source of constant misinformation, which added to feelings of chaos. The authors investigated how high- and low-quality information diffused through the social network by comparing verifiable news and events with the spread of rumors, which led to a significant finding: true tweets tended to have extremely low rates of contradiction (that is, for every 100 tweets about a piece of true information, one might see only one tweet that contradicted that information), and rumors exhibited much higher rates of questioning and contradiction (nearly a 1:1 ratio). This result suggested a potentially powerful feature in identifying misinformation in Twitter.

Mendoza et al. followed up this work in 2011 with automated methods for

applying their lessons learned during the 2010 Chile earthquake [8]. This study leveraged information propagation (via retweet analysis), user certainty, use of external sources (i.e., web links), and user characteristics like follower count as features for a supervised machine learning system capable of classifying high- and low-quality information. The authors relied on Amazon's Mechanical Turk to generate a labeled set of tweets with varying degrees of plausibility/credibility and used decision trees to learn this classification task with 86% accuracy. From these features and their classifiers, they concluded the most credible data on Twitter would generally start with one to a few users and exhibit deep retweet networks. This result, however, is possibly contradicted by more recent events like the various instances of compromise of otherwise credible accounts (e.g., the AP and White House attack).

Similarly, Gupta and Kumaraguru's analysis of Twitter usage during the Mumbai terrorist attacks in 2011 showed tweets by authoritative users (users with many followers) were exceedingly rare [12]. Since a significant amount of useful situational information is communicated via Twitter in the moments after a crisis [1], one must rely on information posted by potentially inaccurate users during these times. Fortunately, Gupta and Kumaraguru returned in 2012 with methods for ranking tweets by credibility during such high-impact events [9].

At the same time, Qazvinian et al. created a data set of tweets and Twitter users replete with labels of which tweets were rumors versus non-rumors and which users were rumor believers/propagators versus rumor disbelievers [67]. From this data set, the authors built a framework capable of classifying tweets and users accordingly. Their classifiers leveraged features of tweet content and constituent

parts of speech, network structure with respect to original tweets versus retweets, and embedded entities like hashtags and external references. Qazvinian et al. then demonstrated feasibility in using Bayesian and log-linear models built around these features to discriminate between rumors and non-rumors successfully. This work is particularly interesting in the context of the research presented in this dissertation because it focuses more on the quality of informational tweets rather than particular users, a focus also present in Chapter 6.

2012 saw a slight shift in how credibility analysis was being applied to social networks like Twitter. Rather than focusing on the informational content and its credibility, researchers began using social media to identify potentially valuable or credible human *sources*. First, Kang et al. published a paper on modeling topic-specific credibility, but instead of considering the credibility of a particular topic (as done previously regarding astroturf rumors), they focused on finding social media users whose opinions or posts about the topics would be credible [7]. Diakopoulos et al. followed a similar path in their piece of assessing social media sources for journalism [11]. Again, rather than analyzing topical credibility, Diakopoulos et al. designed a journalism support tool to find credible eye witnesses to an event of interest based on several cues, such as location, external references, mentions of entities, and whether the user was using a mobile platform. While this research potentially conflated trust and credibility (as cautioned by Fogg and Tseng 15 years ago), cues like whether a post is from a mobile device or not are compelling in that such information is easily available and potentially informative in a real-time setting.

This shift towards finding credible users is not the only change in credible social media research. Several approaches have sought to be more proactive in supporting human credibility assessment. In 2011, Schwarz and Morris explored methods for augmenting search results with visualizations to support users in assessing credibility of sources on the front end [68]. Morris returned in 2012 with a Twitter-specific version of this work in which Twitter search results were augmented to include otherwise obscured credibility cues like follower counts, user locations, verified accounts, perceived expertise, and consistency of topics posted by the author [69]. The value here is that these studies show feasibility in augmenting systems to support enhanced credibility, which establishes part of this dissertation's foundation.

It is worth noting here that, in addition to work on credibility in social media, an extensive corpus of research covers issues of trust in users. A number of algorithms attempt to gauge user trustworthiness, potential trust between users, or recommend items based on trust between users [70–72]. While these resources are certainly important, the research presented in this dissertation focuses more on aggregate analysis of credible information from many users rather than evaluating trust between two specific users.

Chapter 3: Twitter Response to Terrorism

Social media has become an important tool during crises, and public interest in using these social platforms during crises is well-documented in recent surveys [26]. This interest has translated into an expectation that government organizations provide updated information about emergencies through social network channels [73], with recent federal efforts seeking to improve communication during emergency situations [30]. Companies are also responding to these needs with offerings like Twitter Alerts[1] and Facebook's SafetyCheck[2].

Though emergency situations manifest in many forms and timeframes, terrorist attacks are of particular importance given their destabilizing and panic-inducing effects. With many people turning to social media for coverage and information about terrorist attacks, understanding user response to these events on these platforms could yield valuable insights. Such insights could assist event detection and computational journalism algorithms to discover terrorist attacks as they occur or improve recommendation systems that highlight information from local sources (local law enforcement or news organizations).

Despite these potential benefits, the (fortunate) paucity of terrorist acts in

[1] https://about.twitter.com/products/alerts

[2] https://www.facebook.com/about/safetycheck

developed countries with large Twitter populations complicates this research. Some work has broached this topic, but much of it focuses on single terrorist acts with limited generalization [12, 26, 52, 74]. Olteanu, Vieweg, and Castillo's work on social media response to crises enhances this generalizability in the broader context of all disasters, but has limited coverage of terrorist attacks [4]. Their work shed light on the types of content users share during these crises but omits analysis of behavioral responses on social media (e.g., information sharing or seeking behaviors like sharing media or other user's content). Behavioral signals like increased message or user volume, hashtag use, or sharing other users' posts (called "retweeting" in Twitter) could signal the onset of high-impact events like terrorist attacks or other crises.

To elucidate these responses, this chapter presents an investigation into social media use in developed, Western countries during five terror events: the 2013 Boston Marathon bombing, the 2014 Sydney hostage crisis, the 2015 Paris Charlie Hebdo shootings, the 2015 Paris November attacks, and the 2016 Brussels bombings. This work analyzes information sharing behaviors (retweeting, hashtag usage, mentions, etc.), information seeking behaviors through changes in follower counts, and social interactions before and after these events. This exploration characterizes public response to these events and identifies important accounts during these crises (findings show a preference for local police, if present on Twitter, and local news affiliates).

3.1 Events and Data Collection

As mentioned, this work characterizes public response on Twitter across five terror-related events. To test significance of behavioral changes in response to these events, the analysis covers two weeks before and after each event, based on the following dates (in Coordinated Universal Time, or UTC):

- The Boston Marathon bombing and resulting manhunt on 15-19 April 2013,
- The 2014 Sydney hostage crisis on 14-15 December 2014,
- The Charlie Hebdo attack and manhunt between 7-9 January 2015,
- The 2015 Paris attacks on 13-14 November 2015, and
- The 2016 Brussels bombings on 22 March 2016.

This work leveraged a corpus of tweets gathered from Twitter's 1% public sample stream between April 2013 and April 2016, with an average of approximately four million tweets per day. This corpus was created using the twitter-tools library[3] developed for evaluations at the National Institute for Standards and Technology's (NIST's) TExt Retrieval Conferences (TRECs). Others have explored bias in this sample and have shown that, while local events and long-tail coverage are lost in this sample, high-impact events, trending topics, and network structure for many accounts are conserved [75, 76]. Given the highly impactful nature of these events, this 1% sample should be adequate for judging Twitter's immediate responses.

[3] https://github.com/lintool/twitter-tools

Biases aside, Twitter data is also known to contain a non-trivial amount of spam and noise [77, 78], motivating the need for a method to remove irrelevant content. To that end, this analysis compares unfiltered and relevancy-filtered data from each four-week period. Similar to the work by Olteanu et al., relevant Twitter data is identified using a simple keyword search method [4]. These keywords were "boston" for the Boston Marathon bombing, "sydney" for the Sydney Hostage Crisis, either "paris" or "hebdo" for the Charlie Hebdo Attack, "paris" for the November attacks, and "brussels", "bruxelles", "brussel", "zaventem" for the Brussels bombings.. This search is case-insensitive and matches keywords embedded in hashtags. Furthermore, neither retweets nor short tweets are removed since retweets are one of the behaviors of interest. Furthermore, retweets affect structure in the interaction graph and supports identifying central actors during these crises. Table 3.1 reports dates, keywords, and tweet counts for each event.

3.2 Twitter Behavioral Analysis

While Twitter restricts messages to 140 characters, users have a wide variety of content they can share in this small space: links to other websites, hashtags, multimedia, mentions of other users, and retweets. Trends in these artifacts evolve over time, exhibiting patterns in information sharing. To this end, these artifacts are analyzed for significant shifts in usage around target events, as determined through two tests: 1) calculating whether an activity's frequency changes by more than three times the median absolute deviation (MAD) in response to the event, and 2)

Table 3.1: Twitter Data During Terrorist Attacks

Event	**Date**	**Date Range**	**Keywords**	**Tweet Count**	**Relevant Count**
Boston Marathon Bombing	15 Apr. 2013	1 Apr. 2013 to 30 Apr. 2013	boston	134,287,450	316,993
Sydney Hostage Crisis	15 Dec. 2014	1 Dec. 2014 to 31 Dec. 2014	sydney	134,288,848	50,842
Charlie Hebdo Attack	7 Jan. 2015	24 Dec. 2014 to 23 Jan. 2015	paris, hebdo	137,226,841	305,177
Paris Nov. Attacks	13 Nov. 2015	1 Nov. 2015 to 31 Nov. 2015	paris	114,210,893	610,948
Brussels Bombings	22 Mar. 2016	8 Mar. 2016 to 5 Apr. 2016	brussel, bruxelles, zaventem	101,250,958	110,704

performing a Welch's t-test on whether data before the event differs significantly from after (all tests are two-tailed, assume different variances, and performed at $p < 0.05$). The following questions explore these activities:

- **RQ1** Is overall Twitter activity affected by the event?
- **RQ2** Does relevant tweet volume change during the event?
- **RQ3** Does the proportion of retweets, links, hashtags, mentions, or media change during the target event?

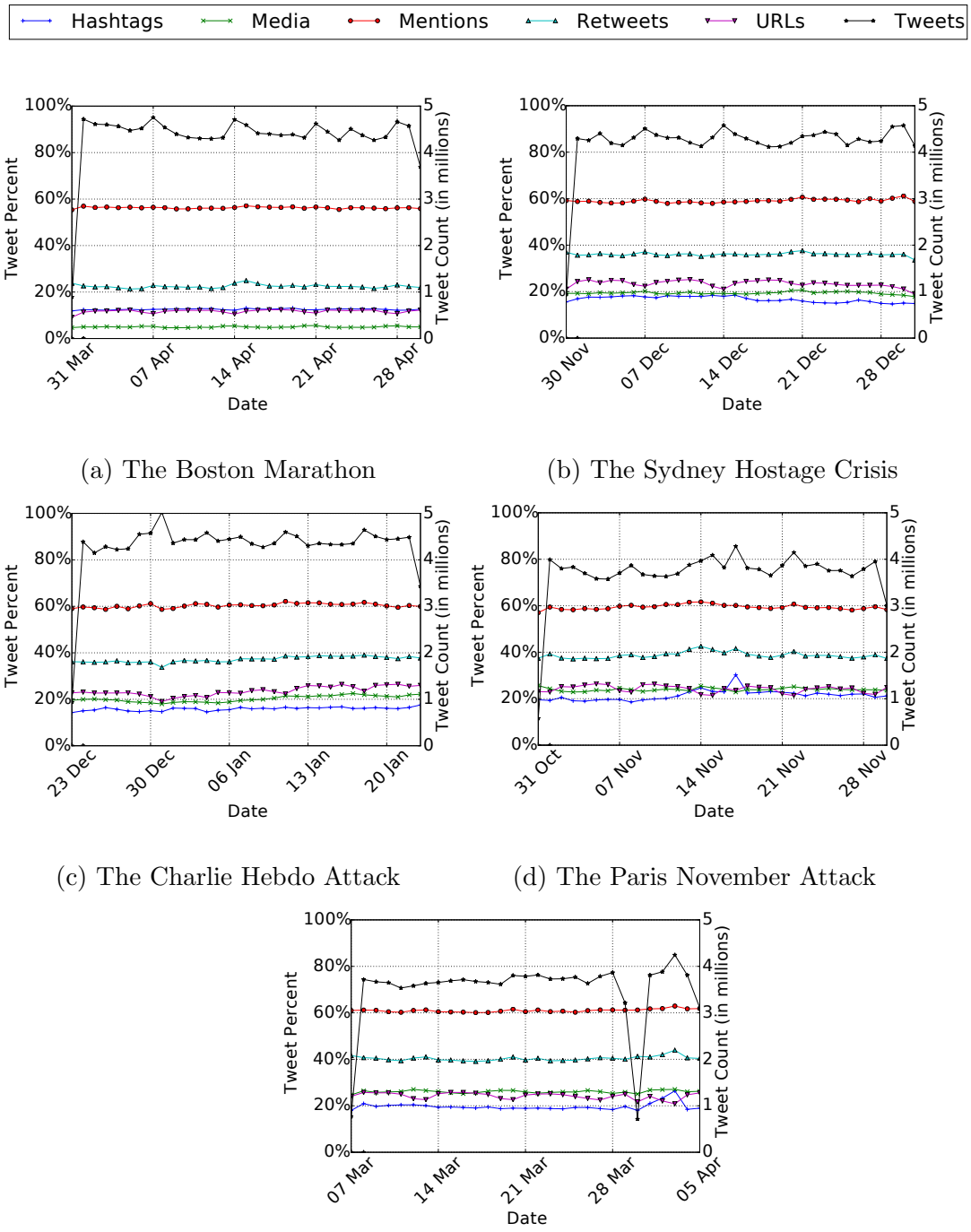
- **RQ4** Do follower counts change for important accounts during the event?
- **RQ5** How long do these changes persist?
- **RQ6** What users emerge as important during terror attacks?

RQ1 and **RQ2** are motivated by research that shows people's intent to seek/share information increases in response to crises and asks whether this effect holds for Twitter [79]. **RQ3** then identifies which specific artifacts are most affected by these events; e.g., significant increases in retweets might indicate higher information sharing as Twitter users rebroadcast information to their followers or join the conversation. **RQ4** then measures information seeking behavior by quantifying users' subscriptions to and thus seeking from important Twitter accounts. **RQ5** characterizes duration of these effects, about which existing research conflicts. Olteanu et al. [4] showed changes on Twitter can persist for a few days to nearly two months, with the Boston Marathon bombing persisting for 60 days, but Koutra, Bennett, and Horvitz demonstrated shocking events rarely influenced long-term user behavior in digital communities beyond social networks [80].

During such crises, it is also unclear which accounts contribute the most to discussion and coverage. This question is investigated in **RQ6** by converting Twitter's retweet and mention activity into a directed graph of interactions, where the vertices represent Twitter users, and the edges denote mentions/retweets. Research shows users with many followers or retweets often are not the most influential users [81]. Instead, this work follows Kwak et al. and uses a version of the PageRank algorithm to identify important accounts in these networks [82].

3.3 Results

A first step in understanding public Twitter response is to examine general behavior through 1% sample stream and its artifacts. Analysis of this data in Figures 3.1a-3.1e shows general Twitter activity is unaffected by these events (the black-asterisk curve denotes tweet volume on the right vertical axis and remaining curves show tweet proportions on the left axis). Statistical analysis shows few significant changes in activity: increases in retweets and hashtags on the day of the Boston Marathon bombing; increases in retweets, mentions, and hashtags during the Paris November attacks; and an increase in retweets during the Brussels bombings. The data set for the Brussels bombings, however, has a discontinuity on 31 March where data collection failed, resulting in a large drop in tweets collected on that day. In general, overall tweet and user volume and other activities are unaffected by these events, thereby addressing **RQ1**.



(a) The Boston Marathon

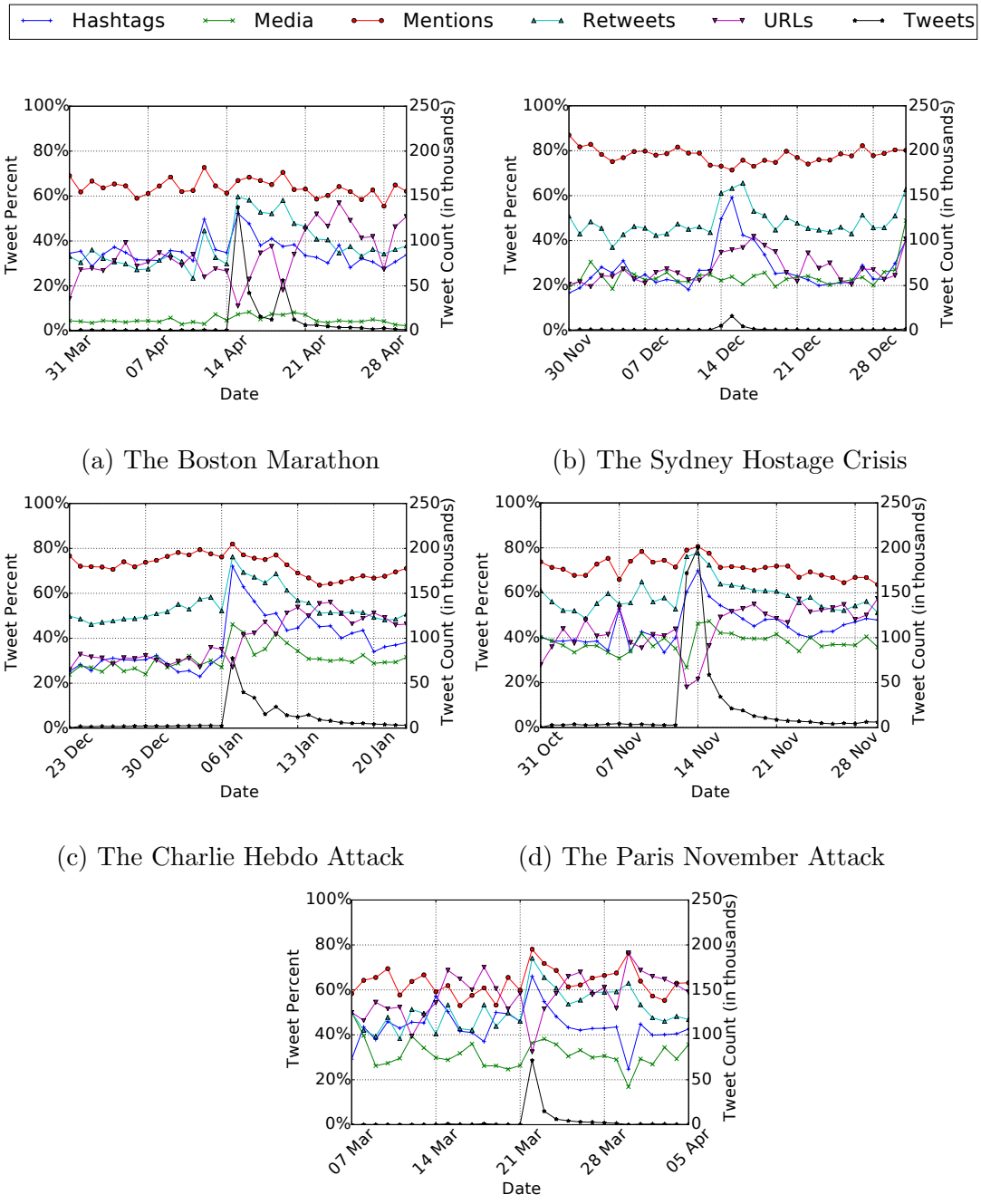
(b) The Sydney Hostage Crisis

(c) The Charlie Hebdo Attack

(d) The Paris November Attack

(e) The Brussels Bombings

Figure 3.1: General Twitter Activity



(a) The Boston Marathon

(b) The Sydney Hostage Crisis

(c) The Charlie Hebdo Attack

(d) The Paris November Attack

(e) The Brussels Bombings

Figure 3.2: Relevant Twitter Activity

For **RQ2** and **RQ3** (relevant tweet volume and changes in activities), Figures 3.2a-3.2e show the black-asterisk curve of tweet volume on the right axis with remaining curves corresponding to tweet proportions on the left. Tweet volume increased on the date of each terror event, with references to Boston increasing from about 828 tweets per day to 138,000 on the day of the bombing. Sydney saw a smaller increase from an average 1,119 tweets per day to 16,000, the Charlie Hebdo attack increased from 2,288 to nearly 78,000, the Paris November attacks increased from near 3,000 to 172,000, and Brussels went from 320 to 72,000. Each event's first day was between 42 and 823 times the MAD. In all cases except Brussels, relevant tweets returned to pre-event levels within four days of the event ($p < 0.05$). In answer to **RQ2**, Twitter saw a significant increase in references to the target events on the day the terror attack starts, but these references quickly decreased in frequency.

For **RQ3**, in Boston, proportions of retweets, web links, media, and hashtags exceeded three times the MAD for 15 April, with retweets, media, and hashtags seeing increases while links decrease. Proportion of tweets containing mentions were not significantly affected. Of these activities, a significant change occurred in retweets and link sharing before and after the event ($p < 0.05$). Sydney saw a similar deviation from the MAD on 15 December in retweets, links, and tweets containing hashtags, with mentions being the only activity that decreased. The Welch's t-test also shows links continued to deviate from pre-event levels for the remainder of the time period, while all other activities returned to pre-event levels within four days. During the Charlie Hebdo attack on 7 January, retweets, media sharing, and hashtags also significantly deviated from the MAD. Links once again see a drop

in usage, but the change is not significant on the first day of the attack. Average trends in all sharing activities except retweets differed significantly before and after the event ($p < 0.01$ for mentions, links, media, and hashtags). Furthermore, average daily mentions decreased following the event, whereas all other activities saw an increase. In the Paris November attacks, retweets, hashtags, and links were the only activities to deviate from the MAD. Finally, for Brussels, retweets, mentions, and hashtags deviate from the MAD. Retweets and hashtags increased immediately following these events, while link sharing responded more slowly. Tweets including media and mentions are inconsistently affected.

