EVENT MODELING IN SOCIAL MEDIA WITH APPLICATION TO DISASTER DAMAGE ASSESSMENT

A Thesis

by

YUAN LIANG

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved by:

Chair of Committee,	James Caverlee
Committee Members,	John B. Mander
	Frank Shipman
Head of Department,	Hank Walker

August 2013

Major Subject: Computer Engineering

ABSTRACT

This thesis addresses the modeling of events in social media, with an emphasis on the detection, tracking, and analysis of disaster-related events like the 2011 Tohuku Earthquake in Japan. Successful event modeling is critical for many applications including information search, entity extraction, disaster assessment, and emergency monitoring. However, modeling events in social media is challenging since: (i) social media is noisy and oftentimes incomplete, in the sense that users provide only partial evidence of their participation in an event; (ii) messages in social media are usually short, providing only little textual narrative (thereby making event detection difficult); and (iii) the size of short-lived events typically changes rapidly, growing and shrinking in sharp bursts. With these challenges in mind, this thesis proposes a framework for event modeling in social media and makes three major contributions:

- The first contribution is a signal processing-inspired approach for *event detection* from social media. Concretely, this research proposes an iterative spatial-temporal event mining algorithm for identifying and extracting topics from social media. One of the key aspects of the proposed algorithm is a signal processing-inspired approach for viewing spatial-temporal term occurrences as signals, analyzing the noise contained in the signals, and applying noise filters to improve the quality of event extraction from these signals.

- The second contribution is a new model of *population dynamics* of event-related crowds in social media as they first form, evolve, and eventually dissolve. Toward robust population modeling, a duration model is proposed to predict the time users spend in a particular crowd. And then a time-evolving population model is designed for estimating the number of people departing a crowd, which

enables the prediction of the total population remaining in a crowd.

- The third contribution of this thesis is a set of methods for *event analytics* for leveraging social media in an earthquake damage assessment scenario. Firstly, the difference between text tweets and image tweets is investigated, and then three features – tweet density, re-tweet density, and user tweeting count – are extracted to model the intensity attenuation of earthquakes. The observation that the relationship between social media activity vs. loss/damage attenuation suggests that social media following a catastrophic event can provide rapid insight into the extent of damage.

ACKNOWLEDGEMENTS

I would like to give my sincere thanks to all who have helped me during my graduate study, including my graduate advisor, my committees, my families and my friends.

In the past two years, my advisor Dr. James Caverlee has given me unlimited help to both my academic study and life. As my academic advisor, he is always passionate and inspiring, and would like to encourage me to try all kinds of ideas. And when I stumbled, he always offers me valuable advice and support so I can move forward on my work. His cheerful attitude not only helped me go through all difficulties, but also makes me always feel excited and proud of my work. He is also the wonderful life teacher giving me many important lessons including willing to encourage and help other and holding a positive attitude toward failures.

I am also thankful to my lab mates at Infolab. Senior graduate students Krishna Kamath and Zhiyuan Cheng gave me tremendous help when I first joined the lab. Thanks to them for sharing their data and experiences with me, discussing about my research work and revising papers for me. Their have helped me improve both my research and communication abilities. I am also thankful to Elham Khabiri, Kyumin Lee, McGee, who solved a lot of confusions for me in my studies and life in the U.S. Their patience helped me get used to this new life quickly. Also, I want to thank my other lab mates, Amir Fayazi, Vandana Bachani, Wei Niu, Cheng Cao and Haokai Lu, for bringing joys and happiness to my life in the past two years.

I wish to thank my husband for his love and support. He has always been massively supportive to me, and brought me lots of courage and happiness. Because of his help, I can adjust myself to this new study and living environment quickly.

Also many thanks to my parents and parents-in-law, they are so supportive to my every decision, even sometimes they disagreed with me, they gave me enough room to make up my own mind. Their understanding and support are the main reasons I can successfully finish my study today. Love them all.