To answer **RQ4**, the Boston Police Department (@bostonpolice), the New South Wales Police Department (@nswpolice), Charlie Hebdo (@Charlie_Hebdo_-), the Parisian "Recherche Personne" account (@SOSParis1311), and the Brussels Airport (@BrusselsAirport) were examined. The Boston Police Department and Chalie Hebdo saw massive increases in followers. Prior to the event, the Boston Police Department had an average of 54k followers, which increased significantly to 264k followers on the day of the manhunt and peaked at over 300k through the end of the month. Similarly, followers of Charlie Hebdo increased from an average of 77k to 318k. The NSW Police Department also sees a significant increase, though not as substantial, from 61k to 80k followers. The "Recherche Personne" account was unique in that it was created in response to the event, quickly amassed 5k followers on the day of the event, and saw a maximum of nearly 14k followers over the next four days. The Brussels airport account saw an increase similar to the NSW Police Department, going from 27k followers to 34k on the day of the event

with a maximum near 50k over the next two weeks.

Increases in the first three accounts' followers were also tested for significance, and each differed significantly from the average: Users referring to Boston, Sydney, and Paris saw a mean increase of 78, 136, and 253 followers ($\sigma = 2,753$, 989, and 2,731) respectively. Therefore, all three accounts experienced significant increases in followers ($p < 0.01$), with rapid increases that level off within 7 days.

For effect duration (**RQ5**), retweets and hashtags exhibited significant but short-lived surges. Retweets referencing Boston retreated to 36% from a high of 59% on 15 April, and hashtags dropped from 52% to 31%; hashtag usage before the bombing and seven days after was not significantly different ($p > 0.05$). Similarly for the Sydney Hostage Crisis, retweets and hashtags returned to pre-event levels with no statistical difference ($p > 0.01$) within seven days. During the Charlie Hebdo attack, retweets were similar, but hashtags remained significantly higher than their pre-event levels for at least the next two weeks ($p < 0.01$). During the Paris November attacks, Welch's t-test showed mentions, links, and hashtags diverged from pre-event levels for the remainder of the time period ($p < 0.01$). Brussels likewise saw links diverge from pre-event levels over the next two weeks along with retweets ($p < 0.01$) (though these effects may result from the data collection failure mentioned above). For links, after all events except the Sydney Hostage Crisis, link sharing increased and remained high, and while followers showed a slight decrease after the event, they remained significantly higher than pre-event levels. In answer to **RQ5**, changes in retweets and hashtags persisted for only a few days, whereas link sharing and follower counts saw sustained increases for at least two weeks after

the events.

Table 3.2 answers **RQ6** and depicts the ten most central accounts during each event. The Boston Police Department was the most central account during the Boston Marathon bombing, closely followed by the Boston Globe. With the exception of @JFKLibrary and @bostonmarathon, all other accounts belong to news affiliates, three of which are local to the city of Boston (@BostonGlobe, @BostonDotCom, and @7News). Central accounts in Sydney also included government/law enforcement agencies (@nswpolice, @TonyAbbottMHR), news organizations (@abcnews, @BBCBreaking), and several unaffiliated accounts. Response to the Charlie Hebdo Attack was similar in that many central accounts belong to news organizations but differed noticeably with the absence of law enforcement. For the Paris November attacks, the most central account, @SOSParis1311, was an account created in response to the event to help people find friends and loved ones with several others being local to Paris (with the notable exception of US presidential candidate Donald Trump's account). The Brussels bombings continued these trends with the most central accounts belonging first to a Belgian newspaper, Le Soir, the Brussels airport account, and a European Union crisis center in Belgium, @CrisiscenterBE, followed by major media organizations.

3.4 Observations

An important observation is the limited response in Twitter's sample stream; none of these events altered much of the overall Twitter activity. Given that the US and

Table 3.2: Central Accounts (Highest Rank to Least)

Rank	Boston	Sydney	Charlie Hebdo	Paris Nov.	Brussels
1	BostonPolice	abcnews	itele	SOSParis1311	lesoir
2	BostonGlobe	TonyAbbottMHR	Charlie_Hebdo_	Rech_Paris	lesoirplus
3	JFKLibrary	nswpolice	AFP	ParisFilmes	BrusselsAirport
4	AP	9NewsSyd	AFPphoto	AFP	realDonaldTrump
5	BostonDotCom	CottomSydney	Le_Figaro	le_Parisien	CrisiscenterBE
6	7News	sydneyharbert	jmdecugis	AureliaBAILLY	AP
7	bostonmarathon	abit_wp	le_Parisien	alafolieparis19	MailOnline
8	ReutersUS	WolfSpirit2013	plantu	AP	SkyNews
9	YourAnonNews	9NewsAUS	BFMTV	ParisVictims	Conflicts
10	Reuters	BBCBreaking	ctxt_es	YouTube	FoxNews

France account for more than 25% of Twitter's user base [83], one would expect such national events to have a stronger effect. Gupta and Kumaraguru also suggest that sharing links increase during terrorist events [12], but this response is only present when constraining the analysis to relevant tweets. Despite absent overall response, Figures 3.2a-3.2e show a portion of Twitter does respond to these events, with relevant tweets accounting for $1 - 4\%$ of all tweet activity on those days. Taken together, surges in these activities suggest higher information sharing during these times even if these surges drop off quickly, which demonstrates Twitter's short memory as relevant tweet volume dropped by 80% within three days. Follower counts are an exception as they remained high for at least two weeks after the events.

Central accounts also show an interesting result: In all five cases, several news

affiliates emerged as leading sources of information. This result is unsurprising given media organizations report breaking news, but the presence of *local* organizations suggests users value information sourced close to the event. This result also corroborates Sutton et al. [74], who showed local actors emerged as highly influential in Boston. Information from authoritative sources like local law enforcement have more influence on the public than random users and major media organizations [26], and the popularity of local crisis response organizations is consistent here.

3.5 Consequences for Event Detection Algorithms

This chapter explores public Twitter response by characterizing trends in activities and accounts mentioned during terrorist attacks. Results show relevant tweets, retweets, and hashtags increase significantly and immediately and return to pre-event levels within days. Longer-lived responses appear in link sharing and followers for central accounts. At the same time, the public coalesces around police/government (where available) and local news organizations when sharing information. Results show a surge of interest in local crisis responder agencies during the majority of events, but these results show little government presence during the attacks in Paris, with the majority of response being from local response or media organizations.

For further work in event detection, these results suggest one cannot rely on simple frequency-based analysis methods to identify high-impact events. That is, these events did not drive increases in overall Twitter message or user volume.

Instead, results revealed bursts in topically relevant activity and relevant account followership, suggesting these events are identifiable with mechanisms that detect changes at the topic and account levels. Known, relevant keywords like "paris" or "boston" experience bursts in usage, as demonstrated above, lending credence to these methods. Future work therefore should explore methods for identifying these relevant, bursty keywords, methods for which are discussed in the following chapters.

3.5.1 Other Future Work

Related to the aforementioned results on topically relevant bursts, two additional avenues of future work could follow on from this work. First, the similarities across user responses to these events in Twitter suggest one may be able to construct a vocabulary of crisis/terrorist attack response. A data-driven approach to understanding these responses could inform future crisis response research and allow analysts to identify potentially local terror-related events that do not achieve the same level of notoriety as the events described herein.

Furthermore, given that these crisis-level events effect changes in Twitter's topical distribution without increasing the volume of tweets, an interesting corollary is that other topics of discussion must receive less attention in response to these events. Therefore, and interesting open question is whether patterns exist in what topics become less interesting to users when they begin discussing crisis-related topics.

3.5.2 Limitations

This work has several limitations, each of which can be grouped into one of two categories: Twitter-specific limitations, and selection limitations.

The main limitation specific to Twitter is that relevant tweets are likely under-sampled given the relatively simplistic method employed for identifying relevant tweets. An example of this issue can be seen with relevant content like the Boston Police Department's tweet after capturing the second suspect in the Boston Marathon bombings: "CAPTURED!!! The hunt is over. The search is done. The terror is over. And justice has won. Suspect in custody." This tweet is clearly relevant to the event but does not include specific references to Boston or the bombing. Rather than searching for tweets that contain specific substrings from a small set of keywords, future work could leverage more sophisticated information retrieval methods like query expansion or identify seed users to capture additional relevant data.

Another limitation in the Twitter data used here is that it provides limited insight into information seeking or data consumption behaviors. Follower counts yield some insight into users' reading habits, but Twitter provides both a logged-out experience (i.e., a user can consume Twitter content without logging into the platform), and the data set used here has no information on tweet views. As a result, the analysis presented herein is unable to track or model passive interactions or which tweets are the most viewed. Partnerships with Twitter or leveraging additional analytics platforms may be able to address these weaknesses.

Other limitations in this work focus more on issues of selecting platforms and data. The most glaring weakness here is the omission of other platforms. Typical social networking sites like Facebook and reddit and atypical platforms like Snapchat or regionally popular networks like WeChat and Weibo also contain valuable responses. Platforms will likely exhibit unique responses in coverage beyond structural differences (e.g., Facebook users share content via "shares" rather than "retweets"), and exploring these responses might yield additional insight into population-level and cultural response to these events. Furthermore, aligning or triangulating data across these platforms could yield a more holistic picture of events (e.g., Instagram is focused on sharing images, which might give a different view into responses).

Another potential limitation in this work comes from the focus on terrorist attacks. While they are impactful at a national level, it is possible that the global audience may not be as interested in these events as they are in truly international events like the World Cup or Olympics. Such international events are potentially more interesting to a wider audience, suggesting they are in some sense more newsworthy or interesting. The possibly limited international newsworthiness of these terrorist attacks may then be the driving force behind why they have little to no effect on general, global Twitter activity.

Finally, this work is specific to developed, Western countries, but these results may not hold for terror attacks in less developed countries or where terrorism occurs more often. Future efforts should broaden topical and geographic scope and explore other major social media platforms.

Chapter 4: Language-Agnostic Event Discovery in Streams

Social media's utility during crises along with the prevalence of media organizations on the platforms has driven research around the role of social media in news consumption and production. Works by Petrovic et al., Kwak et al., and Vis have all explored social media as a platform for disseminating and consuming news and comparing Twitter to newswire sources [21, 25, 82]. At the same time, works by Diakopoulos et al. have leveraged social media as a journalistic tool to evaluate debate performance and identify expert and eye witness sources [11, 84]. These areas of research have laid the foundation for the new field of computational journalism.

One of social media's major advantages for computational journalism is its rapid nature: near real-time feedback and information are available about nearly any event. To track this data, however, one must be able to identify these events at the same velocity. Many current approaches to event detection rely on prior knowledge and keyword engineering to detect events of interest or are delayed while relevant keywords are determined. While straightforward and capable, such approaches are often constrained to events one can easily anticipate or describe in general terms, potentially missing impactful but *unexpected* key moments. For instance, one can follow the frequency of "goal" on Twitter during the 2014 World Cup to detect when

goals are scored [14], but interesting occurrences like penalties or missed goals would be excluded. One might respond to this weakness by tracking additional penalty-related tokens, but one cannot continually enlarge the keyword set for all cases. Furthermore, one would still be unable to identify an unexpected moment like Luis Suarez's biting Giorgio Chiellini during the Uruguay-Italy World Cup match; who would have thought to include "bite" as a relevant token? Relying on predefined keywords also restricts these systems to languages represented by the seed keywords, a significant issue for international events like the World Cup.

Given social media's high volume, one could forgo seed keywords and leverage time series analysis to track bursts in message frequency (as with Vasudevan et al. [46]). Such methods gain flexibility by sacrificing semantic information about detected events (one would need to extract keywords causing such bursts manually) but rely on the signal to be present in the data. In social media's case, this reliance implies that either the volume of messages or the volume of users on a social media platform changes in response to the target events. As shown in the previous chapter, however, major events like terrorist attacks have limited effect on overall Twitter message or user volume.

This chapter instead introduces LABurst, a language-agnostic burst detection algorithm that uses machine learning and distributed high-performance processing to identify topical or token-specific bursts in Twitter without requiring pre-specified domain keywords. LABurst combines the topical and frequency analysis approaches to identify bursts at the individual token level. The number of tokens experiencing a burst at any given time is then an indicator of a high-impact or key moment; that

is, as more tokens experience a simultaneous burst, the higher the impact of that moment. Intuition behind this approach is that, when a major event occurs, many users post messages about the event relatively simultaneously, but these users may use different keywords to describe the same event. Therefore, when many keywords experience a burst, or are "bursty," at approximately the same time, they are likely to be related, and the larger the set of bursty keywords, the higher impact the event. Examples of this intuition include many users posting about a goal in the World Cup using various forms of the word "goal" (e.g., "goal," "gol," "gooooal," etc.) or about a bombing with related terms like "bomb" or "explosion."

Contrasting with existing work, LABurst operates in the streaming context. This flexibility is illustrated with experiments on Twitter's sample stream surrounding key moments in large sporting competitions and two natural disasters. The motivation for using sporting events here is two-fold: first, large-audience sporting events are more common and regularly occurring than natural disasters, and major sporting events have well-defined events in terms of goals, fouls, and similar moments of play.

LABurst is evaluated by comparing it to two baselines: a time series-based burst detection technique, and a domain-specific technique with a pre-determined set of relevant keywords. Results from these experiments demonstrate LABurst's competitiveness with existing methods. This work makes the following contributions:

- Presents a streaming algorithm and feature set for the discovery and descrip-

tion of impactful and unexpected key moments in Twitter's public sample stream without requiring manually defined keywords,

- Demonstrates LABurst's performance is both competitive and flexible, and
- Transfers sports-trained models to the higher-impact domain of disaster response.

4.1 Moment Discovery Defined

This chapter demonstrates LABurst's ability to discover and describe impactful moments from social media streams without prior knowledge of event types or domains. To that end, one must first lay LABurst's foundations by defining the problem LABurst seeks to solve and presenting the model around which LABurst is built.

4.1.1 Problem Definition

Given an unfiltered (though potentially down-sampled) stream S of messages m consisting of various tokens w (where a "token" is defined as a space-delimited string)[1], the objective is to determine whether each time slice t contains an impactful moment and, if so, extract tokens that describe the moment. Identifying and describing such moments separately is difficult because, by the time one can react to a key moment with a separate analysis, the moment may have passed. A "key moment" is defined as a brief instant in time, lasting on the order of seconds, that a journalist would

[1] "Token" is more general than a "keyword" as it includes numbers, emoticons, hashtags, or web links

label as "breaking news." Key moments might comprise the highlights of a sporting competition, the moment an earthquake strikes, the moment a terrorist attack occurs, or similar. Such moments often generate significant popular interest, affect large populations, or represent an otherwise instrumental moment in larger event (e.g., the World Cup). By focusing on these instantaneous moments of activity, the complexities of defining "events" and the hierarchies among them are avoided.

Formally, E denotes the set of all time slices t in which a key moment occurs. The indicator function $\mathbb{1}_E(S_t, t)$ takes the stream S up to time t and returns a 1 for all times t in which an impactful moment occurs, and 0 for all other values of t. The moment discovery task is then to approximate this indicator function. The function $B_E(S_t, t)$ is also included and returns a set of words w that describe the discovered moment at time t if $t \in E$ and an empty set otherwise. To account for possible lag in reporting the event, typing out a message about the event, and the message actually posting to a social media server, a delay parameter τ relaxes the task by constructing the set E' where, for all $t \in E$, $t, t+1, t+2, ..., t+\tau \in E'$. Since LABurst's evaluation compares methods that share the same ground truth, and controlling τ affects the ground truth consistently, comparative results should be unaffected. For the following experiments, the following value is used: $\tau = 2$.

False positives/negatives and true positives/negatives follow in the normal way for some candidate function $\widehat{\mathbb{1}_{E'}}(S_t, t)$: a false positive is any time t such that $\widehat{\mathbb{1}_{E'}}(S_t, t) = 1$ and $\mathbb{1}_{E'}(S_t, t) = 0$; likewise, a false negative is any t such that $\widehat{\mathbb{1}_{E'}}(S_t, t) = 0$ and $\mathbb{1}_{E'}(S_t, t) = 1$. True positives/negatives follow as expected.

4.1.2 The LABurst Model

LABurst combines the language-agnostic flexibility of burst detection techniques with the specificity of domain-specific algorithms. This integration results from ingesting a social media stream, maintaining a sliding window of frequencies for each token contained within the stream, and using the number of bursty tokens in a given time period as an indicator of the moment's impact. Critically, these tokens can be of any language and are neither stemmed, normalized, or otherwise modified. Most other approaches use language models to collapse these various token forms, whereas LABurst leverages this information as a predictor.

In more detail, LABurst runs a sliding window over the incoming data stream S and divides it into slices of a fixed number of seconds δ such that time $t_i - t_{i-1} = \delta$. LABurst then combines a set number ω of these slices into a single window (with an overlap of $\omega - 1$ slices), splits each message in that window into a set of tokens, and tabulates each token's frequency. By maintaining a list of frequency tables from the past k windows up to time t, LABurst constructs features describing a token's changes in frequency. From these features, one can separate tokens into two classes: bursty tokens $\mathbf{B}_t$, and non-bursty tokens $\mathbf{B}'_t$. Following this classification, if the number of bursty tokens exceeds some threshold $|\mathbf{B}_t| \geq \rho$, LABurst flags this window at time t as containing a high-impact moment. In this manner, LABurst approximates the target indicator function with $\widehat{\mathbb{1}_{E'}}(S_t, t) = |\mathbf{B}_t| \geq \rho$ and yields $\mathbf{B}_t$ as the set of descriptive tokens for the given moment.

Unlike the previous chapter, retweets are discarded to avoid spurious bursts

since existing literature shows retweets propagate extremely rapidly, which could lead to false bursts [82].

4.1.3 Temporal Features

To capture token burst dynamics, LABurst uses a set of temporal and graphical features to model these effects, summarized in Table 4.1. These features are calculated per token and normalized into the range $[0, 1]$ to avoid scaling issues. Each feature's relative importance is then examined through an ablation study described later.

Table 4.1: Features

Feature	**Description**
Frequency Regression	Scores a token by how well its frequency fits an exponential curve.
Average Frequency Difference	Score using the difference between the average frequency over the past few minutes and the current minute.
Inter-Arrival Time	The average number of seconds between token occurrences in the previous k windows.
Entropy	The entropy of the set of messages containing a given token.
Density	The density of the @-mention network of users who use a given token.
TF-IDF	The term frequency, inverse document frequency for a each token.
TF-PDF	A modified version of TF-IDF called term frequency, proportional document frequency [85].
BursT	Weighted combination of token's actual and expected frequencies [86].

4.1.3.1 Frequency Regression

Exponential curves are often associated with bursts in usage or cascades in social networks, and capturing whether a token's usage fits this pattern could indicate an important event. To measure this fit, this feature is generated by applying linear regression in logarithmic (log) space to each token's per-slice frequency over the current window. This estimation is performed in log space since exponential curves are linear in log space, and linear regression is a fast approximation for linear curves. Regression results yield an approximation of the frequency's slope (the ratio between vertical change and horizontal change) of a line that best fits this time series. The slope of the best-fitting line is then taken as the token's score since higher slopes indicate steeper exponential curves. This feature is duplicated across token frequency, message frequency, and user frequency.

4.1.3.2 Frequency Differences

Another method for capturing the intensity of a burst in token usage is to compare a token's current frequency with its prior frequencies. Simple first differences (i.e., the difference between the current slice's frequency and previous slice's frequency) can be noisy, however, so this feature compares the current frequency with an average over the previous slices. Eq. 4.1 shows how this feature is calculated, where k is the number of slices in the sliding window. The frequency of a token w for a given time slice t is defined as $\text{freq}_t(w)$. As with the regression feature, this feature is calculated for token, message, and user frequency, meaning that $\text{freq}_t(w)$ returns

raw token frequency, the number of messages containing the token, and the number of users using the token respectively.

$$\text{score}_t(k, w) = \text{freq}_t(w) - \frac{1}{k-1} \sum_{j=t-k}^{t-1} \text{freq}_j(w) \tag{4.1}$$

4.1.3.3 Inter-Arrival Time

When many users are posting about the same event, one might expect all these users to post at approximately the same time. In contrast, if a few users are posting about a less impactful event, long periods of time may pass without seeing a tweet containing a relevant token. Therefore, the amount of lag time between the arrival of two tweets that contain the same token could indicate how impactful an event is. To this end, this feature measures the average number of seconds between token occurrences in the previous k windows.

4.1.3.4 Entropy

In text processing/information theory, Shannon entropy measures the amount of information conveyed by an observation. Accordingly, events that occur very rarely or very frequently convey less information as they are more predictable than events that occur more randomly (e.g., events with a probability of 0.5). Calculating a token's entropy then captures the degree of randomness or information in that token; for instance, stop words occur very frequently and therefore convey little information, and rare strings are not being used often and are likely uninteresting.

Hence, the information entropy for a token w is a useful feature and is calculated according to Eq. 4.2, where $P_t(w) = \frac{\text{freq}_t(w)}{\sum_{i=0}^{V} \text{freq}_t(w_i)}$ and V is the number of unique tokens (or vocabulary size).

$$H_t(w) = -(P_t(w) \log_2(P_t(w))) \tag{4.2}$$

4.1.3.5 Interaction Graph Density

Another indicator for a moment's impact is the number of users discussing the moment. If many of these users are connected to each other through the social network, however, just the number of users could be misleading. For instance, if many of Justin Bieber's followers all start tweeting about a single event, one could see a major spike in this count even though little of interest may be happening in the real world. One way to account for this potential bias is to weight a token by the number of connections shared among the users using the token. Network density captures this information by measuring the ratio between the number of edges present in the network and the number of possible edges. While Twitter's friend/follower network is not captured in the data used here, one can instead build a directed graph G from the interactions among these users where nodes V represent users, and edges E represent retweet and mention interactions. To this end, this model includes graph density as a feature, where density is calculated according to

Eq. 4.3 and binom$(V, 2)$ is the binomial coefficient (or V-choose-2).

$$D(G) = \frac{E}{2 \text{ binom}(V, 2)} \tag{4.3}$$

4.1.3.6 Term-Frequency, Inverse Document Frequency (TF-IDF)

When building search engines, a standard measure for weighting terms is the TF-IDF measure, which balances a token's overall frequency by the number of documents (or tweets here) in which it appears. It is included here as a standard feature from the information retrieval community and is calculated as shown in Eq. 4.4 where N is the number of tweets in the current window, and n_w is the number of tweets containing token w.

$$tfidf(w) = \text{freq}(w) \cdot \log \frac{N}{n_w} \tag{4.4}$$

4.1.3.7 Term-Frequency, Proportional Document Frequency (TF-PDF)

TF-IDF purposely downweights tokens that appear in many documents, but this construct may not be appropriate for detecting bursts in social media. A modified version of this measure, called TF-PDF, was proposed to capture this bursty behavior by Khoo and Ishizuka and is included as a feature here [85, 87].

4.1.3.8 BursT Score

In 2011, Lee, Wu, and Chien proposed a metric, called BursT, that tries to explicitly capture a token's bursty behavior by comparing actual arrival rates with expected values [86]. In some sense, this metric could combine the average difference feature and regression scores described above and is therefore included here as well.

4.1.4 Bursty Token Classification

The LABurst model differentiates between bursty and non-bursty tokens by integrating these temporal features into feature vectors for each token. These vectors are then processed using an ensemble classification algorithm of support vector machines (SVMs) [88] and random forests (RFs) [89].

Training these burst detection classifiers requires both positive and negative samples of bursty tokens. For positive samples, one can identify high-impact events and construct a set of seed tokens that *should* experience bursts along with these events (as done in typical seed-based event detection approaches). Negative samples, however, are difficult to identify since one cannot know all events occurring around the world at a given moment. To address this difficulty, LABurst relies on a trick of linguistics and uses stop words as negative samples, the justification being that stop words are highly but consistently used (i.e., stop words are intrinsically non-bursty). Therefore, LABurst is trained on a set of events with known bursty tokens and stop words in both English and Spanish. This semi-supervised task also includes a self-training phase to expand the list of bursty tokens.

4.2 Evaluation Framework

LABurst was evaluated via two studies: the first compared LABurst's accuracy in discovering events to two baseline methods. The second experiment tested LABurst's domain independence by transferring models trained the sports context to the disaster context.

4.2.1 Accuracy in Event Discovery

The first research question **RQ1** was: Is LABurst's accuracy in identifying key moments competitive with simpler baseline algorithms? To answer this question, an experiment was constructed for enumerating key moments during major sporting competitions. Such competitions were interesting given their large followings (many fans to post on social media), thorough coverage by sports journalists (high-quality ground truth), and regular occurrence (large volume of data), making them ideal for both data collection and evaluation. Target events were also complex in that they include multiple types of events and unpredictable patterns of events around scores, fouls, and other compelling moments of play.

Data was collected from a number of major sporting competitions, and several key moments were identified in each competition. Moments and the times they occurred were extracted from sports journalism articles, game highlights, box scores, blog posts, and social media messages. Determining event times was a non-trivial task, however, as the majority of sports coverage reported in-game times rather than wall-clock times (e.g., World Cup goal times were reported relative to the game's

start). These journalistic sources then comprised the ground truth.

Once a testable data set was established, attention turned to the baseline algorithms, the first of which was a time-series algorithm based on raw message frequency. While results from the previous chapter suggested this algorithm would perform poorly, this approach was popular enough in the literature to warrant inclusion. The second algorithm relied on domain knowledge and seed keywords like those presented in Cipriani and Zhao et al. [14, 47]. Details on these baselines are presented below.

4.2.1.1 Sporting Competitions

To minimize bias, these competitions covered several different sporting types, from horse racing to the National Football League (NFL), to Fédération Internationale de Football Association (FIFA) premier league soccer, to the National Hockey League (NHL), National Basketball Assoc. (NBA), and Major League Baseball (MLB). Each competition also contained four basic types of events: beginning of the competition, its end, scores, and penalties. Table 4.2 lists the events and the number of key moments in each.

This data set tracked four Premier League games in November 2012. For the 2013 World Series between the Boston Red Sox and the St. Louis Cardinals, it covered the final two games on 28 October and 30 October of 2013. In 2014, the data set contained a subset of playoff games during the 2014 NHL Stanley Cup and NBA playoffs and a number of early matches during stages 1 and 2 and the

Table 4.2: Sporting Competition Data

Sport	**Key Moments**
Training Data	
2010 NFL Division Championship	13
2012 Premier League Soccer Games	21
2014 NHL Stanley Cup Playoffs	24
2014 NBA Playoffs	3
2014 Kentucky Derby Horse Race	3
2014 Belmont Stakes Horse Race	3
2014 FIFA World Cup Stages A+B	80
Testing Data	
2013 MLB World Series Game 5	7
2013 MLB World Series Game 6	8
2014 NFL Super Bowl	13
2014 FIFA World Cup Third Place	11
2014 FIFA World Cup Final	7
Total	193

the final two matches of 2014 World Cup. These final games included the 12 July match between the Netherlands and Brazil for third place, and the match on 13 July between Germany and Argentina for first place.

These events were split into training and testing sets; training data covered the 2010 NFL championship, 2012 premier league soccer games, NHL/NBA playoffs, the Kentucky Derby/Belmont Stakes horse races, and several days of World Cup matches in June of 2014. The testing data covered the 2013 MLB World Series, 2014 NFL Super Bowl, and the final two matches of the 2014 FIFA World Cup.

4.2.1.2 Burst Detection Baselines

The first baseline algorithm, referred to below as RawBurst, follows the "activity peak detection" method presented in Lehmann et al. and a similar method presented in Vasudevan et al. [45, 46]. RawBurst takes the difference between the number of messages seen in the current time slice and the average number of messages seen over the past k time slices.

Formally, a series of time slices $t \in T$ is segmented into δ seconds and a social media stream S containing messages m is defined such that S_t contains all messages in the stream between $t-1$ and t. The frequency of a given time slice t is then defined $\text{freq}(t, S) = |S_t|$ and the average over the past k time slices as $\text{avg}(k, t, S)$, shown in Eq. 4.5.

$$\text{avg}(k, t, S) = \frac{\sum_{j=t-k}^{t} \text{freq}(j, S)}{k} \tag{4.5}$$

Given these functions, the difference between the frequency at time t and the average

over the past k slices is defined as $\Delta_{t,k} = \text{freq}(t, S) - \text{avg}(k, t, S)$. If this difference exceeds a threshold ρ such that $\Delta_{t,k} \geq \rho$, an event is detected at time t.

Following Cipriani from Twitter's Developer Blog and others, RawBurst is then modified to detect events using frequencies of a small set of seed tokens $w \in W$ [14]. This domain-specific implementation is referred to as TokenBurst. To convert RawBurst into TokenBurst, the $\text{freq}(t, S)$ function is modified to return the summed frequency of all seed tokens, as shown in Eq. 4.6 where $\text{count}(w, S_t)$ returns the frequency of token w in the stream S during time slice t. These seed tokens are chosen such that they likely exhibit bursts in usage during the key moments of sporting event data, such as "goal" for goals in soccer/football or hockey or "run" for runs scored in baseball. This TokenBurst implementation also includes some rudimentary normalization to collapse modified words to their originals (e.g., "gooaallll" down to "goal"). Many existing stream-based event detection systems use a similar approach to track specific types of events.

$$\text{freq}(t, S) = \sum_{w \in W} \text{count}(w, S_t) \tag{4.6}$$

Since this analysis covers three separate types of sporting competitions, the seed keyword set also includes tokens from each event. Separate keyword lists are avoided to provide an even comparison to LABurst's general nature. These seed tokens are shown in Table 4.3.

Table 4.3: Predefined Seed Tokens

Sport	**Tokens**
World Series	run, home, homerun
Super Bowl	score, touchdown, td, fieldgoal, points
World Cup	goal, gol, golazo, score, foul, penalty, card, red, yellow, points

4.2.1.3 Evaluating Accuracy

LABurst, RawBurst, and TokenBurst were evaluated by constructing a series of receiver operating characteristic (ROC) curves across test sets of the sports data. These ROC curves were compared via their respective areas under the curves (AUCs), generated by varying each method's threshold parameters. The ROC AUC is useful because it is robust against imbalanced classes, which one should expect given the sparsity of high-impact events (i.e., important events are rare, and most of the time, no major event has occurred).

In RawBurst and TokenBurst, the threshold parameters were ρ in $\Delta_{t,k} \geq \rho$. For LABurst, the ROC curve was generated by varying the minimum ρ in $\widehat{\mathbb{1}_{E'}}(S_t, t) = |\mathbf{B}_t| \geq \rho$.

4.2.2 Domain Independence

One of LABurst's design goals is to discover and describe interesting moments regardless of domain. **RQ2** tests whether this goal was met by asking: Can LABurst models trained in one context be transferred to another context and remain com-

petitive?

Detecting key moments within sporting competitions is useful for advertising or automated highlight generation, but a more compelling task is to detect higher-impact events like natural disasters. The typical seed-token-based approach is difficult here as it is impossible to know what events are about to happen where, and a list of target keywords to detect all such events would be long and lead to false positives. LABurst could be beneficial here as one need not know details like event location, language, or type. This context presents an opportunity to evaluate LABurst in a new domain and compare it to existing work by Sakaki, Okazaki, and Matsuo [17]. Thus, to answer **RQ2**, one can take the LABurst model as trained on sporting events presented for **RQ1** and apply them directly to this context.

For this earthquake detection task, LABurst was compared to the TokenBurst baseline using the keyword "earthquake," as in Sakaki, Okazaki, and Matsuo [17]. This evaluation used the two of the most severe earthquakes in Japan between 2014 and 2015: the 7.1-magnitude quake off the coast of Honshu, Japan on 25 October 2013, and a 6.5-magnitude quake off the coast of Iwaki, Japan on 11 July 2014. Rather than generating ROC curves for this comparison, this evaluation compared the lag between the earthquake event and the time in which the two methods detected the earthquake. Ideally, this lag between TokenBurst and LABurst should be small for **RQ2** to be affirmed.

4.3 Data Collection

Data for these evaluations was sourced from three Twitter corpora. The Edinburgh Corpus [90] provided coverage of the 2010 NFL division championship game. A second corpus was sourced from Twitter's firehose source targeted at Argentina during November of 2012, which covered the four Premier League soccer games. All remaining data was extracted from Twitter's 1% sample stream over the course of October 2013 to July 2014. This public sample stream was connected to the Twitter API endpoint without any filter, and retrieved approximately 4,000 tweets per minute.

For each event (both sporting and earthquake), this analysis used all tweets from the 1% stream starting an hour before the target event and ending an hour after the event, yielding over 15 million tweets. Table 4.4 shows the breakdown of tweets collected per event.

4.4 Experimental Results

4.4.1 Setting Model Parameters

LABurst has several tunable parameters, and the following section discusses the method and outcomes for setting these parameters. For LABurst's slice size δ, window size ω, and k previous window parameters, preliminary experimentation yielded acceptable results with $\delta = 60$ seconds, $\omega = 180$ seconds, and $k = 10$. These δ and k parameters were used in both RawBurst and TokenBurst as well.

Table 4.4: Per-Event Tweet Counts

Event	Tweet Count
Training Data	
2010 NFL Division Championship	109,809
2012 Premier League Soccer Games	1,064,040
2014 NHL Stanley Cup Playoffs	2,421,065
2014 NBA Playoffs	500,170
2014 Kentucky Derby Horse Race	233,172
2014 Belmont Stakes Horse Race	226,160
2014 FIFA World Cup Stages A+B	5,867,783
Testing Data	
2013 MLB World Series Game 5	1,052,852
2013 MLB World Series Game 6	1,026,848
2013 Honshu Earthquake	444,018
2014 NFL Super Bowl	1,024,367
2014 FIFA World Cup Third Place	809,426
2014 FIFA World Cup Final	1,166,767
2014 Iwaki Earthquake	358,966
Total	16,305,443

LABurst's classifier implementations relied on the Scikit-learn[2] package for implementations of SVMs, RFs, and the ensemble classifier AdaBoost [91], each of which also provided a number of hyperparameters. For SVMs, the primary hyperparameter is the type of kernel, either linear or higher order. The number of features precluded a direct analysis of the decision plane between bursty and non-burst tokens, so principal component analysis was used to reduce the data to a three-dimensional space. Inspection of this reduced space showed a decision boundary more consistent with a sphere rather than a clear linear plane. Therefore, LABurst's SVM implementation uses the radial basis function (RBF) kernel.

Remaining hyperparameters were determined via distributed parameter grid searches for SVMs and RFs. The grid for SVM's two parameters, cost c and kernel coefficient γ, covered powers of two such that $c, \gamma = 2^x$, $x \in [-2, 10]$. RF parameters were similar for the number of estimators n and feature count c' such that $n = 2^x$, $x \in [0, 10]$ and $c' = 2^y$, $y \in [1, 12]$.

Each parameter set was scored using the AUC metric across a randomly split 10-fold cross-validation set, with the best scores determining the parameters used in the ensemble. The two classifiers were then combined via AdaBoost, yielding the results shown in Table 4.5. These grid search results show RFs perform better than SVMs, and the AdaBoost ensemble outperforms each individual classifier.

[2]http://scikit-learn.org/

Table 4.5: Per-Classifier Hyperparameter Scores

Classifier	**Parameters**	**ROC-AUC**
SVM	kernel = RBF, $c = 64$, $\gamma = 0.0625$	87.48%
RF	trees = 1024, features = 2	88.35%
AdaBoost	estimators = 2	89.84%

4.4.2 Ablation Study

As with hyperparameters, LABurst can use different combinations of the features presented earlier in this chapter, and each feature likely has different predictive capability for this event detection task. To determine each feature's utility, an ablation study builds a set of models in which each feature is removed, and the resulting classifiers are evaluated. These degenerate classifiers are then compared with the full AdaBoost classifier using the same 10-fold cross-validation strategy as above. Table 4.6 shows each model's AUC and its difference with that of the full model. These results suggest the regression and entropy features contribute the most while the average difference features hinder performance.

4.4.3 Event Discovery Results

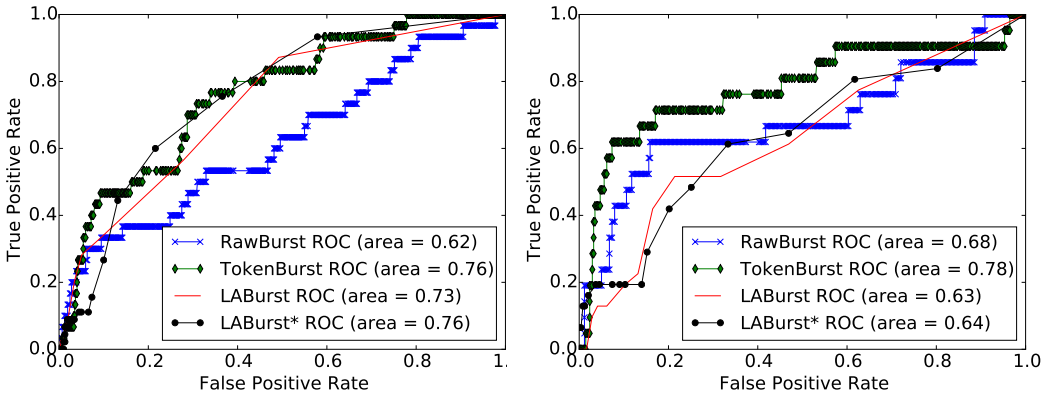
To refresh, (**RQ1**) asked whether LABurst performs as well as existing methods in detecting key moments. Performance across RawBurst, TokenBurst, and LABurst

Table 4.6: Ablation Study Results

Feature Sets	**ROC-AUC**	**Difference**
AdaBoost, All Features	89.84%	–
Without Regression	87.79%	-2.05
Without Entropy	87.94%	-1.90
Without TF-IDF	88.85%	-0.99
Without TF-PDF	89.00%	-0.84
Without Density	89.07%	-0.77
Without InterArrival	89.46%	-0.38
Without BursT	89.52%	-0.31
Without Average Difference	90.56%	0.72

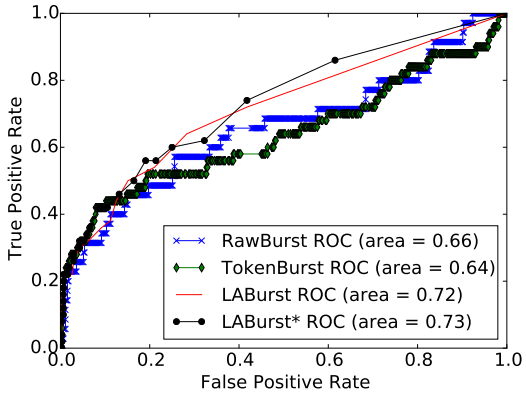
in this task are presented below. These results also include a restricted version of LABurst, called LABurst*, that is trained using the best features from the ablation study (i.e., all features but average difference).

For the 2013 World Series, RawBurst's AUC is 0.62, TokenBurst's is 0.76, LABurst achieves 0.73, and LABurst* yields 0.76. The two LABurst models clearly dominate RawBurst and exhibit performance on par with TokenBurst. During the Super Bowl, RawBurst and TokenBurst achieve an AUC of 0.68 and 0.78 respectively, while LABurst and LABurst* perform worse with an AUC of 0.63 and 0.64. During the 2014 World Cup, both LABurst and LABurst* (AUC = 0.72 and 0.73) outperformed both RawBurst (AUC = 0.66) and TokenBurst (AUC = 0.64).



(a) 2013 World Series

(b) 2014 Super Bowl



(c) 2014 World Cup

Figure 4.1: Per-Sport ROC Curves

4.4.4 Composite Results

Figure 4.2 shows comprehensive performance, which shows ROC curves for all three methods across all three testing events. The blue and green lines showing the ROC curves for RawBurst and TokenBurst respectively. The red line shows the ROC curve for the LABurst model trained using all features, and the black line shows LABurst*. From this figure, LABurst (AUC=0.7) and LABurst* (AUC=0.71) both outperform

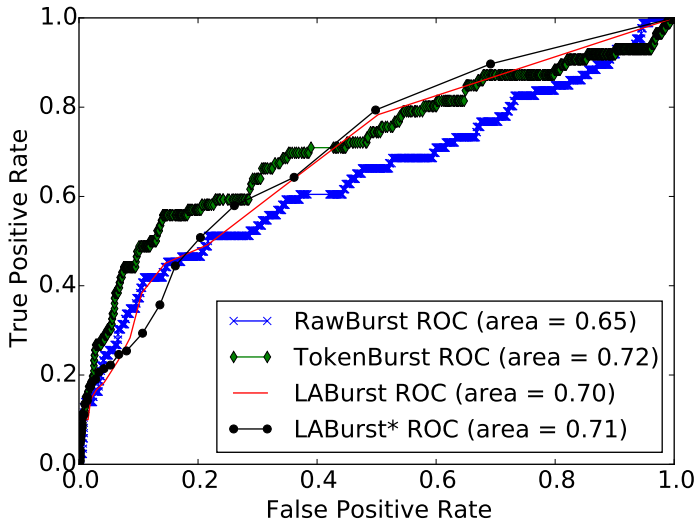
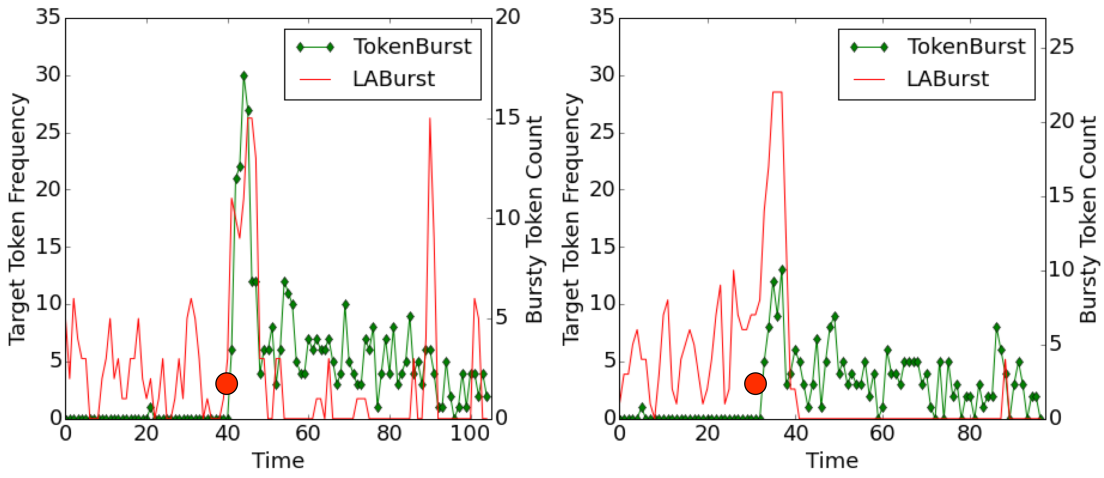


Figure 4.2: Composite ROC Curves

RawBurst (AUC=0.65) and perform nearly as well as TokenBurst (AUC=0.72). Given these results, one can answer **RQ1** in that, yes, LABurst is competitive with existing methods.

Assuming equal cost for false positives and negatives and optimizing for the largest difference between true positive rate (TPR) and false positive rate (FPR), TokenBurst shows a TPR of 0.56 and FPR of 0.14 with a difference of 0.42 at a threshold value of 13.2. LABurst, on the other hand, has a TPR of 0.64 and FPR of 0.28 with a difference of 0.36 at a threshold value of 2. From these values, LABurst achieves a higher true positive rate at the cost of a higher false positive rate. This effect is possibly explained by the domain-specific nature of the test set and TokenBurst implementation, as discussed in more detail in Section 4.5.3.



(a) Honshu Earthquake - 25 October 2013 (b) Iwaki Earthquake - 11 July 2014

Figure 4.3: Japanese Earthquake Detection

4.4.5 Earthquake Detection

RQ2 asks whether adapting LABurst's sports models can compete with existing techniques across domains. To this end, the top-performing LABurst model from the previous section was also applied to Twitter data surrounding known earthquake events in Japan in 2013 and 2014. For comparison, the TokenBurst algorithm provided a baseline using the token "earthquake" as in Sakaki et al. [17].

Figures 4.3a and 4.3b show the detection curves for both methods for the 2013 and 2014 earthquakes respectively; the red dots indicate the earthquake times as reported by the United States Geological Survey. The left vertical axis for each figure reports the frequency of the "earthquake" token, and the right axis shows the number of tokens classified as bursty by LABurst. From the TokenBurst curve, one can see the token "earthquake" sees a significant increase in usage when the

earthquake occurs, and LABurst experiences a similar increase simultaneously. It is worth noting that LABurst exhibits bursts prior to the earthquake event, but these peaks *are unrelated* to the earthquake event since LABurst does not differentiate between the earthquake and other high-impact events that could be happening on Twitter. In addition, the peak occurring about 50 minutes after the earthquake on 25 October 2013 is consistent with an aftershock event[3]. Given the minimal lag between LABurst and TokenBurst's detection, LABurst is shown to be effective in cross-domain event discovery (**RQ2**).

One can now ask what tokens were identified as bursting when the earthquakes occurred. Table 4.7 presents tokens extracted using LABurst. For additional context, several tweets containing these bursty tokens were extracted from Twitter as well: "地震だあああああああああああああああああああああ," "今回はチト使ってないから地震わからなかった," and "地震だ." Google Translate[4] translates these tweets as "Ah ah ah ah ah ah ah ah ah Aa's earthquake," "I did not know earthquake because not using cheat this time," and "Over's earthquake" respectively.

Table 4.7: Discovered Bursty Tokens

Earthquake	**Bursty Tokens**
Honshu, Japan – 25 October 2013	ç 丈, 地, 夫, 怖, 波, 注, 津, 源, 福, 震
Iwaki, Japan – 11 July 2014	び, ゆ, ビビ, 地, 怖, 急, 福, 警, 速, 震

[3]http://ds.iris.edu/spud/aftershock/9761021

[4]http://translate.google.com

4.5 Comparative Analysis

In comparing LABurst with the baseline techniques, it is important to note the strengths and weaknesses of each baseline: RawBurst requires no prior information but provides little in the way of semantic information regarding detected events, while TokenBurst provides this semantic information at the cost of missing unknown tokens or significant events that do not conform to its prior knowledge. LABurst attempts to combine these two approaches by supporting undirected event discovery while yielding insight into these moments by tagging relevant bursting tokens.

4.5.1 Identifying Event-Related Tokens

As mentioned, where the baselines sacrifice either insight or flexibility, LABurst jointly attacks these problems and produces event-related tokens automatically. These tokens may include misspellings, colloquialisms, and language-crossing tokens, which makes them hard to know a priori. The 2014 World Cup provides an illustrative case for such unexpected tokens given its enormous viewership: many Twitter users of many different languages are likely tweeting about the same event. Table 4.8 shows a selection of events from the final two World Cup matches and a subset of those tokens classified as bursting during the events (the list is not exhaustive).

Several interesting artifacts emerge from this table, first of which is that one can get an immediate sense of what happened in the detected moment from tokens presented. For instance, the prevalence of the token "goal" and its variations

Table 4.8: Tokens Classified as Busting During Events

Match	Event	Bursty Tokens
Brazil v. Netherlands, 12 July 2014	Netherlands' Van Persie scores a goal on a penalty at 3', 1-0	0-1, 1-0, 1:0, 1x0, card, goaaaaaaal, goal, gol, goool, holandaaaa, kırmızı, pen, penal, penalti, pênalti, persie, red
Brazil v. Netherlands, 12 July 2014	Brazil's Oscar get's a yellow card at 68'	dive, juiz, penalty, ref
Germany v. Argentina, 13 July 2014	Germany's Götze scores a goal at 113', 1-0	goaaaaalllllll, goallll, godammit, goetze, golllll, gooooool, gotze, gotzeeee, götze, nooo, yessss, ドイツ

clearly indicate a team scored in the first and third events in Table 4.8; similarly, bursting tokens associated with the middle event regarding Oscar's yellow card reflect his penalty for diving. Beyond the pseudo event description put forth by the identified tokens, references to diving and specific player/team names in the first and third events are also of significant interest. In the first event, one can infer that the Netherlands scored since "holandaaaa" is flagged along with "persie" for the Netherlands' player, Van Persie, and likewise for Germany's Götze in the third event (and the accompanying variations of his name). These tokens would be difficult to capture beforehand as TokenBurst would require, and such tokens would likely not be related to every event or every type of sporting event.

Finally, the last artifact of note is that the set of bursty tokens displayed

includes tokens from several different languages: English for "goal" and "penalty," Spanish for "gol" and "penal," Brazilian Portuguese for "juiz" (meaning "referee"), as well as the Arabic for "goal" and Japanese for "Germany." Since these words are semantically similar but syntactically distinct, typical normalization schemes could not capture these connections. Instead, capturing these words in the baseline would require a pre-specified keyword list in all possible languages or a machine translation system capable of normalizing within different languages.

4.5.2 Discovering Unanticipated Moments

Results show LABurst is competitive with the domain-specific TokenBurst, but TokenBurst's specificity makes it unable to detect unanticipated moments, and one can see instances of such omissions in the last game of World Cup. Figure 4.4 shows target token frequencies for TokenBurst in green and LABurst's volume of bursty tokens in red. This graph shows several instances where LABurst exhibits a peak that is missed by TokenBurst. The first, peak #1, includes tokens "puyol," "gisele," and "bundchen," which correspond to former Spanish player Carles Puyol and model Gisele Bundchen, who presented the World Cup trophy prior to the match. While not necessarily a sports-related event, many viewers were interested in the trophy reveal, making it a key moment. At peak #2, slightly more than eighty minutes into the data (which is sixty minutes into the match), LABurst sees another peak otherwise inconspicuous in the TokenBurst curve. Upon further exploration, tokens present in this peak refer to Argentina's substituting Agüero for Lavezzi at the

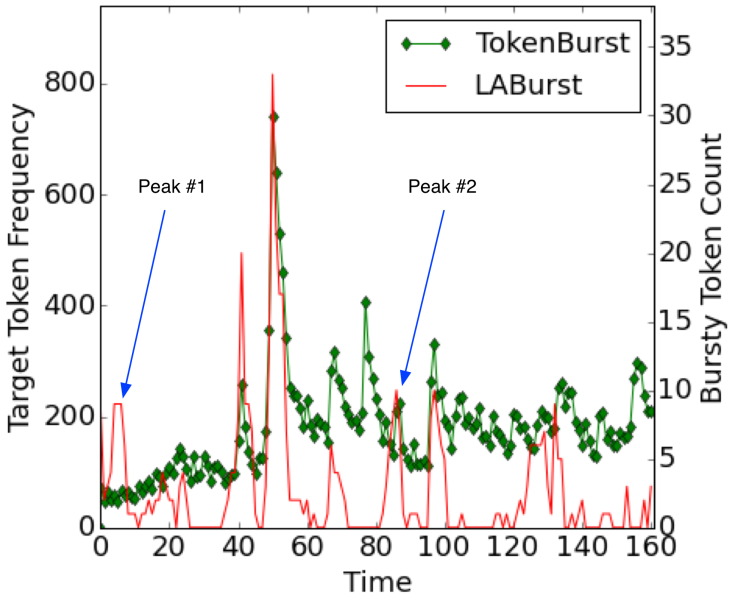


Figure 4.4: Baseline and LABurst Frequencies

beginning of the match's second half.

4.5.3 Addressing the Super Bowl

While LABurst performs as well as the domain-specific TokenBurst algorithm in both the World Series and World Cup events, one cannot ignore its poor performance during the Super Bowl. Since LABurst is both language agnostic and domain independent, it likely detects additional high-impact events outside of the game start/end, score, and penalty events present in the experiment's ground truth. For instance, during the Super Bowl, spectators tweet about moments beyond sports plays: they tweet about the half-time show, commercials, and massive power outages. Since our ground truth disregards such moments, LABurst's higher false-positive rate is less surprising, and TokenBurst's superior performance might result

from its specificity in domain knowledge with respect to the ground truth (i.e., both include only sports data). Hence, LABurst's ability to detect unanticipated moments potentially penalizes it in domain-specific tasks.

LABurst's propensity towards more organic moments of interest becomes obvious when one inspects the tokens LABurst identified when it detected a large burst early on that TokenBurst missed. Approximately four minutes before the game started (and therefore before when TokenBurst would detect any event), LABurst saw a large burst with tokens like "joe", "namath", "fur", "coat", "pimp", "jacket", "coin", and "toss". As it turns out, Joe Namath, a retired American football player, garnered significant attention from fans when he tossed the coin to decide which team would get first possession. Since neither the ground truth data nor TokenBurst's domain knowledge captured this moment, LABurst's detection is counted as a false positive much like the trophy presentation during the World Cup.

4.6 Limitations and Extensions

The approach adopted herein is fundamentally limited regarding tracking potentially interesting events that do not garner mass awareness on social media. Since LABurst presupposes significant bursts in activity during key moments, if only a few people are participating in or following an event, LABurst will be unable to detect moments in that event. This effect is clear in applying LABurst to regular season baseball games: Major League Baseball sees over 2,400 games in a season, and experiments showed too few viewers were posting messages to Twitter during these games to

generate any significant burst. As a result, many key moments in these games were exceedingly difficult to capture via burst detection.

This deficiency leads to a potential opportunity in combining domain knowledge with LABurst's domain-agnostic foundations. For example, one could apply domain-specific filters to the Twitter stream prior to LABurst in the detection pipeline. Since LABurst uses relative frequencies to identify bursts, this pre-filtering step should amplify the signal of potentially bursty tokens in the stream and increase LABurst's likelihood of detecting them. Returning to the baseball example, one could use domain information to filter the Twitter stream to contain only relevant tweets, and the baseball-specific key moments should become more apparent. The following chapter explores this possibility in real-time interest tracking.

4.7 Conclusions

Revisiting motivations, this research sought to demonstrate whether LABurst, a streaming, language-agnostic, burst-centric algorithm, can discover key moments from unfiltered social media streams (specifically Twitter's public sample stream). Results show temporal features can identify bursty tokens and, using the volume of these tokens as an indicator, one can discover key moments across a collection of disparate sporting competitions. This approach's performance is competitive with existing baselines. Furthermore, these sports-trained models are adaptable to other domains with a level of performance exceeding a simple time series baseline and rivaling a domain-specific method. LABurst's performance relative to the domain-

specific baseline shows this method's potential given its omission of manual keyword selection and prior knowledge.

Beyond this comparison, this approach also offers notable flexibility in identifying bursting tokens across language boundaries and in supporting event description; that is, one can get a sense of the occurring event by inspecting bursty tokens returned by LABurst. These features combine to form a capable tool for discovering unanticipated moments of high interest, regardless of language. This technique is particularly useful for journalists and first responders, who have a vested interest in rapidly identifying and understanding high-impact moments, even if a journalist or aid worker is not physically present to observe the event. Possibilities also exist to combine LABurst with other domain-specific solutions and yield insight into unanticipated events, events missed by existing approaches, or events that might otherwise be lost in the noise.

Chapter 5: Real-Time Event Discovery

One of social media's most significant powers is the rapidity with which new information is shared. With recent events showing social media to be a major news source, methods for discovering and summarizing events from this deluge of data in real time are increasingly important. The previous chapter set the foundations for this task in social media streams, and this chapter extends this work to the real-time context with the RTTBurst algorithm, a streamlined version of LABurst.

At the same time, once an event is discovered, a user needs methods for tracking new developments in those events over time. Currently, if the user wants to track important events or topics in these data streams, she must remain at her computer and manually filter through potentially many duplicate posts to track the event. The previous chapter suggests LABurst could be modified to track particular topics of interest, and to this end, this chapter explores topic-directed event discovery/summarization and its performance in the 2015 NIST TREC Microblogs Track [92].

This chapter's thesis is that RTTBurst, through regression analysis across individual token frequencies, can identify high-impact and new information in social media streams in real time and improve classical information retrieval. The follow-

ing sections detail RTTBurst's mechanics, the steps needed to construct a system capable of processing Twitter's 1% stream in real time, and how RTTBurst tracks developments in a particular topic. Following RTTBurst's mechanics is a discussion on tuning RTTBurst given data from TREC's 2015 Microblog track, relative performance between RTTBurst and other systems participating in this track, and methods for integrating RTTBurst with other systems. While the previous chapter focused heavily on the open-domain context, evaluating such an undirected system is difficult given the unbounded number of high-impact events that could occur in any given time period. While RTTBurst's construction as described here is easily applied to these open-domain streams, this chapter focuses on content streams relevant to a pre-defined set of user interests to facilitate evaluation.

This chapter makes the following contributions:

- Presents a real-time streaming algorithm, RTTBurst, for discovering and summarizing relevant moments on Twitter,
- Details RTTBurst's performance relative to similar real-time systems, and
- Demonstrates that burst detection can enhance real-time tracking systems.

5.1 Real-Time Extensions

The LABurst algorithm presented in the previous chapter is designed for data streams, but several features used LABurst are poorly applicable to real-time processing. Network density, for example, does not directly lend itself to streaming computation as the underlying data structure would require modification at each

time step. As shown in the previous chapter, LABurst's most important feature is regression based on a token's frequency. These factors necessitate a streamlined implementation of the burst detection strategy employed by LABurst, which is presented at RTTBurst (for Real-Time Temporal Burst). Furthermore, LABurst's codebase was not designed for distributed computation; RTTBurst, on the other hand, leverages a new platform in distributed computing: Apache Spark[1], which is built on Apache's Hadoop platform[2].

At a high level, RTTBurst uses only one feature as a scoring mechanism: the same linear regression feature described in Chapter 4. As with LABurst, RTTBurst generates per-token frequency vectors for each token in Twitter's 1% stream, where each vector element is the token's frequency in a sliding window. This frequency data is fit to an exponential curve by taking the natural logarithm of each frequency and uses linear regression to calculate the curve's slop and fit. Tokens with steep slopes exhibit exponential increases in frequency and are therefore tagged as "bursty." While LABurst used only the number of bursty tokens to identify high-impact events, RTTBurst extracts tweets that contain these bursty tokens as a means for moment summarization similar to Chakrabarti and Punera [93].

RTTBurst's pipeline is composed of several stages, from collecting the Twitter stream, to finding bursty tokens, to using these tokens to extract the most interesting tweets for moment summarization. Each of these stages is described below.

[1] https://spark.apache.org

[2] https://hadoop.apache.org

5.1.1 Processing the Twitter Stream

For input, RTTBurst uses Twitter's unfiltered public sample stream, corresponding to approximately 1% of the full stream (though larger samples should also work), as provided by Spark's built-in Twitter receiver. Each tweet is then tokenized to extract individual keywords and build quality metrics. Previous results show a large amount of spam in Twitter, so RTTBurst includes a series of filters to remove "low-quality" tweets based on the number of hashtags, web links, token counts, and whether the tweet contained the string "follow" (motivated by the large amount of "follow-me" spam on Twitter). Each of these thresholds are tunable as well.

After this round of quality-based pruning, web links and hashtags are extracted from each tweet, and the tweet is tokenized by replacing remaining punctuation with whitespace and splitting on white space to create a bag-of-words model. Each tweet's bag-of-words is then converted into a time-stamped inverted index matching tokens to the *users* who tweeted them. User-based frequency is used here rather than raw token frequency based on empirical results from the previous chapter.

5.1.2 Identifying Bursty Tokens

This time-stamped inverted index captures changes in a token's usage over time. RTTBurst maintains a sliding window over all tweets generated by the Twitter streaming API within the past two minutes and increments the window by 60-second time slices. Each window therefore overlaps with the previous 60 seconds to smooth the input.

For each two-minute window, RTTBurst calculates the number of users tweeting each token and stores this frequency over the previous N windows. These frequencies are normalized by the number of unique tokens in the past N windows using add-one additive smoothing to correct for tokens with zero occurrences in a single window. Following the features set forth in Chapter 4, linear regression is then used to fit a line to the natural logarithm of this frequency data. By transforming this frequency data to logarithmic space, exponential curves will appear linear, simplifying the linear regression step, and the steeper the slope of the best-fit line, the steeper the exponential growth of the token's usage. Based on this fit, tokens are then scored by the product of the slope of the best-fit line and its coefficient of determination R^2. Since R^2 coefficient is in the range $[0, 1]$, this product reduces scores for highly deviant frequency curves. In this manner, tokens experiencing large bursts in usage, which one would expect to exhibit exponential growth, are scored highly. All tokens with scores below a burst threshold γ and any token whose length is less than four characters are discarded.

5.1.3 Moment Summarization

Every sixty seconds, RTTBurst identifies a new (possibly empty) set of bursty tokens, which corresponds to noteworthy moments in the stream. Some context around the moment is lost with only bursty tokens, however, so RTTBurst also finds social media messages that contain the largest number of bursty tokens and returns those as summarizations, similar to the ReDites system [51].

To this end, every sixty seconds, RTTBurst parses all tweets in the previous N windows to create a subset of tweets containing these bursty tokens. To ensure each summary messages adds new information about the moment, RTTBurst calculates a Jaccard similarity score for each message by comparing it to messages returned in previous windows. Any new message whose Jaccard similarity is above a threshold $J_t = 0.7$ is discarded, and the remaining Twitter messages are sorted by their similarity scores in decreasing order. Finally, the top M most unique messages containing bursty tokens from the past N windows are returned as summaries for this moment.

5.2 Real-Time Topic Tracking

Adding topic tracking to RTTBurst requires additional capabilities for expanding user-provided queries and filtering results. These extensions are also designed according to the specifications presented in the TREC 2015 Microblog track [92].

5.2.1 Query Construction and Expansion

In principle, RTTBurst could track a given topic using a small set of seed keywords, but the dynamic nature of Twitter's vocabulary implies such an approach would miss relevant data. To account for this possibility, RTTBurst includes a methods for expanding a user's query based on its structure and social media data.

This query construction comprises a filtering task, whose goal is to identify tweets relevant to a given set of user interests, described by a few keywords. These

queries are referred to as "interest profiles."

For each interest profile, RTTBurst constructs a query by first removing stop words and then lemmatizing the remaining keywords to get a cleaner keyword set for filtering. As an example, RTTBurst would ingest the profile, "arson fires in inner cities," and produce the following tokens for filtering: "arson," "fire," "inner," and "city."

Once this initial keyword set is constructed, RTTBurst uses them to construct a set of relevant tweets from the previous few weeks of Twitter data. This subset of tweets is then used to build a model of keyword distributions in this set, which constitutes a foreground model for the next stage of analysis. This next stage uses Kullback-Leibler divergence to compare this foreground model against the unfiltered Twitter data set (i.e., the background model). Each keyword in the foreground model is scored based on its deviation from the background, such that keywords that are more common in the foreground than the background model receive higher scores. The top five most divergent keywords are then added to the set of query keywords for further filtering.

5.2.2 Filtering the Twitter Sample Stream

After constructing a set of query keywords, RTTBurst applies this set to the incoming, unfiltered Twitter sample stream. Since RTTBurst already tokenizes each tweet, the filtering stage here calculates the intersection between each tweet's token set and the set of all search keywords. Each tweet with a non-empty intersection set

(i.e., only those tweets that contained at least one keyword from a user's interest profile) is kept for further processing. It is important to note that, while RTTBurst does remove irrelevant tweets by matching tokens from interest profiles, this filtering step occurrs *after* ingesting tweets from Twitter's *unfiltered* sample stream. These tokenized tweets are then converted into a time-stamped inverted index as before, and RTTBurst's processing proceeds as described in the previous section.

5.2.3 Topic-Specific Summarization

In the unfiltered version of RTTBurst, moment summarization ensures dissimilarity in tweets and records all sufficiently unique tweets. For topic-specific summarization, however, RTTBurst performs one last pass through the tweets to select those that were most relevant to the given interest profile. For each candidate tweet stored up to this point, RTTBurst selects only those tweets that contain at least X tokens from the relevant interest profile. All other tweets are discarded.

5.3 Evaluating Real-Time Topic Tracking

RTTBurst was originally developed as an open-domain model and was adapted to the interest tracking domain for the TREC 2015 Microblog track, which focused on identifying new, topically relevant information on Twitter in real time. As mentioned in the track's 2015 overview paper [92], this filtering task's goal was to identify new tweets relevant to a set of given interest profiles. The evaluation occurred in July of 2015 over ten days and was broken across two tasks: a push notification task

that enforced a limit of 10 tweets per interest profile per day and penalized tweets based on the delay between posting and reporting (Scenario A), and a daily email digest task with the relaxed constraint of 100 messages per day and no temporal penalty (Scenario B).

Each tweet returned across all systems were assigned a "qrel" score (for query relevance) by NIST evaluators according to the relevance that tweet had to the interest profile it matched: either "not relevant," "relevant," or "highly relevant" (a score of 0, 1, or 2 respectively). Qrel judgements for retweets were then propagated to their source tweets, and all retweets of this source tweet received the same qrel score. For Scenario A, the mobile notification task, each system was scored based on this relevance score and by how rapidly the system was able to push that tweet to the user. As described in the Microblog Track overview [92], the two Scenario A evaluation metrics were expected latency-discounted gain (ELG), shown in Eq. 5.1, where N is the count of returned tweets, and $G(t)$ is each tweet's gain. A tweet's gain was defined by the relevance score assigned to the tweet by NIST evaluators: 0 if the tweet was not relevant, 0.5 if the tweet was relevant, and 1 if the tweet was highly relevant. This ELG score also received a penalty for latency, such that a tweet received 100 minutes after its creation time received no score. Otherwise, each tweet is penalized according to $MAX(0, (100 - d)/100)$ to model user fatigue in receiving many updates, where d is the number of minutes between a tweet's publication time and it is delivered to the user. Along this same line, Scenario A imposed a limit of no more than ten tweets per topic per day; any additional tweets were not scored. Scenario A also had a second scoring mechanism, normalized cumulative

gain, shown in Eq. 5.2, where Z normalizes by the maximum gain. More details are available in [92].

$$\frac{1}{N}\sum G(t) \tag{5.1}$$

$$\frac{1}{Z}\sum G(t) \tag{5.2}$$

Scenario B's daily email digest task was similar to Scenario A but relaxed the per-day limit to one hundred tweets and scored systems based on normalized discounted cumulative gain (nDCG) for the top ten ranks (nDCG@10), a standard scoring metric for recommender systems.

Prior to the July evaluation period, NIST created a set of 225 topics for evaluation. Of these topics, NIST assessors reviewed tweets returned for 51 of the topics for relevance, resulting in a set of 94,066 scored tweets.

5.4 NIST Evaluation Results

RTTBurst's TREC 2015 evaluation version lacked tweet quality metrics (i.e., it did not filter tweets with many hashtags, many links, or few tokens) and tweet similarity methods to prevent duplicate tweets from being reported. For example, while the original RTTBurst implementation did prevent the same tweet ID from being reported twice, two different tweets with the same content could still be reported, and many Twitter bots spammed the same tweet content with only slight differences (one token at the end of the tweet might differ from one spam tweet to the next).

This official run motivated these quality metrics as RTTBurst reported a significant amount of spam in this early run.

Following the TREC evaluation period and the release of the NIST-judged tweets, these quality metrics were implemented along with a series of post hoc parameter optimization experiments. This optimization used a randomized parameter search over window size $N \in [7, 43]$, maximum tweets delivered per minute $N \in [10, 50]$, and burst thresholds $\gamma \in [0.015, 0.18]$. The range for γ was determined from an early study that calculated mean and maximum token scores for the two weeks prior to evaluation run. For each parameter set, the number of tweets RTTBurst delivered to the user (across all topics), the number of these tweets that did not have associated relevance judgments from NIST (unjudged tweets), and their scores were recorded.

Table 5.1 shows the top-scoring sets for both scenarios from the official run (indicated by the †) and post hoc parameter optimization. Official scores placed RTTBurst 11th out of 32 automatic runs in Scenario A (ranked by ELG) and 4th out of 38 in Scenario B. After parameter optimization, RTTBurst would move up one rank in Scenario A and would remain in fourth in Scenario B. Note that randomized parameter optimization produced more scored tweets than the official run, which was almost silent. It is worthwhile to note that RTTBurst was exceedingly conservative in the emission of tweets, and that this approach occupies a different point in the tradeoff space compared to standard retrieval-based systems.

Table 5.1: Optimized Parameters, Tweets Delivered to Users, and Scores (Best in Bold)

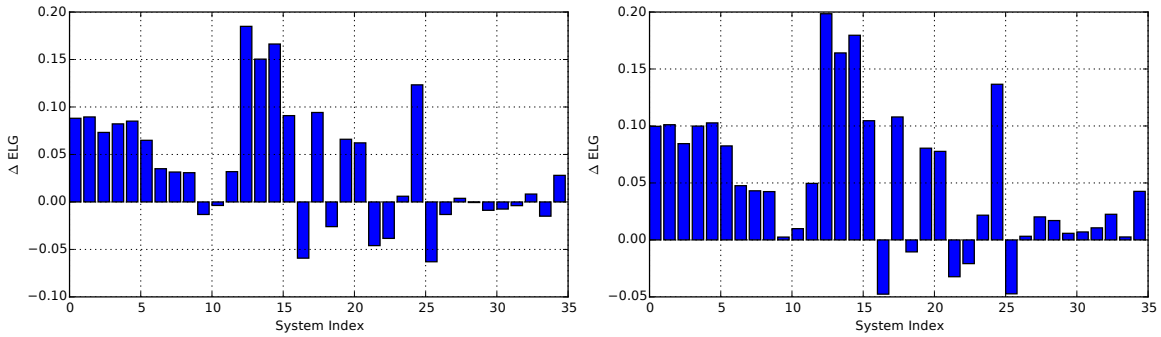
Parameters			Scenario A				Scenario B		
Window Size (N)	Top M Tweets	Burst Threshold γ	Delivered Tweets	Unjudged Tweets	ELG	nCG	Delivered Tweets	Unjudged Tweets	nDCG
† 30	10	0.07	1	0	0.2471	0.2471	1	0	0.2471
37	13	0.036854	29	15	**0.2549**	0.2464	29	15	0.2420
18	34	0.138824	15	7	0.2525	**0.2494**	15	7	0.2479
37	48	0.067306	6	1	0.2506	0.2479	6	1	**0.2489**

† – Parameters used for TREC 2015 evaluation

5.5 Ensembles with RTTBurst

While analyzing results after the official TREC 2015 evaluation, a significant dissimilarity between the tweets returned by RTTBurst and those returned by the other systems became apparent. This observation led to a question: If RTTBurst's burst detection approach were applied to the output of a traditional information retrieval system, could this traditional system's performance be improved? To explore this possibility, a simple gating mechanism was implemented such that, given a set of tweets returned by system A, RTTBurst only allowed those tweets containing a bursty token to be reported (i.e., tweets without bursty tokens were not allowed through the gate).

In addition, to evaluate further the ensembles of RTTBurst gating any given system, additional ensembles were implemented using all possible pairs of systems submitted to the Microblog track. For this investigation, an ensemble system took the union of any two systems' returned tweets and then applied RTTBurst's gating mechanism to filter the results. To ensure that RTTBurst did not benefit from simply combining multiple systems, this experiment scored the outputs of each pair of systems with and without RTTBurst gating. Duplicate tweets were removed from this paired output, the output was ordered by delivery time, and ensembles that delivered more tweets than scenario limits allowed were truncated to the appropriate size.



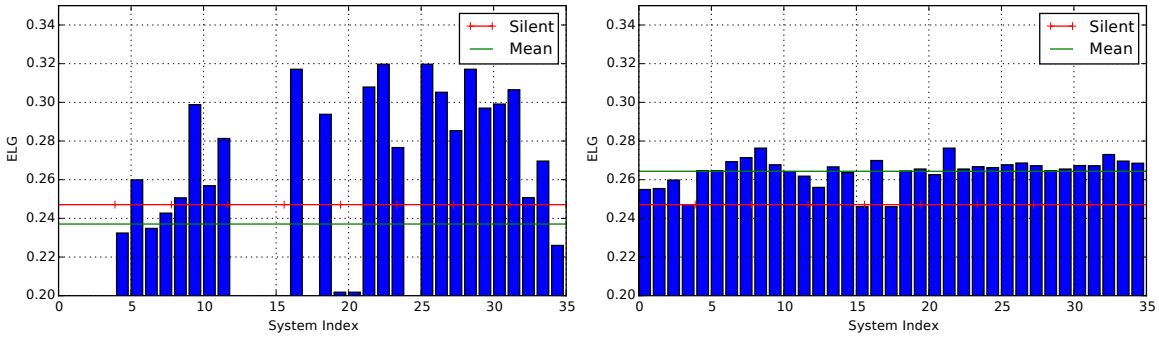
(a) Single System vs. Gated System (b) Single System vs. Paired, Gated System

Figure 5.1: Performance Differences in ELG. Systems arranged alphabetically.

5.5.1 Gating with RTTBurst

Applying RTTBurst's gating mechanism to a single Scenario A system resulted in an average increase in ELG and nCG by 17% and 13% respectively but decreased the ELG of the best-performing system (presented by Tan et al. [94]) by about 19%. A two-sided Welch's t-test on the original scores and the gated scores determined this increase in ELG was statistically significant ($t(33) = 3.28$, $p < 0.01$). In total, RTTBurst increased the performance of 22 systems and decreased the performance of 13 systems, as shown in Figure 5.1a. For Scenario B, gating with RTTBurst resulted in a 9% decrease in nDCG@10.

For system pairs, comparing an individual system with its highest-scoring pair (that is, comparing it to all other systems and taking the pair that achieves the highest ELG) yielded an 11% average ELG increase. Only three systems achieved higher scores without pairing. Using RTTBurst to gate these pairs yielded a 24% increase in ELG over the individual, ungated systems, and five systems performed worse than their unpaired, ungated counterparts. Differences in single system ELG



(a) Best Pair vs. Silent System

(b) Best Gated Pair vs. Silent System

Figure 5.2: Average ELG vs. Silent System (systems arranged alphabetically)

and paired, gated system scores are shown in Figure 5.1b.

For completeness, a comparison was also made between the best pairs' ELG and a silent system (Figure 5.2a) and the best gated pairs of systems (Figure 5.2b). Note these figures show absolute scores rather than score differences. The best pairs of systems did not perform as well as a silent system, but applying RTTBurst as an additional gating filter raised all pairs up to or above the score for a silent system.

5.6 Observations on RTTBurst and Ensembles

Experimental results and data from the official Microblog track exhibited a negative correlation between scores and reported tweets (i.e., systems returning fewer tweets scored higher). This link was first apparent given the score for a system that returned no tweets at all: an ELG, nCG, and nDCG@10 of 0.2471, which placed in the upper third of rankings in both TREC scenarios. During RTTBurst's parameter optimization experiments, this trend became more evident in a strongly negative, nearly linear correlation ($R^2 = 0.8172$) between the number of tweets RT-

TBurst returned and the TREC score produced. This preference towards silence could explain why gating with RTTBurst increased the average score in Scenario A: Summed across all topics, gating reduced the average number of tweets delivered by two orders of magnitude (from 1,600 tweets to 57).

Such a significant reduction in the number of delivered tweets suggested another issue regarding similarity of results returned by the original systems and their gated counterparts. From Figure 5.2, all systems' scores tended to converge to the same value; this convergence would be easily explained if all gated systems were converging to the same set of tweets. By calculating the Jaccard similarity among the returned tweets for each system and then among the gated systems, however, one could determine whether all systems were converging on the same set: For the original systems, the average similarity across all systems was 0.045, and for the gated systems, average similarity was 0.55. Therefore, gains made from gating with RTTBurst were not the result of reducing all output to a common set of tweets. This result suggests burst analysis did provide a valuable relevance signal.

While this convergence is a positive effect for many systems, one must address why RTTBurst decreases the top performing run by Tan et al. [94] by 19%. One possibility is the absence of more sophisticated query expansion techniques. RTTBurst therefore potentially discards many relevant tweets, something that future versions of the system should address.

A further limitation is the potential imbalance in the "bursty-ness" of some topics; thresholds for bursts about celebrities may be too high for more esoteric topics. In its current form, RTTBurst maintains a single burst threshold value

across all topics, which made sense for its original open-domain event detection goal. For tracking specific topics as in TREC's Microblog task, however, thresholds for different topics may vary widely, necessitating separate thresholds for each topic, and some topics may not exhibit bursts at all. Analysis of the posting patterns in tweets scored as relevant in the 2015 TREC evaluation support both possibilities; Figure 5.3 is a bar graph representing the frequency of relevant tweets in logarithmic scale per topic per day in the 2015 evaluation period. The graphs are normalized by the global maximum frequency across all topics and show relatively few topics exhibiting bursts in activity (e.g., topic #243, 348, and 401), with most topics having little difference in day-to-day activity. Normalizing these relevant tweet frequencies by the per-topic maximum instead of the maximum across all topics, however, yields a potentially different view, as shown in Figure 5.4. The red bars in the graphs identify days where relevant tweet frequency exceed three times the MAD, suggesting a burst in activity. Normalizing based on the per-topic frequency is difficult since one does not know the a topic's scale a priori.

This work is also limited by including unjudged tweets in the returned tweet sets, which makes a true performance comparison between official and post hoc runs difficult. That is, while the NIST assessors provided relevance judgments for approximately 94k tweets, the Twitter sample stream over the TREC evaluation period contains around 40 million tweets, so it is highly likely post hoc runs of RTTBurst may return tweets without these judgments. This limitation may be the driving force behind the connection between returned tweet set size and low scores.

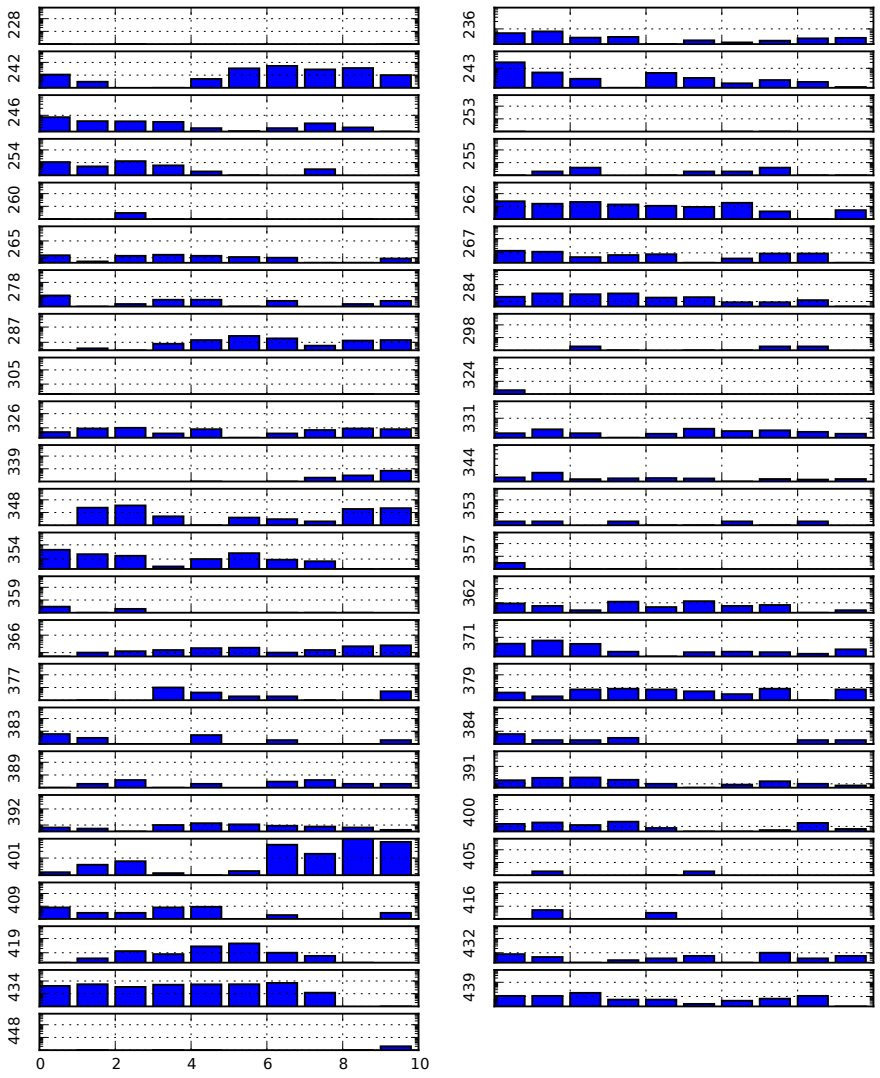


Figure 5.3: Daily Relevant Tweet Frequency Per Topic (Log-Scale), Globally Normalized

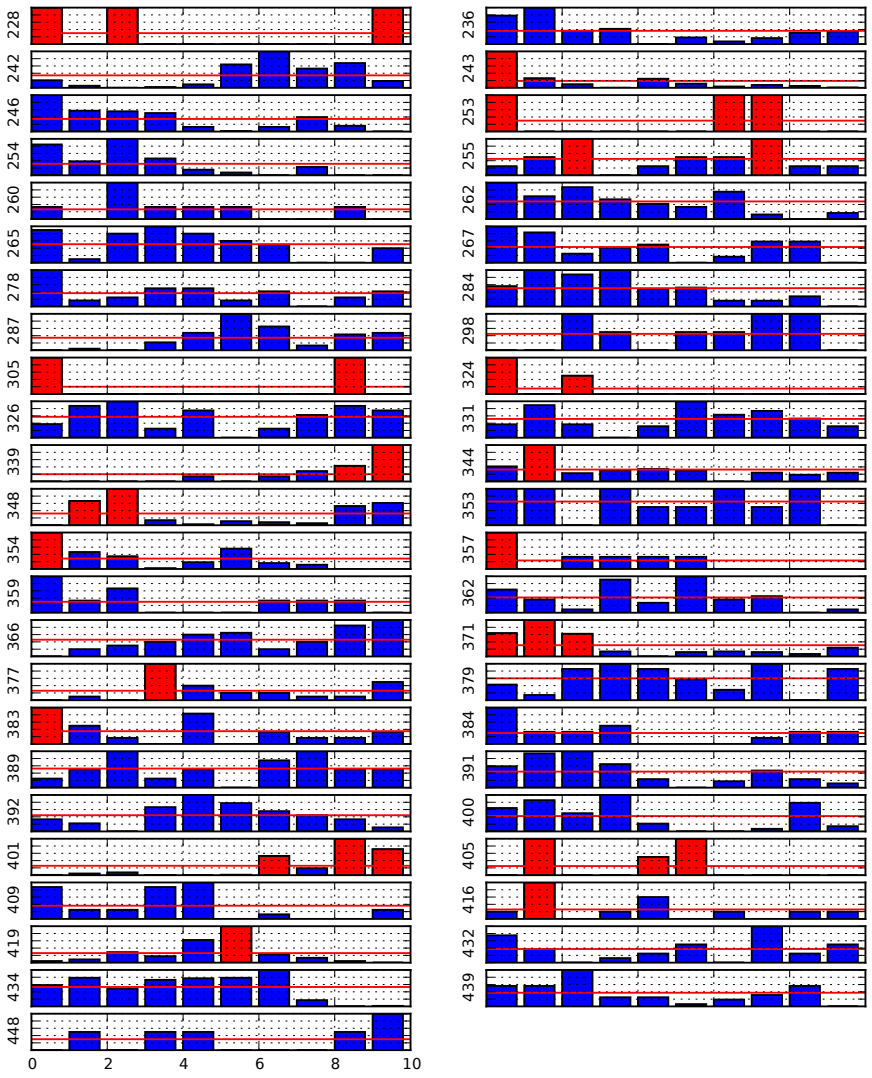


Figure 5.4: Daily Relevant Tweet Frequency Per Topic, Per-Topic Normalization – Red bars exceed three times the MAD

5.7 Conclusions

This chapter extends Chapter 4's language-agnostic burst detection algorithm to the real-time context with RTTBurst, a streamlined implementation of LABurst built on the Apache Spark distributed platform. RTTBurst's real-time topic tracking performance is shown to be competitive with state-of-the-art systems at the TREC 2015 Microblog track. These results demonstrate RTTBurst's regression-based method is effective at identifying new and topical information from social media streams like Twitter in real-time. Furthermore, RTTBurst represents a fundamentally different approach to real-time topic tracking and summarization, with results for combining RTTBurst with other TREC systems yielding improved performance. Given RTTBurst's simple model and its stream-oriented processing, it is at least a useful tool for standalone operation and can be easily integrated into other approaches.

Placing these results in the larger frame of this dissertation, RTTBurst provides a platform for real-time event discovery. It integrates insights from the Twitter's response to terrorism by focusing on topical bursts and merges LABurst's most predictive features for event discovery into a single, end-to-end system. By producing tweet-based event summaries, RTTBurst also provides a starting point for tracking threads of conversation through retweeting and other sharing behaviors, which can be directly leveraged for credibility analysis, as will be shown in the following chapter.

Chapter 6: Evaluating Truth in Social Media

Determining the degree of trust one should place in others (whether human or electronic, spoken or written, individual or group) has been an important strategic decision for as long as humans have lived and worked in communities. Increasing reliance on social media as a journalistic source, both by journalists and readers [24, 95], and recent examples where trust in these sources was misplaced (e.g., misinformation during the 2010 Chilean earthquake [8, 66], 2013 Boston Marathon bombing [3, 33, 96], and others) demonstrate the need for mechanisms to evaluate trust in these sources. This problem is exacerbated by social media's high velocity where traditionally trustworthy entities like major news organizations or well-known journalists disseminate incorrect information to satisfy users' hunger for rapid updates. When first responders, disaster response agencies, and automated systems rely on this information, incorrectly trusting inaccurate content can have real-world impacts.

To address these issues, this chapter investigates truth and credibility in Twitter streams. Disentangling these factors is difficult, however, since the review Chapter 2 shows truth, credibility, and trust are highly nuanced characteristics of information and information sources. This chapter therefore operationalizes "truth" as

accuracy/veracity, or whether a story conforms to facts. Credibility is then defined as *the perception of truth*, or believability, of a story [56]. One should note, while these terms are connected, a true story may not necessarily imply that the story is credible, and a credible story may not necessarily be true.

Following from these definitions, this chapter focuses on evaluating content rather than assessing the accounts posting this content. Content takes the form of threads of conversation in Twitter. One can then analyze the structure present in these threads' networks, textual content, and user features to develop algorithms to determine accuracy and credibility.

Such assessments could be useful to a variety of users and organizations, from first responders looking to allocate resources to journalists looking for accurate information to share with the general public. The goal in this chapter is to develop algorithms that can rapidly assess accuracy to support decision making.

Researchers are working to address these issues in a variety of ways [7–11]. Two recently published, publicly available data sets explicitly address accuracy in social media: the CREDBANK data set [97], and the PHEME rumour scheme data set [98, 99]. This chapter leverages these two data sets to first develop algorithms assessing accuracy in Twitter streams and then explores the connection between accuracy and credibility in this context.

This work makes the following contributions:

- Presents a set of features for evaluating accuracy of conversations in Twitter streams,

- Evaluates the utility of various features in determining accuracy,
- Constructs classification models for differentiating between accurate and inaccurate threads in Twitter, and
- Transfers accuracy classification models between data sets to compare journal-derived accuracy labels to crowdsourced accuracy labels.

6.1 Data Set Descriptions

Recent interest in rumor, accuracy, and credibility in social media has yielded the two aforementioned data sets, PHEME and CREDBANK. While both data sets provide accuracy annotations in Twitter, their constructions differ substantially.

6.1.1 The PHEME Rumor Data Set

The PHEME rumor scheme data set was developed by the University of Warwick in conjunction with Swissinfo, part of the Swiss Broadcasting Company [100]. Swissinfo journalists, working with researchers from Warwick, constructed the PHEME data set by following a set of major events on Twitter and identifying threads of conversation that were likely to contain or generate rumors. A "rumor" in this context was defined as an unverified and relevant statement being circulated, and a rumor could later be confirmed as true, false, or left unconfirmed. Relating "rumor" to this chapter's research, an unconfirmed rumor is a statement with unknown accuracy, a rumor that is later confirmed to be true is accurate, and a rumor later confirmed as false is inaccurate.

PHEME's events included the social unrest in Ferguson, MO in 2014 following the shooting of Michael Brown, the Ottawa shooting in Canada in October 2014, the 2014 Sydney hostage crisis and Charlie Hebdo attacks (mentioned in Chapter 3), and the Germanwings plane crash. The data set also contained conversations around four known rumors: conversations about a footballer, Michael Essien, possibly contracting ebola (later confirmed as false); a rumored secret concert performance in Toronto, Canada (later confirmed as false); a rumor about a museum in Bern, Germany accepting a Nazi-era art collection (later confirmed as true); and rumors about Russian president Vladimir Putin going missing in March 2015 (later confirmed as false).

During each of these events, journalists selected popular (i.e., highly retweeted) tweets extracted from Twitter's search API and labeled these tweets as rumor or non-rumor. This construction resulted in a set of 493 labeled rumorous source tweets. For each tweet in this labeled set, the authors then extracted follow-up tweets that replied to the source tweet and recursively collected descendant tweets that responded to these replies. This collection resulted in a tree-like set of conversation threads of 4,512 additional descendant tweets. In total, the currently available version of PHEME contains 4,842 tweets.

The Swissinfo journalists labeled source tweets for each of these threads as true or false. Once this curated set of labeled source tweets and their respective conversation threads were collected, the PHEME data set was then made available to crowdsourced annotators to identify characteristics of these conversation threads. This crowdsourced annotation task asked annotators to identify levels of support

(does a tweet support, refute, ask for more information about, or comment on the source tweet), certainty (tweet author's degree of confidence in his/her support), and evidentiality (what sort of evidence does the tweet provide in supporting or refuting the source tweet) for each tweet in the conversation.

PHEME crowdsourcing results showed annotators were skewed towards labeling authors as certain, tweets lacking evidence, and tweets being classified as comments. Since commentary tweets neither support nor refute the veracity of the source tweet, these results demonstrate annotators see the majority of tweets as uninformative with respect to the thread's truth. The PHEME authors suggest these results justify altering the annotation scheme and removing the certainty feature from tweets labeled as comments since certainty is ill-defined for opinion or remark.

6.1.2 The CREDBANK Data Set

In 2015, Mitra and Gilbert introduced CREDBANK, a large-scale crowdsourced data set of approximately 60 million tweets, 37 million of which were unique. The data set covered 96 days starting in October of 2014, broken down into over 1,000 sets of event-related tweets, with each event assessed for accuracy by 30 annotators from Amazon's Mechanical Turk (AMT) [97]. At a high level, CREDBANK was created by collecting tweets from Twitter's public sample stream, identifying topics within these tweets, and using human annotators to determine which topics were about events and which of these events contained accurate content. Then, the authors used Twitter's search API to expand the set of tweets for each event.

CREDBANK's initial set of tweets from the 96-day capture period contained approximately one billion tweets that were then filtered for spam and grouped into one-million-tweet windows. Mitra and Gilbert used online LDA from Lau et al. [101] to extract 50 topics (a topic here is a set of three tokens) from each window, creating a set of 46,850 candidate event-topic streams. Each potential event-topic was then passed to 100 annotators on AMT and labeled as an event or non-event, yielding 1,049 event-related topics (the current version of CREDBANK contains 1,377 events). These event-topics were then sent to 30 AMT users to determine the event-topic's accuracy.

This accuracy annotation task instructed users to assess "the credibility level of the Event" by reviewing tweets returned from searching for the event's keywords on Twitter's website (see Figure 5 in Mitra and Gilbert [97]). After reviewing the relevant tweets, annotators were asked to provide an accuracy rating on a 5-point Likert scale of "factuality" (adapted from Sauri et al. [102]) from $[-2, +2]$, where -2 represented "Certainly Inaccurate" and $+2$ was "Certainly Accurate" [97]. Annotators were required to provide a justification for their choice as well. This task appeared difficult for annotators, however, with justifications for lower accuracy ratings of events related to the State of the Union included, "Some rhetoric, not necessarily factual," and, "A lot of conflicting sentiment about state of the union." These comments indicated annotators were conflating accuracy of the event with accuracy of the content.

Once these tweets, topics, event annotations, and accuracy annotations were

collected, Mitra and Gilbert published this data as the CREDBANK data set.[1] Data provided in CREDBANK includes the three-word topics extracted from Twitter's samples stream, each topic's event annotations, the resulting set of event-topics, a mapping of event-topics relevant tweets, and a list of the AMT accuracy annotations for each event-topic (one should note CREDBANK contains no labels of whether an event is accurate or not).

Mitra and Gilbert then turned to analyzing accuracy annotations and found the vast majority ($> 95\%$) of events had a simple majority rating of "Certainly Accurate" [97]. In fact, only a single event had a majority label of inaccurate: the rumored death of Chris Callahan, the kicker from Baylor University's football team, during the 2015 Cotton Bowl (this rumorous event was clearly false as Callahan was tweeting about his supposed death after the game). After presenting this tendency towards high ratings, Mitra and Gilbert explored raising the threshold for majority agreement and found that 76.54% of events had more than 70% agreement, and 2% of events had 100% agreement among annotators. The authors then chose the 70% majority-agreement value as their threshold, and the 23% of events in which less than 70% of annotators agreed were "not perceived to be credible" [97]. While Mitra and Gilbert did not explicitly define "credibility" in the CREDBANK paper, this tendency toward high accuracy ratings and the potential confusion between event accuracy and content accuracy discussed above suggests this metric is measuring *human annotations about credibility.*

PHEME's skewed results may shed light on this annotator bias in CRED-

[1] Available online http://compsocial.github.io/CREDBANK-data/

BANK. Since CREDBANK did not identify subjective content in its accuracy annotation scheme, annotators might have been biased by the majority of commentary in the topic threads they were asked to annotate. Taken together, however, these two data sets provide a resource for analyzing conversational and social networking aspects of rumor propagation in Twitter.

6.1.2.1 Twitter Data Acquisition

While CREDBANK is publicly available, to conform to Twitter's terms of service, the authors are not allowed to share Twitter content directly. Instead, CREDBANK contains a unique identifier for each tweet in the data set (this ID is provided by Twitter), and users of CREDBANK can "rehydrate" this tweet content using Twitter's APIs. The CREDBANK set contains 37 million of these unique tweet IDs.

As mentioned in Chapter 3, the University of Maryland maintains a set of tweets gathered from Twitter's 1% public sample stream from 1 April 2013 to 31 December 2015. This set contains the tweets captured in the first stage of CREDBANK's construction (research has shown tweets captured from the 1% stream are consistent across users and geographic locations [92]). The intersection between the University of Maryland's data set and CREDBANK contains 371,610 tweets to analyze, or about 1% of the tweets in CREDBANK, which is consistent with the accuracy analysis performed on the Twitter 1% sample stream.

6.1.2.2 Labeling CREDBANK Topics

In their published forms, one cannot compare CREDBANK and PHEME directly. While PHEME contains truth labels for each rumorous thread, CREDBANK contains only the collection of annotator accuracy labels. Given the limited number of events with majority labels of "inaccurate," annotator bias towards labels of high accuracy, and previous research showing Twitter contained more inaccurate information than this small percentage [8, 99, 103–105], a labeling approach that could address bias and account for events with conflicting ratings like those about President Obama's State of the Union address is required.

A natural extension of these ratings was to analyze the average accuracy ratings for each event and use this average as a proxy for overall accuracy. This aggregation captures the spread of accuracy ratings better than Mitra and Gilbert's agreement and allowed for bias correction.

Across all of CREDBANK, the global average accuracy rating was 1.7 ($\sigma = 0.25$), and a single event's average could be compared to this global average to determine relative quality. While many events had aggregate ratings close to the global average, a nearly balanced set of credible and non-credible events was constructed by leveraging the global rating's standard deviation. Specifically, any event whose average accuracy rating exceeded the global by at least one standard deviation (≥ 1.95) was labeled as credible, and any event with an accuracy rating less than two standard deviations from the global (≤ 1.4) was labeled as non-credible. Events between these values were indeterminate and left unlabeled. This method

resulted in 99 credible events and 67 non-credible events[2].

6.2 Accuracy Features

This chapter's stated goal is to develop models for evaluating accuracy in social media streams. These models are built on a collection of features, which can be classified into three types: structural features, user-based features, and content-based features. Structural features capture Twitter-specific properties of the tweet stream, including tweet volume and activity distributions (e.g., proportions of retweets or media shares). User features capture properties of users posting about a specific topic; such properties include interaction graphs, account age, friend/follower counts, and Twitter verified status. Lastly, content features express components of the text in tweets about a given topic, like subjectivity and agreement.

6.2.1 Structural Features

Structural features are specific to each Twitter conversation thread. For PHEME, these features are calculated for each source tweet's conversation thread, while features for CREDBANK are generated across a topic's conversation (the set of tweets relevant to that conversation as determined by LDA). Each of these features are calculated across the entire conversation thread. The first set of structural features cover frequencies of different tweet types (one should note a given tweet can have multiple types):

[2]This augmented set of credible/non-credible events is available at https://github.com/cbuntain/CREDBANK-data.

- **Tweet Frequency:** count of tweets posted in this conversation,
- **Hashtag Frequency:** count of tweets posted in this conversation that contain hashtags,
- **Media Share Frequency:** count of tweets posted in this conversation that contain media elements,
- **Mention Frequency:** count of tweets posted in this conversation that contain mentions of other users,
- **Retweet Frequency:** count of retweets posted in this conversation, and
- **Web Link Frequency:** count of tweets posted in this conversation that include a link to another webpage.

For each feature based on a specific type of tweet (e.g., tweet containing hashtags or mentions), the structural feature set also includes proportions of tweet types:

- **Hashtag Proportion:** proportion of tweets that contain hashtags,
- **Media Share Proportion:** proportion of tweets that contain media elements,
- **Mention Proportion:** proportion of tweets that contain mentions of other users,
- **Retweet Proportion:** proportion of retweets posted in this conversation, and

- **Web Link Proportion:** proportion of tweets that include a link to another webpage.

The final structure captures the longevity of a conversation thread on Twitter:

- **Thread Lifetime:** the number of minutes between the first and last tweets in the conversation.

6.2.2 User Features

While the previous set of features focus on conversation characteristics, the following features represent attributes of the users taking part in the conversations. Unlike the previous features, these features are calculated at each time step in the conversation's Twitter stream, where a time step is one minute in duration, and time steps in which no activity occurs are discarded. These features capture the connectedness of these users and the density of interaction between these users. The first few user features analyze account age and reputation (as proxied by followers and friends), intuition suggesting that younger accounts or accounts with fewer followers have less reputation to lose by propagating non-credible information, so events with more content from these accounts might be less credible.

- **Account Age:** the average age of a user's account (in minutes) with respect to Twitter's first day of use (20 March 2006), and

- **Age Difference Between Tweet and Account:** time (in minutes) between when a tweet was posted and when the user posting it created her account (this feature is based on spam accounts that are created to post a tweet),

- **Follower Count:** the average number of users' followers in a conversation,
- **Friend Count:** the average number of friends users have in a conversation,
- **Verified Accounts:** the total number of tweets posted by "verified"[3] users in a conversation,
- **Status Count:** the average number of statuses posted by users in a conversation, and
- **Interaction Network Density:** the density of the interaction graph comprised of users taking part in the conversation.

This last user-centric feature, network density, is measured by first creating a graph representation of interactions between a conversation's constituent users. Nodes in this graph represent users, and edges correspond to mentions and retweets between these users, the intuition being that highly dense networks of users are responding to each other's posts, whereas sparser interaction graphs suggest the conversation's topic is stimulated by influences outside the social network.

6.2.3 Content Features

Content features are similar to user features in that they are calculated for each time step but differ in that they leverage tweet text to determine user response to their conversation threads.

[3]Twitter provides a service called "Verification" for celebrities and organizations to ensure the accounts are operated by legitimate spokespeople. These verified accounts have badges denoting their status.

- **Polarity:** the average positive or negative feelings expressed in tweet content up to the current time step, ranged from $[-1, +1]$ for very negative (-1) to neutral (0) to very positive $(+1)$,
- **Subjectivity:** the average score on a spectrum $[0, +1]$ between objective (0) and subjective (1) tweet content, and
- **Disagreement:** the amount of tweets expressing disagreement with the conversation.

Polarity and subjectivity are features derived from a lexicon built by De Smedt and Daelemans, in which each word is tagged with a part of speech, polarity, sentiment, intensity, and other characteristics [106, 107]. To analyze a given tweet for polarity and subjectivity, the tweet is analyzed to identify sentence boundaries using punctuation, and each sentence is parsed to generate part-of-speech tags for each token in the sentence. These part-of-speech tags then identify the token's sense, which provides an index into the sentiment analysis lexicon. If the token and its associated sense are present in the lexicon, the token's polarity and subjectivity are added to a running sum for that tweet. The tweet's overall polarity and subjectivity are determined by the average values for the tokens that appear in the sentiment lexicon.

As mentioned in PHEME's description, tweet annotations include whether a tweet supports, refutes, comments on, or asks for information about the story presented in the source tweet. These annotations directly support evaluating the hypothesis put forth in Mendoza, Poblete, and Castillo [66], stating that rumors

contain higher proportions of contradiction or refuting messages. CREDBANK, however, lacks these support annotations for individual tweets.

6.2.3.1 Automatically Classifying Disagreement

To address the absent support labels in CREDBANK, a classifier for differentiating between tweets that express disagreement from tweets that express agreement was developed using the support labels in PHEME as ground truth. This classifier used a pipeline for converting tweet text into TF-IDF vectors, tweets with PHEME support labels of "disagree" were added to the positive class, and tweets with support of "agree" were added to the negative class. This data set contained 664 samples of agreeing tweets and 347 disagreeing tweets. A naive Bayes classifier trained on a 10-fold cross validation set of this data performed marginally well, achieving a mean AUC for the ROC curve of 0.7266. A version of this classifier trained on all the data was then stored and applied to the CREDBANK data set to assign disagreement labels for each tweet. While human annotators would be better for this task, an automated classifier was preferable given CREDBANK's size.

6.3 Sampling Methods

As mentioned, PHEME's and CREDBANK's construction processes differ significantly, which leads to differences in how tweets are sampled from these data sets. In PHEME, journalists identified a set of highly shared (i.e., retweeted) tweets as the roots of rumorous conversations and sampled tweets that replied to these roots

to build threads. CREDBANK identified triples of event-related topical keywords and used these keywords as search terms in Twitter's search API to extract a large portion of relevant tweets. As discussed in the tweet acquisition section above, however, owing to limitations in Twitter's historical search API and the data sets available, this research leverages a 1% random sample of these relevant tweet sets. It is important to note that these fundamental differences in construction may hinder comparative analysis, as will be discussed below. Similarly, the methods for identifying topics for which to extract tweets used in PHEME (journalist-identified) and CREDBANK (Twitter-data-driven) may also interfere with analysis.

Beyond sampling topics and tweets, this work also requires a process for sampling accurate and inaccurate conversations/topics to build distributions that can be compared. For each conversation, each of the aforementioned features are evaluated at each time step, and the feature's median value across all time steps is used as that conversation's sampled observation for that feature (to ensure resilience against outliers). This set of median values across all true and false conversations then comprise the distributions for each feature. Therefore, even though the parent distributions for each feature may not be normally distributed (as one would expect for skewed events like retweets), these sample distributions for the medians are approximately normal. One could develop a more sophisticated sampling process that instead uses characteristics of the underlying data generating processes for each feature, but calculating median values is computationally less expensive, which is important given the motivation for rapid assessment here.

6.4 Feature Analysis

The previous section presents a feature collection covering a variety of structures and behaviors in rumorous conversation threads on Twitter. While intuition suggests each feature may be useful in assessing accuracy, these features likely differ in importance for this task. The following section evaluates each feature's predictive power across three studies: comparing feature distributions to determine whether a feature differs significantly between accurate and inaccurate conversations, leveraging ensembles of tree-based learning algorithms to evaluate Gini importance [108], and performing ablation studies to find an optimal feature set. These evaluations are performed on the PHEME data set since its evaluations are backed by input from journalists rather than crowd-sourced and inferred labels.

6.4.1 Statistical Differences

This first study compared feature values from the sample of credible conversation threads to those of the non-credible threads using a two-tailed Welch's t-test. A two-tailed Welch's t-test assuming heteroskedasticity is appropriate here given the sample distributions over the medians should be normally distributed. Furthermore, to address issues of multiple comparison and correct for the experimentwise error rate, a Bonferroni correction is used. Therefore, for the experimentwise significant level $\alpha = 0.05$ and $k = 22$ features to test, a feature is said to be significantly different if $p \leq \alpha_B = 0.05/22 = 0.002272$.

For features calculated across time steps (e.g., polarity or network density), the

cumulative values from the final time step were used. For example, if a conversation thread covered sixty minutes, this test would use the feature's value at the sixtieth minute. Since these time-dependent features were constructed using running sums, this construct equated to evaluating features over the entire conversation, thereby removing temporal dependency. Results from these studies are shown in Table 6.1.

These results show only one feature is significantly different between the two classes after applying the Bonferroni correction: the conversation thread's lifetime (or length in minutes). Disagreement and the number of verified accounts are close but are not significantly different after applying the Bonferroni correction.

6.4.2 Gini-based Feature Importance

Analyzing statistical differences in features shows differences between credible and non-credible conversation threads but does not directly estimate which features were most useful for delineating between credible and non-credible threads. Many tree-based classifiers estimate this utility when determining features on which to split. One method for calculating this utility uses the Gini Importance measure, also called the mean decrease in impurity, which is a measure of the number of samples a given feature separates across all nodes in the tree that use this feature. Higher Gini Importance values for a given feature show the feature reduces impurity (i.e., mismatched labels) in a subset more than features with lower Gini Importance values. Gini Importance is also well-suited for ensembles of trees by averaging a feature's Gini Impurity across all trees in the ensemble.

Table 6.1: Statistical Differences Across Features

Feature	**Credible Mean**	**Non-Credible Mean**	**t-Statistic**	p
Tweet Frequency	226.3041	181.5535	1.3913	0.1651
Retweet Frequency	211.7193	163.2642	1.5342	0.1259
Web Link Frequency	108.4094	73.5786	1.6576	0.0984
Media Frequency	84.4737	71.1635	0.6092	0.5428
Hashtag Frequency	134.5439	119.6164	0.5187	0.6043
Mention Frequency	225.3041	180.4591	1.3945	0.1641
Retweet Proportion	0.8504	0.8048	2.1345	0.0335
Web Link Proportion	0.4957	0.3787	2.3949	0.0172
Media Shares Proportion	0.3588	0.3171	0.8652	0.3876
Hashtag Proportion	0.5636	0.6066	-0.9165	0.3601
Mention Proportion	0.9834	0.9859	-0.7021	0.4831
Thread Lifetime	1,367	2,683	-3.7971	0.0002
Account Age	2,789,885	2,769,758	0.5633	0.5736
Tweet-Age Difference	1,832,055	1,790,337	1.2481	0.2129
Follower Count	18,184.42	18,826.63	-0.1488	0.8818
Friend Count	1,043.50	1,023.51	0.3231	0.7468
Verified Accounts	5.7251	3.327	3.0006	0.0029
Status Count	22,522	23,502	-0.878	0.3806
Network Density	0.0406	0.0412	-0.0719	0.9427
Polarity	0.0232	0.004	0.9843	0.3257
Subjectivity	0.2279	0.2471	-0.6879	0.492
Disagreement	0.0065	0.0153	-2.9922	0.003

This study made use of a random forest classifier with 100 trees and calculated the Gini Importance for each feature, as shown in Table 6.2, which is sorted in order of decreasing importance.

As with statistical difference results, thread lifetime is the most important feature. The majority of remaining features cluster around the average, with the frequency-based features tending to be the least important.

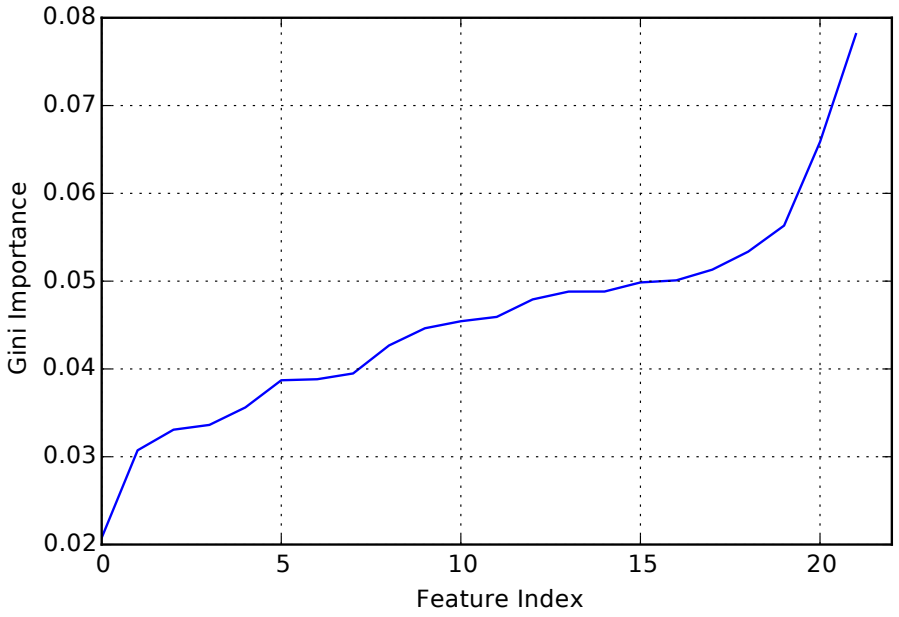


Figure 6.1: Gini Importance Values

6.4.3 Feature Ablation

The final study on feature analysis constructs a classifier from all features, evaluates that classifier's performance, then iteratively removes features. The classifier created at each iteration is also evaluated, and its performance is compared against the

Table 6.2: Gini Importance Across Features (ordered by decreasing importance)

Feature Index	**Feature**	**Gini Importance**
21	Thread Lifetime	0.0781
20	Disagreement	0.0659
19	Follower Count	0.0563
18	Polarity	0.0534
17	Account Age	0.0513
16	Subjectivity	0.0501
15	Tweet-Age Difference	0.0499
14	Retweet Proportion	0.0488
13	Friend Count	0.0488
12	Status Count	0.0479
11	Hashtag Proportion	0.0459
10	Verified Accounts	0.0454
9	Web Link Proportion	0.0446
8	Mention Frequency	0.0427
7	Retweet Frequency	0.0395
6	Tweet Frequency	0.0388
5	Mention Proportion	0.0387
4	Network Density	0.0356
3	Hashtag Frequency	0.0336
2	Web Link Frequency	0.0331
1	Media Proportion	0.0307
0	Media Frequency	0.0208

original classifier. This method demonstrates which features improve versus hinder performance: Removing (or ablating) important features will result in a large decrease in performance, and removing unimportant or bad features will either increase performance or have little impact.

Classifiers used in this study consisted of extra-tree classifiers using 100 estimators, and they were evaluated by the mean ROC-AUC over a 10-fold cross validation set. Each feature estimate was performed twenty times with random permutations of the data set to reduce variance in scores. Feature ablation results are shown in Table 6.3.

Thread lifetime, author follower count, disagreement, and proportion of mentions contributed the most to the classifier performance (as their removal results in the largest drops in performance). The performance impact after removing the mention proportions feature was steeper than between removing any other two features, so the first four features appeared the most important.

As a follow-up to these results and to determine the minimum number of features necessary, an additional recursive feature elimination experiment was performed. This study removed the least contributing feature iteratively, re-running a round of ablation with the remaining features until a single feature remained. The resulting curve of ROC-AUC scores then showed a maximum score for the ideal number of features. Determining which feature to remove used the same ROC-AUC, 10-fold cross-validation method as above. The list of removed features is shown in Table 6.4, with the resulting score curves shown in Figure 6.2.

These results show the best-performing classifier included about half of the

Table 6.3: Feature Ablation Results

Feature Set	**ROC-AUC**	**Difference**
All Features	68.91%	–
Without Thread Lifetime	66.30%	-2.61%
Without Follower Count	67.12%	-1.79%
Without Disagreement	67.14%	-1.77%
Without Mention Proportion	67.24%	-1.67%
Without Verified Accounts	67.69%	-1.22%
Without Tweet Frequency	67.69%	-1.21%
Without Mention Frequency	67.71%	-1.19%
Without Subjectivity	67.72%	-1.19%
Without Web Link Proportion	67.74%	-1.16%
Without Web Link Frequency	67.86%	-1.05%
Without Account Ages	67.89%	-1.02%
Without Polarity	67.99%	-0.92%
Without Friend Count	68.06%	-0.85%
Without Media Frequency	68.11%	-0.80%
Without Media Proportion	68.14%	-0.77%
Without Status Count	68.21%	-0.70%
Without Retweet Proportion	68.25%	-0.66%
Without Hashtag Frequency	68.25%	-0.66%
Without Tweet-Age Difference	68.28%	-0.63%
Without Hashtag Proportion	68.29%	-0.62%
Without Retweet Frequency	68.39%	-0.52%
Without Network Density	68.56%	-0.35%

Table 6.4: Recursive Feature Eilimination Results (* denotes maximum)

Features Removed	**Removed Feature**	**ROC-AUC**
0	–	68.91%
1	Network Density	68.56%
2	Tweet-Age Difference	68.83%
3	Polarity	69.07%
4	Media Proportion	69.30%
5	Friend Count	69.17%
6	Web Link Proportion	68.97%
7	Retweet Proportion	69.21%
8	Verified Accounts	69.37%
9	Status Frequency	69.21%
10	Media Frequency	69.32%
11	Hashtag Frequency	70.09%
12	**Retweet Frequency**	**70.23%**
13	Tweet Frequency	70.02%
14	Mention Proportion	69.96%
15	Account Ages	69.43%
16	Hashtag Proportion	68.76%
17	Disagreement	67.81%
18	Follower Count	66.53%
19	Subjectivity	65.03%
20	Mention Frequency	60.43%
21	Thread Lifetime	58.55%

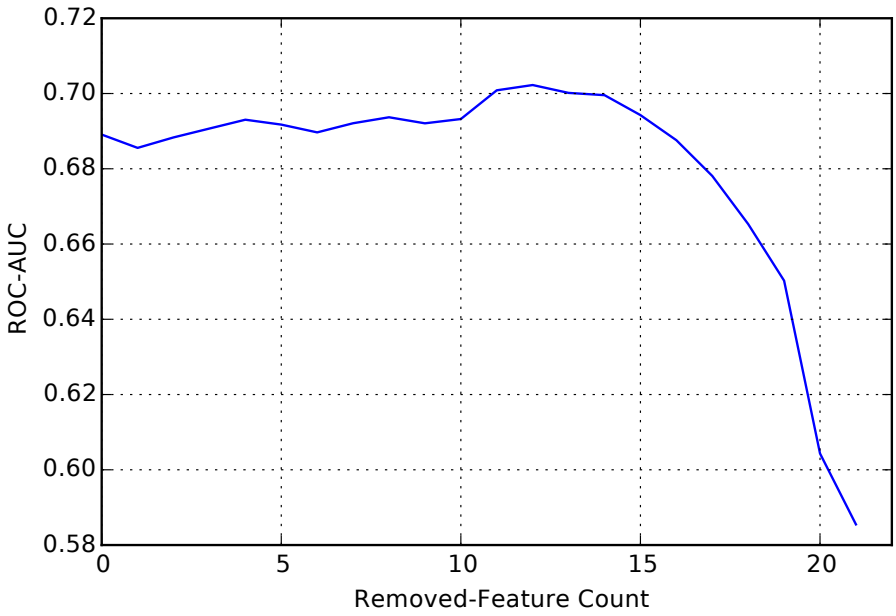


Figure 6.2: Recursive Feature Elimination Results

features: web link frequency, thread lifetime, mention frequency, subjectivity, author follower count, disagreement, hashtag proportion, account age, mention proportion, and tweet frequency. Including thread lifetime and disagreement was consistent with other feature analysis results.

6.5 Models of Accuracy

Feature analysis results show the best-performing RF classifier for PHEME uses ten features and achieves an ROC-AUC score of 70.23%. This section takes this classifier and feature set as the baselines for exploring accuracy models in PHEME. Part of this study focuses on evaluating minimum observation times necessary to achieve this performance (i.e., how long must a conversation thread be observed before a

sufficiently accurate evaluation can be made). Then, these models are applied to the CREDBANK data set to determine how well these accuracy models transfer to crowdsourced assessments.

6.5.1 Accuracy in PHEME

The previous feature analysis section shows which features contribute to accuracy but not how those features are combined to determine accuracy. While non-linear SVMs, random forests, deep neural networks, and other classification models can achieve high accuracy, they do not lend themselves to explaining how that accuracy was achieved in an understandable or intuitive manner. Explainability is an important characteristic for journalist support tools, as users are less likely to trust a simple black box system.

To address this issue, a linear SVM model was trained on all the PHEME data using the ten most predictive features from the previous section. These features' coefficients were then extracted from the model to give insight into how these features interact. These weights denote how much each feature contributed to the overall classification, with positive weights pushing the label towards credible, and negative weights pushing the label towards non-credible. These results are shown in Table 6.5.

These results show thread lifetime has the strongest effect on accuracy with the largest absolute weight; the conversation thread's lifetime is also highly negative, which one can interpret as the longer the conversation takes, the more likely

Table 6.5: Linear SVM Feature Weights for Accuracy

Feature	**Weight**
Account Age	0.1958
Disagreement	-1.8587
Follower Count	-0.08834
Hashtag Proportion	-0.3522
Mention Frequency	0.5639
Mention Proportion	-0.08516
Subjectivity	-0.4068
Thread Lifetime	-3.1098
Tweet Frequency	0.5588
Web Link Frequency	0.7631

it is to be non-credible. This result is consistent with the statistical analysis for thread lifetime shown above, in which the average accurate conversation persisted for about half as long as the average inaccurate thread. Similarly, the amount of disagreement present in a conversation is also strongly indicative of a inaccurate topic, with disagreement having the second largest weight magnitude and a highly negative weight as well. Inaccurate threads having higher amounts of disagreement is intuitive and also consistent with the statistical differences shown above and the hypothesis put forth by Mendoza, Poblete, and Castillo [66].

While the remaining weights are weaker, a few observations are interesting: The overall number of mentions present in a conversation indicates the conversation should be accurate, but the proportion of mentions suggests the conversation should be inaccurate. Though this result seems paradoxical, when taken with the tweet frequency feature, it suggests that having many tweets with fewer mentions indicates accurate conversation. Author account age and and frequency of web links are also intuitive in that conversations in which older accounts participate and have more instances of links to third-party content suggest more accurate content. Surprisingly, the more followers the average conversation participant has, however, the less likely a topic is to be accurate, though this feature has one of the weakest weights. Lastly, the subjectivity feature has a weakly negative weight, which one might expect given that higher values for the subjectivity feature denote users are posting more subjective content.

6.5.1.1 Minimum Observation Times

Another important factor about credibility is the amount of observation time necessary before a accuracy assessment can be made. While one could expect a high-quality accuracy assessment after several days of observing a conversation thread, such an assessment is of limited utility for first responders or decision makers who need to take action rapidly.

To determine the effect observation duration has on model performance, an experiment was developed in which the observation time (in minutes) began two minutes into each conversation and iteratively increased by five-minute intervals to four hours. Thread lifetime was modified for this experiment to return the minimum between the full thread's length and the current observation interval. For each observation duration, an RF model was trained only on data during that duration, and the resulting model was scored using 10-fold cross validation as before. Results are shown in Figure 6.3.

Within the first few minutes, classifier performance is slightly better than random, but after observing approximately twenty minutes of the conversation, the model's score increases significantly. After twenty minutes, the score decreases and slowly climbs back up to its maximum slightly more than two hours into the conversation.

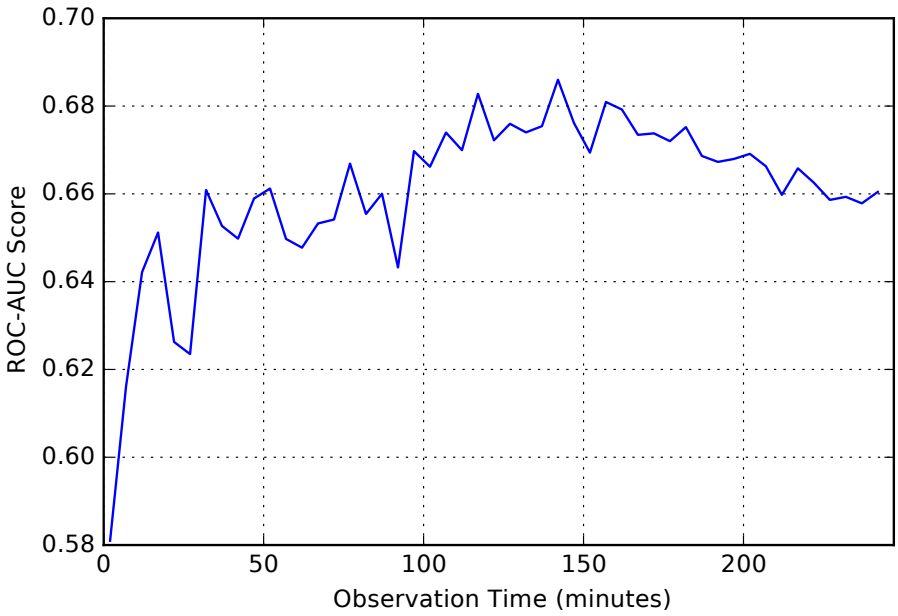


Figure 6.3: Model Scores Across Observation Times

6.5.2 Transferring PHEME Models to CREDBANK

Having explored accuracy and its features in the PHEME data set, the next question concerns how well these models perform on crowdsourced. To answer this question, two RF models were trained on the PHEME data and applied to CREDBANK: the best-performing model from the PHEME data set, a 100-estimator RF trained with the best ten features, and an additional model using all possible features. These models were then evaluated by ROC-AUC according to the inferred labels discussed above. Results for these models are shown in Figure 6.4.

PHEME's models performed no better than random guessing for CREDBANK's inferred accuracy labels, with the model using all features performing

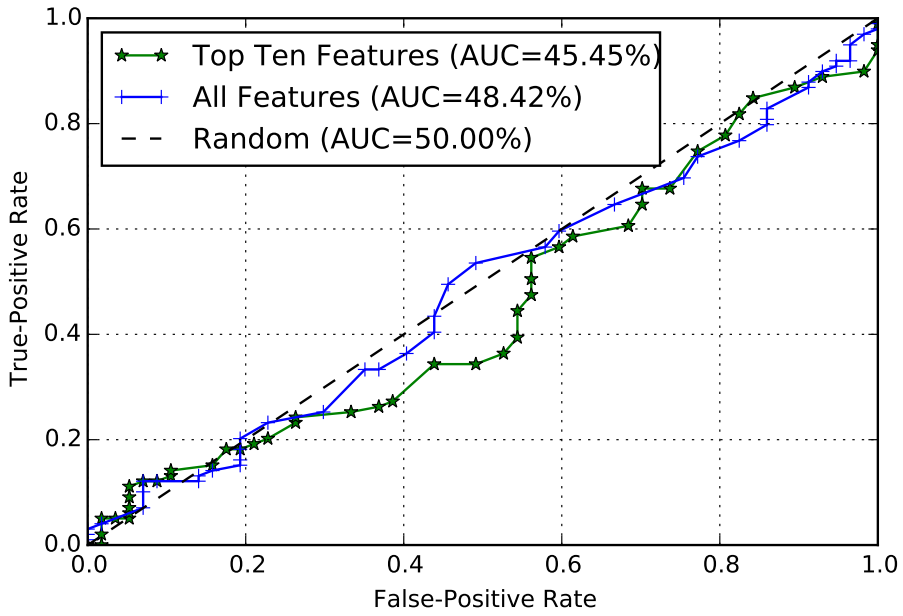


Figure 6.4: Transferring Accuracy Models to CREDBANK

slightly better than the top-ten-feature model. To determine whether this poor performance stems from the accuracy labeling inference used in CREDBANK, these two models were also used to rank the CREDBANK topics based on label probability (i.e., the probability that a given topic was credible), and this ranking was compared to the ranking induced by taking the average annotator accuracy rating for each topic. $p-$Values were then calculated using Kendall's τ to measure correlation between these two rankings: The top-ten model had $p = 0.4273$, and the all-feature model model had $p = 0.9511$, meaning neither model had a significant correlation with the annotator-based rankings.

6.5.3 Accuracy in CREDBANK

With the PHEME models' poor performance in CREDBANK, an open question was whether accuracy in CREDBANK could be assessed automatically. Since all the same features were already calculated for the labeled CREDBANK topics, a new model was trained directly on CREDBANK and evaluated using 10-fold cross-validation. This new model (shown in Figure 6.5) performed much better than the transferred models, with a ROC-AUC score of nearly 92%, suggesting that the accuracy annotations provided by CREDBANK's crowd source users differed in some fundamental way from the truth labels journalists provided in PHEME. This difference was further confirmed by an analysis of feature importance in CREDBANK, shown in Table 6.6, where large numbers of web links, media, and verified accounts have the highest scores.

6.6 Observations on Accuracy and Credibility

Investigations into automatically assessing accuracy in PHEME suggest first that the duration of a conversation and the amount of disagreement present within that conversation are the two most important features. While several other features (primarily author follower count and mentions) also contribute to accuracy in PHEME, disagreement and thread lifetime are always near the top in each feature analysis test. Social network-based and structural features (e.g., network density, participating verified accounts, hashtag usage, etc.) are less important in PHEME. In contrast, frequency of web links, media, and verified accounts are some of the most

Table 6.6: Gini Importance in CREDBANK (ordered by decreasing importance)

Feature Index	**Feature**	**Gini Importance**
21	Web Link Frequency	0.1141
20	Media Frequency	0.0814
19	Media Proportion	0.0735
18	Verified Accounts	0.0734
17	Web Link Proportion	0.0529
16	Hashtag Frequency	0.0482
15	Tweet Frequency	0.0476
14	Tweet-Age Difference	0.0420
13	Mention Frequency	0.0414
12	Retweet Proportion	0.0402
11	Retweet Frequency	0.0401
10	Thread Lifetime	0.0388
9	Hashtag Proportion	0.0363
8	Account Ages	0.0357
7	Follower Count	0.0354
6	Mention Proportion	0.0354
5	Friend Count	0.0342
4	Status Count	0.0307
3	Network Density	0.0281
2	Objectivity	0.0260
1	Polarity	0.0249
0	Disagreement	0.0195

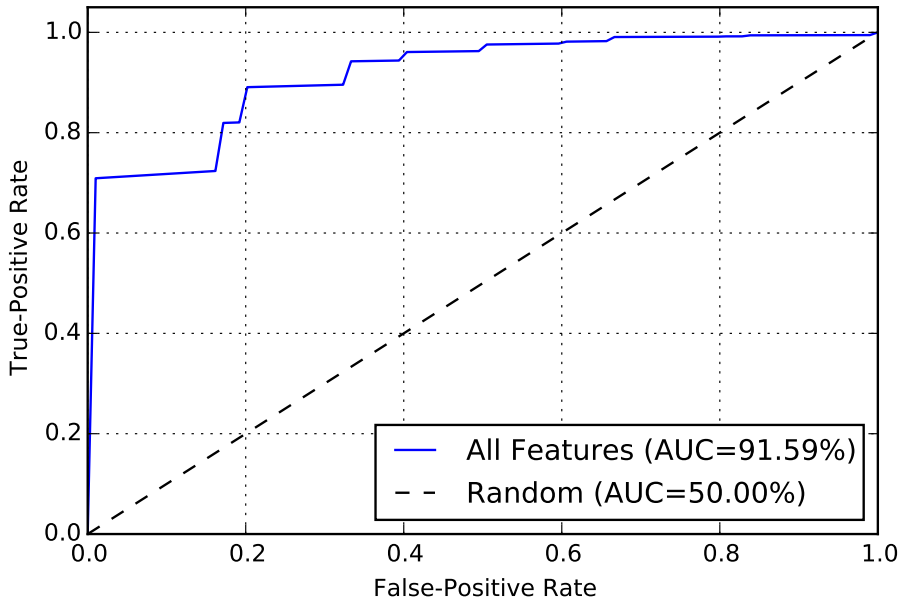


Figure 6.5: Accuracy Models to CREDBANK

important features of accuracy in CREDBANK.

Diverging feature importance between PHEME and CREDBANK suggest the aspects of accuracy addressed in the two data sets are fundamentally different. While PHEME captures accuracy in an objective fashion as determined by journalists, CREDBANK captures the perception of accuracy (or how believable are tweets about a given topic) as determined by crowdsourced workers. This difference highlights the distinction between accuracy and credibility discussed at the beginning of this chapter: Relying on journalists to assess accuracy in social media measures veracity and objective truth, whereas relying on crowdsourced assessments in CREDBANK instead measure whether a topic is believable, or credible, and these two measures do not necessarily imply the other. Given the nuanced differences be-

tween truth and believability, future research could explore these factors and how different patterns of communication in social media influence them.

CREDBANK's orientation towards perceptions of accuracy could explain the importance of "off-site support" in accurate events, where "off-site support" refers to links to content outside of the Twitter network, like webpages and media. Since frequency and proportion of media posts are positively correlated to labels of accuracy in CREDBANK, exposure to other sources reinforcing a topic may increase its believability, or perception of credibility, by providing visual confirmation of the event [11].

6.6.1 Limitations and Future Work

A main limitation of this work is in the differences between PHEME and CREDBANK, both in their construction and the different events and timeframes they cover. First, PHEME is based on tree-like threads of conversation, starting with a highly-retweeted source tweet and analyzing replies to that tweet. CREDBANK, on the other hand, is less structured, with tweets grouped by topic similarity, which obfuscates the threaded structure present in PHEME. This difference may explain why conversation lifetime is more important in PHEME than in CREDBANK. In PHEME, the majority of tweets in accurate threads happen early in the conversation, with little content posted after the first day, whereas inaccurate threads persist for nearly twice as long. This bias towards early content for accurate topics is consistent with Zubiaga's analysis of the average amount of time necessary to resolve

whether a rumor is true or not: approximately two hours for true rumors and fifteen hours for false rumors [100]. CREDBANK, however, shows virtually no difference in conversation lifetime, which may be an artifact of pre-specified observation periods used in CREDBANK's crowdsourcing task. Future work could address this problem and identify threads of conversation in CREDBANK by identifying highly retweeted tweets in a topic and extracting replies from the CREDBANK set (if these replies are present). Extracting these threads from CREDBANK would support a fairer comparison with PHEME.

Similarly, PHEME's construction includes annotator-provided labels on whether a tweet agrees or disagrees with the thread's topic/originating tweet. Constructing this feature in CREDBANK relies instead on an automated classifier whose performance could be improved. Disagreement may therefore be more important in CREDBANK but is masked by errors in classification.

To understand better the connection between journalists' idea of truth and users' perceptions of credibility, one could construct a single data set that covers the same events and solicits input from both journalists and regular users. A comparison between these two sources on the same data might address issues like biases towards believability or events with differing levels of impact or polarized opinion (e.g., PHEME's rumor of a football player contracting ebola versus CREDBANK's coverage of President Obama's State of the Union address in the United States).

Finally, this work does not explore the sources of disagreement in PHEME and CREDBANK. As a result, the overall contributions of this work must be tempered by the possibility that the sources of disagreement could be journalists or media

organizations. If these entities are the ones responsible for sharing disagreement and attempting to correct the accuracy of stories, then users might be better served simply by following these journalists or news organizations rather than relying on an automated system. More research into the sources of these features is needed to address this concern.

Chapter 7: Conclusions

This dissertation develops the foundations for automatic, real-time discovery of high-impact events from social media streams by first analyzing how social media responds to crises and then developing algorithms for identifying these events. Responding to the population's increasing reliance on such social media streams for breaking news and information, the capstone of this dissertation is an analysis of features the indicate veracity and credibility in these streams. Experiments presented herein rely on Twitter as a model platform for social networks.

The major contributions of this work are:

- A description of Twitter's response (and lack thereof) to terrorist attacks in Western countries,
- The introduction and evaluation of LABurst, an open-domain, language-agnostic algorithm for detecting high-impact events in social media streams,
- A real-time extension of LABurst, called RTTBurst, that uses distributed processing platforms to discover and summarize high-impact events and topics, and
- An analysis of the features that can predict credibility and veracity in Twitter

streams.

Before developing algorithms for detecting high-impact events from social media, one must first understand how social media platforms respond to these events. Chapter 3 illuminates these predictable responses in Twitter with an analysis of terrorist attacks in a Western countries. This analysis shows crisis-level events like acts of terrorism do not drive message or user volume on Twitter. Rather, crises alter Twitter's topic distribution, with content relevant to the attack experiencing significant but short-lived bursts in volume. Furthermore, retweets and hashtags also experience these significant and short bursts, with volumes of relevant tweets, retweets, and hashtags returning to pre-event levels within a few days. Frequency of users sharing links to web pages about the attacks increases more slowly and remains high several days after an attack. Finally, Twitter users tend to prefer sharing content about attacks published by local authorities or local news organizations, with these entities experiencing significant increases in followers.

After demonstrating that one cannot rely on overall Twitter dynamics to indicate major events, Chapter 4 introduces LABurst, an open-domain, language-agnostic algorithm for tracking bursts in individual keywords. Several features are presented and evaluated that capture rapid increases in token usage, with a simple exponential curve-fitting feature being the most indicative. LABurst is also compared against a pair of baselines to show token-based burst detection performs comparably with domain-specific techniques without requiring domain-specific knowledge. Chapter 4 then closes with a demonstration of LABurst's domain agnosticism

by showing how models trained on sporting events can be directly applied to earthquake detection.

Chapter 5 extends this event discovery algorithm to the real-time streaming context, thereby establishing event-related topics can be tracked and summarized as they occur. RTTBurst, the real-time extension of LABurst, uses a modern distributed processing framework to identify high-impact events in user-provided topics and performs competitively with similar real-time tracking systems, as shown by RTTBurst's high placement in an independently evaluated competition run by NIST. Post hoc analysis from NIST's TREC Microblog tasks further demonstrate that real-time burst detection models can be integrated into classical information retrieval systems to increase performance in real-time summarization tasks for social media streams.

These first few chapters establish the framework for real-time, open-domain event discovery in social media. Can information about these events be trusted though? Chapter 6 explores the features that indicate credibility in social media by analyzing a pair of Twitter-based credibility data sets: the PHEME rumor scheme, and CREDBANK. These two sets exhibit differing definitions of credibility: either as veracity or as a perception/belief of truth. Results show veracity in social media can be determined primarily through conversation length and amount of disagreement, while perceptions of credibility are more influenced by large amounts of links to other web pages, retweets, and mentions of other users. To the degree that veracity can be assessed in social media streams, this assessment can be performed with between twenty minutes and two hours of observation.

Chapters 3 and 4 establish feasibility for the event discovery task. Chapter 5 integrates these signals and features into a testable implementation, on which Chapter 6's credibility model can be applied. Together, these research areas demonstrate the detection, summarization, and credibility assessment of high-impact events from social media streams like Twitter.

RTTBurst's implementation and the credibility features described in Chapter 6 can be extended and enhanced to create an end-to-end system usable by journalists, first responders, and the general public. This system can address the currently deficient methods for extracting newsworthy information from social media while also providing a source for credible information in times of crisis. These contributions are directly usable in the relatively new fields of computational journalism and crisis informatics, which seek to improve news gathering and crisis response by leveraging new technologies and data sources like machine learning and social media.

Bibliography

[1] Christopher A Cassa, Rumi Chunara, Kenneth Mandl, and John S Brownstein. Twitter as a Sentinel in Emergency Situations : Lessons from the Boston Marathon Explosions. *PLOS Currents Disasters*, pages 1–10, 2013.

[2] Laura Petrecca. After bombings, social media informs (and misinforms), apr 2013.

[3] Edward F. Davis Iii, Alejandro A Alves, and David Alan Sklansky. Social Media and Police Leadership: Lessons From Boston. In *New Perspectives in Policing Bulletin*. Washington, DC: U.S. Department of Justice, National Institute of Justice, NCJ 244760., 2014.

[4] Alexandra Olteanu, Sarah Vieweg, and Carlos Castillo. What to Expect When the Unexpected Happens: Social Media Communications Across Crises. In *In Proc. of 18th ACM Computer Supported Cooperative Work and Social Computing (CSCW'15)*, number EPFL-CONF-203562, 2015.

[5] Sabrina Siddiqui. Boston Bombings Reveal Media Full Of Mistakes, False Reports, apr 2013.

[6] Christopher Matthews. How Does One Fake Tweet Cause a Stock Market Crash?, apr 2013.

[7] Byungkyu Kang, John O'Donovan, and Tobias Höllerer. Modeling topic specific credibility on twitter. In *Proceedings of the 2012 ACM international conference on Intelligent User Interfaces*, IUI '12, pages 179–188, New York, NY, USA, 2012. ACM.

[8] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. Information credibility on twitter. In *Proceedings of the 20th international conference on World wide web*, WWW '11, pages 675–684, New York, NY, USA, 2011. ACM.

[9] Aditi Gupta and Ponnurangam Kumaraguru. Credibility ranking of tweets during high impact events. In *Proceedings of the 1st Workshop on Privacy and Security in Online Social Media*, PSOSM '12, pages 2:2—-2:8, New York, NY, USA, 2012. ACM.

[10] Srijith Ravikumar, Raju Balakrishnan, and Subbarao Kambhampati. Ranking tweets considering trust and relevance. In *Proceedings of the Ninth International Workshop on Information Integration on the Web*, IIWeb '12, pages 4:1—-4:4, New York, NY, USA, 2012. ACM.

[11] Nicholas Diakopoulos, Munmun De Choudhury, and Mor Naaman. Finding and assessing social media information sources in the context of journalism. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 2451–2460, New York, NY, USA, 2012. ACM.

[12] Aditi Gupta and Ponnurangam Kumaraguru. Twitter explodes with activity in mumbai blasts! a lifeline or an unmonitored daemon in the lurking? 2012.

[13] Investor.fb.com. Facebook Q1 2015 Results. Technical report, Facebook, 2015.

[14] Luis Cipriani. Goal! Detecting the most important World Cup moments. Technical report, Twitter, 2014.

[15] S Milgram. The small world problem. *Psychology Today*, 2(1):60–67, 1967.

[16] Johan Bollen, Huina Mao, and Xiaojun Zeng. Twitter mood predicts the stock market. *Journal of Computational Science*, 2(1):1–8, 2011.

[17] Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. Earthquake shakes Twitter users: real-time event detection by social sensors. In *Proceedings of the 19th international conference on World wide web*, WWW '10, pages 851–860, New York, NY, USA, 2010. ACM.

[18] Michael D. Conover, Emilio Ferrara, Filippo Menczer, and Alessandro Flammini. The Digital Evolution of Occupy Wall Street. *PLoS ONE*, 8(5), 2013.

[19] Michael D Conover, Clayton Davis, Emilio Ferrara, Karissa McKelvey, Filippo Menczer, and Alessandro Flammini. The Geospatial Characteristics of a Social Movement Communication Network. *PLoS ONE*, 8(3):e55957, 2013.

[20] Cristian Vaccari, Augusto Valeriani, Pablo Barber??, Rich Bonneau, John T. Jost, Jonathan Nagler, and Joshua A. Tucker. Political expression and action on social media: Exploring the relationship between lower- and higher-threshold political activities among twitter users in Italy. *Journal of Computer-Mediated Communication*, 20(2):221–239, 2015.

[21] Saša Petrovic, Miles Osborne, Richard McCreadie, Craig Macdonald, Iadh Ounis, and Luke Shrimpton. Can Twitter replace Newswire for breaking news? In *Proceedings of the 7th International AAAI Conference on Weblogs and Social Media*, volume 2011, 2013.

[22] Jagan Sankaranarayanan, Hanan Samet, Benjamin E Teitler, Michael D Lieberman, and Jon Sperling. TwitterStand: news in tweets. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS '09, pages 42–51, New York, NY, USA, 2009. ACM.

[23] Roja Bandari, Sitaram Asur, and Bernardo a Huberman. The Pulse of News in Social Media: Forecasting Popularity. *Proceedings of the Sixth International Conference on Weblogs and Social Media - ICWSM '12*, 2012.

[24] Amy Mitchell and Dana Page. The Evolving Role of News on Twitter and Facebook. Technical report, Pew Research Center, 2015.

[25] Farida Vis. Twitter As a Reporting Tool for Breaking News. *Digital Journalism*, 1(1):27–47, 2013.

[26] Brooke Liu, Julia Daisy Fraustino, and Yan Jin. Social Media Use during Disasters: A Nationally Representative Field Experiment. Technical report, College Park, MD, 2013.

[27] Sarah Elizabeth Vieweg. *Situational Awareness in Mass Emergency : A Behavioral and Linguistic Analysis of Microblogged Communications*. PhD thesis, University of Colorado, 2012.

[28] Julia Daisy Fraustino, Brooke Liu, and Jin Yan. Social Media Use during Disasters: A review of the Knowledge Base and Gaps. pages 1–39, 2012.

[29] Amanda Lee Hughes and Leysia Palen. Twitter adoption and use in mass convergence and emergency events. *International Journal of Emergency Management*, 6(3/4):248, 2009.

[30] Bruce R Lindsay. Social Media and Disasters: Current Uses, Future Options and Policy Considerations. Technical report, 2011.

[31] Onook Oh, Manish Agrawal, and H. Raghav Rao. Information control and terrorism: Tracking the Mumbai terrorist attack through twitter. *Information Systems Frontiers*, 13(1):33–43, 2011.

[32] Rachel Sullivan. Live-tweeting terror: a rhetorical analysis of @HSMPress_-Twitter updates during the 2013 Nairobi hostage crisis. *Critical Studies on Terrorism*, 7(3):422–433, 2014.

[33] Jaeung Lee, Manish Agrawal, and H R Rao. Message diffusion through social network service: The case of rumor and non-rumor related tweets during Boston bombing 2013. *Information Systems Frontiers*, 17(5):997–1005, 2015.

[34] Onook Oh, Kyounghee Hazel Kwon, and H Raghav Rao. An Exploration of Social Media in Extreme Events: Rumor Theory and Twitter during the Haiti Earthquake 2010. In *ICIS*, number October, page 231, 2010.

[35] James Allan, Ron Papka, and Victor Lavrenko. On-line new event detection and tracking. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 37–45. ACM, 1998.

[36] Jon Kleinberg. Bursty and hierarchical structure in streams. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pages 91–101, New York, NY, USA, 2002. ACM.

[37] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[38] Gabriel Pui Cheong Fung, Jeffrey Xu Yu, Philip S Yu, and Hongjun Lu. Parameter free bursty events detection in text streams. In *Proceedings of the 31st international conference on Very large data bases*, VLDB '05, pages 181–192. VLDB Endowment, 2005.

[39] Qiankun Zhao, Prasenjit Mitra, and Bi Chen. Temporal and Information Flow Based Event Detection From Social Text Streams. In *the 22nd national conference on Artificial intelligence - Volume 2*, number Leuski 2004, pages 1501–1506, 2007.

[40] Nilesh Bansal and Nick Koudas. BlogScope: a system for online analysis of high volume text streams. In *Proceedings of the 33rd international conference on Very large data bases*, VLDB '07, pages 1410–1413. VLDB Endowment, 2007.

[41] Nilesh Bansal and Nick Koudas. BlogScope: spatio-temporal analysis of the blogosphere. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 1269–1270, New York, NY, USA, 2007. ACM.

[42] Hila Becker, Mor Naaman, and Luis Gravano. Beyond Trending Topics: Real-World Event Identification on Twitter - Technical Report. 2011.

[43] Hila Becker, Mor Naaman, and Luis Gravano. Beyond Trending Topics: Real-World Event Identification on Twitter. *ICWSM*, 11:438–441, 2011.

[44] Qiming Diao, Jing Jiang, Feida Zhu, and Ee-Peng Lim. Finding bursty topics from microblogs. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 536–544. Association for Computational Linguistics, 2012.

[45] Janette Lehmann, Bruno Gonçalves, José J Ramasco, and Ciro Cattuto. Dynamical Classes of Collective Attention in Twitter. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, pages 251–260, New York, NY, USA, 2012. ACM.

[46] Venu Vasudevan, Jehan Wickramasuriya, Siqi Zhao, and Lin Zhong. Is Twitter a good enough social sensor for sports TV? In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2013 IEEE International Conference on*, pages 181–186. IEEE, 2013.

[47] Siqi Zhao, Lin Zhong, Jehan Wickramasuriya, and Venu Vasudevan. Human as Real-Time Sensors of Social and Physical Events: A Case Study of Twitter and Sports Games. *CoRR*, abs/1106.4, 2011.

[48] James Lanagan and Alan F Smeaton. Using twitter to detect and tag important events in live sports. *Artificial Intelligence*, pages 542–545, 2011.

[49] Cindy Xide Lin, Bo Zhao, Qiaozhu Mei, and Jiawei Han. PET: a statistical model for popular events tracking in social communities. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '10, pages 929–938, New York, NY, USA, 2010. ACM.

[50] Saša Petrović, Miles Osborne, and Victor Lavrenko. Streaming First Story Detection with Application to Twitter. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 181–189, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[51] Miles Osborne, Sean Moran, Richard McCreadie, Alexander Von Lunen, Martin Sykora, Elizabeth Cano, Neil Ireson, Craig Macdonald, Iadh Ounis, Yulan He, and Others. Real-Time Detection, Tracking, and Monitoring of Automatically Discovered Events in Social Media. *Association for Computational Linguistics*, 2014.

[52] J Rogstadius, M Vukovic, C A Teixeira, V Kostakos, E Karapanos, and J A Laredo. CrisisTracker: Crowdsourced social media curation for disaster awareness. *IBM Journal of Research and Development*, 57(5):4:1–4:13, sep 2013.

[53] Jianshu Weng and Bu-Sung Lee. Event Detection in Twitter. In *ICWSM*, 2011.

[54] Wei Xie, Feida Zhu, Jing Jiang, Ee-peng Lim, and Ke Wang. TopicSketch: Real-time Bursty Topic Detection from Twitter. In *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, pages 837–846. IEEE, 2013.

[55] Laura Sydell. How Twitter's Trending Algorithm Picks Its Topics, dec 2011.

[56] B J Fogg and Hsiang Tseng. The Elements of Computer Credibility. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '99, pages 80–87, New York, NY, USA, 1999. ACM.

[57] Andrew J. Flanagin and Miriam J. Metzger. Perceptions of Internet Information Credibility. *Journalism & Mass Communication Quarterly*, 77(3):515–540, sep 2000.

[58] Princeton Survey Research Associates International. A Matter of Trust : What Users Want From Web Sites, 2002.

[59] Princeton Survey Research Associates International. Leap Of Faith: Using The Internet Despite The Dangers, 2005.

[60] Thomas J. Johnson, Barbara K. Kaye, Shannon L. Bichard, and W. Joann Wong. Every Blog Has Its Day: Politically-interested Internet Users' Perceptions of Blog Credibility. *Journal of Computer-Mediated Communication*, 13(1):100–122, oct 2007.

[61] Danyel Fisher, Marc Smith, and Howard T Welser. You Are Who You Talk To: Detecting Roles in Usenet Newsgroups. In *Proceedings of the 39th Annual Hawaii International Conference on System Sciences - Volume 03*, HICSS '06, pages 59.2—-, Washington, DC, USA, 2006. IEEE Computer Society.

[62] Andrew McCallum, Xuerui Wang, and Andrés Corrada-Emmanuel. Topic and role discovery in social networks with experiments on enron and academic email. *Journal of Artificial Intelligence Research*, 30(1):249–272, oct 2007.

[63] Howard T Welser, Eric Gleave, Danyel Fisher, and Marc Smith. Visualizing the signatures of social roles in online discussion groups. *Journal of social structure*, 8(2):1–32, 2007.

[64] Eni Mustafaraj and Panagiotis Metaxas. From obscurity to prominence in minutes: Political speech and real-time search. In *Proceedings of the WebSci10: Extending the Frontiers of Society On-Line*, Raleigh, 2010.

[65] Jacob Ratkiewicz, Michael Conover, Mark Meiss, Bruno Gonçalves, Snehal Patil, Alessandro Flammini, and Filippo Menczer. Detecting and Tracking the Spread of Astroturf Memes in Microblog Streams. *CoRR*, abs/1011.3, 2010.

[66] Marcelo Mendoza, Barbara Poblete, and Carlos Castillo. Twitter Under Crisis: Can We Trust What We RT? In *Proceedings of the First Workshop on Social Media Analytics*, SOMA '10, pages 71–79, New York, NY, USA, 2010. ACM.

[67] Vahed Qazvinian, Emily Rosengren, Dragomir R. Radev, and Qiaozhu Mei. Rumor Has It: Identifying Misinformation in Microblogs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1589–1599, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[68] Julia Schwarz and Meredith Morris. Augmenting Web Pages and Search Results to Support Credibility Assessment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, pages 1245–1254, New York, NY, USA, 2011. ACM.

[69] Meredith Ringel Morris, Scott Counts, Asta Roseway, Aaron Hoff, and Julia Schwarz. Tweeting is Believing?: Understanding Microblog Credibility Perceptions. In *Proceedings of the ACM 2012 Conference on Computer Supported Cooperative Work*, CSCW '12, pages 441–450, New York, NY, USA, 2012. ACM.

[70] Sepandar D Kamvar, Mario T Schlosser, and Hector Garcia-Molina. The Eigentrust Algorithm for Reputation Management in P2P Networks. In *Proceedings of the 12th International Conference on World Wide Web*, WWW '03, pages 640–651, New York, NY, USA, 2003. ACM.

[71] Xin Liu, Anwitaman Datta, Krzysztof Rzadca, and Ee-Peng Lim. StereoTrust: A Group Based Personalized Trust Model. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, CIKM '09, pages 7–16, New York, NY, USA, 2009. ACM.

[72] Jennifer Golbeck. Combining provenance with trust in social networks for semantic web content filtering. In *Provenance and Annotation of Data*, pages 101–108. Springer, 2006.

[73] American Red Cross. Social Media in Disasters and Emergencies. *American Red Cross Disaster Newsroom*, pages 1–19, 2010.

[74] Jeannette Sutton, Emma S Spiro, Sean Fitzhugh, Britta Johnson, Ben Gibson, and Carter T Butts. Terse Message Amplification in the Boston Bombing Response. *Proceedings of the 11th International ISCRAM Conference*, (May):610–619, 2014.

[75] Sandra González-Bailón, Ning Wang, Alejandro Rivero, Javier Borge-Holthoefer, and Yamir Moreno. Assessing the bias in samples of large online networks. *Social Networks*, 38(1):16–27, 2014.

[76] Fred Morstatter, J Pfeffer, H Liu, and Km Carley. Is the sample good enough? Comparing data from Twitter's streaming API with Twitter's firehose. *Proceedings of ICWSM*, pages 400–408, 2013.

[77] Faris Kateb and Jugal Kalita. Classifying Short Text in Social Media: Twitter as Case Study. *International Journal of Computer Applications*, 111(9):1–12, 2015.

[78] M McCord and M Chuah. Spam Detection on Twitter Using Traditional Classifiers. In *Proceedings of the 8th International Conference on Autonomic and Trusted Computing*, ATC'11, pages 175–186, Berlin, Heidelberg, 2011. Springer-Verlag.

[79] Y. Jin, Julia Daisy Fraustino, and Brooke Liu. The scared, the outraged, and the anxious: How crisis emotions, involvement, and demographics predict publics' conative coping. In *the Annual Convention of the International Communication Association*, San Juan, Puerto Rico, 2015.

[80] Danai Koutra, Paul Bennett, and Eric Horvitz. Events and Controversies: Influences of a Shocking News Event on Information Seeking. In *Proceedings of the 24th International Conference on World Wide Web Companion*, Florence, Italy, 2015. International World Wide Web Conferences Steering Committee.

[81] Meeyoung Cha, Hamed Haddadi, Fabricio Benevenuto, and Krishna P Gummadi. Measuring User Influence in Twitter: The Million Follower Fallacy. In *Fourth International AAAI Conference on Weblogs and Social Media*, may 2010.

[82] 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 conference on World wide web*, pages 591–600. ACM, 2010.

[83] Felix Richter. Twitter's Top 5 Markets Account for 50% of Active Users, 2013.

[84] Nicholas A Diakopoulos and David A Shamma. Characterizing debate performance via aggregated twitter sentiment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1195–1198. ACM, 2010.

[85] Khoo Khyou Bun and Mitsuru Ishizuka. Topic Extraction from News Archive Using TF*PDF Algorithm. In *Proceedings of the 3rd International Conference on Web Information Systems Engineering*, WISE '02, pages 73–82, Washington, DC, USA, 2002. IEEE Computer Society.

[86] Chung-hong Lee, Chih-hong Wu, and Tzan-feng Chien. BursT: a dynamic term weighting scheme for mining microblogging messages. In *Proceedings of the 8th international conference on Advances in neural networks - Volume Part III*, ISNN'11, pages 548–557, Berlin, Heidelberg, 2011. Springer-Verlag.

[87] Khoo Khyou Bun and Mitsuru Ishizuka. Emerging topic tracking system in WWW. *Knowledge-Based Systems*, 19(3):164–171, 2006.

[88] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 1995.

[89] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[90] Saša Petrović, Miles Osborne, and Victor Lavrenko. The Edinburgh Twitter Corpus. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Linguistics in a World of Social Media*, WSA '10, pages 25–26, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[91] Y Freund and Re Schapire. A desicion-theoretic generalization of on-line learning and an application to boosting. *Computational learning theory*, 55(1):119–139, 1995.

[92] Jimmy Lin, Miles Efron, Yulu Wang, Garrick Sherman, and Ellen Voorhees. Overview of the TREC-2015 Microblog Track. In *Proceedings of the Twenty-Fourth Text REtrieval Conference (TREC 2015)*, Gaithersburg, MD, 2015.

[93] Deepayan Chakrabarti and Kunal Punera. Event Summarization Using Tweets. In *ICWSM*, 2011.

[94] Luchen Tan, Adam Roegiest, and Charles L A Clarke. University of Waterloo at TREC 2015 Microblog Track. In *TREC*, 2015.

[95] Marcel Broersma and Todd Graham. Social Media As Beat. *Journalism Practice*, 6(3):403–419, 2012.

[96] Laura Petrecca. After bombings, social media informs (and misinforms), apr 2013.

[97] Tanushree Mitra and Eric Gilbert. CREDBANK: A Large-Scale Social Media Corpus With Associated Credibility Annotations. *International AAAI Conference on Web and Social Media (ICWSM)*, 2015.

[98] Arkaitz Zubiaga, Maria Liakata, Rob Procter, Kalina Bontcheva, and Peter Tolmie. Towards Detecting Rumours in Social Media. *AAAI Workshop on AI for Cities*, pages 25–26, 2015.

[99] Arkaitz Zubiaga, Maria Liakata, Rob Procter, Kalina Bontcheva, and Peter Tolmie. Crowdsourcing the Annotation of Rumourous Conversations in Social Media. *World Wide Web Conference, Florence, Italy,*, 6(1):18–22, 2015.

[100] Arkaitz Zubiaga, Geraldine Wong Sak Hoi, Maria Liakata, Rob Procter, and Peter Tolmie. Analysing How People Orient to and Spread Rumours in Social Media by Looking at Conversational Threads. *PLoS ONE*, pages 1–33, 2015.

[101] JeyHan Lau, Nigel Collier, and Timothy Baldwin. On-line Trend Analysis with Topic Models: #twitter Trends Detection Topic Model Online. *International Conference on Computational Linguistics (COLING)*, 2(December):1519–1534, 2012.

[102] Roser Sauri and James Pustejovsky. *Factbank: A corpus annotated with event factuality*, volume 43. 2009.

[103] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. Predicting information credibility in time-sensitive social media. *Internet Research*, 23(5):560–588, 2013.

[104] Arkaitz Zubiaga and Heng Ji. Tweet, but verify: epistemic study of information verification on Twitter. *Social Network Analysis and Mining*, 4(1):1–12, 2014.

[105] Kate Starbird, Jim Maddock, Mania Orand, Peg Achterman, and Robert M. Mason. Rumors, False Flags, and Digital Vigilantes: Misinformation on Twitter after the 2013 Boston Marathon Bombing. *iConference 2014 Proceedings*, pages 654–662, 2014.

[106] Tom De Smedt and Walter Daelemans. Pattern for Python. *Journal of Machine Learning Research*, 13:2063–2067, 2012.

[107] Tom De Smedt and Walter Daelemans. "Vreselijk mooi!" (terribly beautiful): A Subjectivity Lexicon for Dutch Adjectives. *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 3568–3572, 2012.

[108] Gilles Louppe, Louis Wehenkel, Antonio Sutera, and Pierre Geurts. Understanding variable importances in forests of randomized trees. *Advances in Neural Information Processing Systems 26*, pages 431–439, 2013.