TABLE OF CONTENTS

Page

ABSTRACT . . . ii

ACKNOWLEDGEMENTS . . . iv

TABLE OF CONTENTS . . . vi

LIST OF FIGURES . . . viii

LIST OF TABLES . . . x

1. INTRODUCTION . . . 1

 1.1 Motivation . . . 1
 1.2 Contributions . . . 3
  1.2.1 Event Detection — Spatial-temporal Event Mining . . . 4
  1.2.2 Event Tracking — Population Modeling . . . 4
  1.2.3 Event Analytics — Disaster Damage Assessment . . . 5
 1.3 Thesis Overview . . . 5

2. EVENT DETECTION : SPATIAL-TEMPORAL EVENT MINING . . . 7

 2.1 Introduction . . . 7
 2.2 Related Work . . . 9
 2.3 Problem Statement . . . 11
  2.3.1 Chanllenges . . . 12
 2.4 A Noise-Filtering Approach . . . 14
  2.4.1 Filtering Random Noise . . . 14
  2.4.2 Filtering Event Noise . . . 16
  2.4.3 Iterative Event Extraction Method . . . 19
 2.5 Experiments . . . 21
  2.5.1 Data Collection . . . 22
  2.5.2 Parameter Setup . . . 24
  2.5.3 Event Identification . . . 26
  2.5.4 Event Retrieval . . . 32
 2.6 Summary . . . 37

3. EVENT TRACKING : POPULATION MODELING . . . 39

 3.1 Introduction . . . 39
 3.2 Preliminaries . . . 40
  3.2.1 Challenges to Population Modeling . . . 40

3.2.2 Data Collection 41
3.3 Crowd-Based Population Model 43
3.3.1 Estimating “Immigration” 44
3.3.2 Modeling Duration to Estimate “Emigration” 45
3.3.3 Building the Population Model 50
3.3.4 Emission-Based Modeling 51
3.4 Experiments 53
3.4.1 Is Duration Informative? 54
3.4.2 Traffic Prediction with Population Model 57
3.4.3 User and Post Prediction with Population Model 61
3.5 Summary 66

4. EVENT ANALYTICS : DISASTER DAMAGE ASSESSMENT 67

4.1 Introduction 67
4.2 Related Work 67
4.3 Data Collection 68
4.4 Approach and Findings 69
4.4.1 Epicenter Estimation 70
4.4.2 Intensity Attenuation 73
4.4.3 Spread Speed 75
4.5 Summary 77

5. SUMMARY AND FUTURE RESEARCH OPPORTUNITIES 78

5.1 Summary 78
5.2 Future Research Opportunities 79

REFERENCES 82

LIST OF FIGURES

FIGURE Page

1.1 Thesis Contributions 3

2.1 Noise in Term Temporal Signals 15

2.2 Structure of Proposed Method 18

2.3 Filtered Temporal Signals 27

2.4 Average Purity of the Methods Using Random and Event Noise Filters 31

2.5 Comparison with Baseline Methods 32

2.6 Event Retrieval Evaluation 34

3.1 Examples of Check-in Patterns 45

3.2 Example of Duration Distributions for Location-Driven Crowds . . . 47

3.3 Examples of Duration Distributions for Event-Driven Crowds 48

3.4 Duration Cumulative Distribution Fitness 49

3.5 Comparison of Duration Distributions 55

3.6 Venue Classification 56

3.7 Traffic Prediction of Bridges of Manhattan 59

3.8 NDCG for The Rush Hours Ranked by the Estimated Number of Checkouts 60

3.9 Estimating Check-outs for Events 62

3.10 Estimating Event-Related Posts 63

4.1 Location Estimation Using Different Features 71

4.2 The Densities Versus the Radius in JPEQ 74

4.3 The Spread Speed of Tweets in JPEQ 76

5.1 Connection between Social Media Models with Architecture Models. 81

LIST OF TABLES

TABLE Page

2.1 Event Dataset . . . 22

2.2 Purity Results in Temporal Domain . . . 27

2.3 Purity in Spatial Domain . . . 29

2.4 Purity in Spatial-temporal Domain . . . 30

2.5 Average Purity Comparison . . . 31

2.6 The Number of Detected Clusters . . . 35

2.7 T-test Results . . . 35

2.8 Extracted Events . . . 36

3.1 Event-driven Crowd Dataset . . . 43

3.2 Estimating Checkout User Count . . . 64

3.3 Estimating Post Count . . . 65

4.1 Earthquake Data Sets . . . 69

4.2 The Euclidean Distance Between Estimated Epicenter and Actual Epicenter (Degree) . . . 72

1. INTRODUCTION

1.1 Motivation

Social media has recently become an omni-present platform for broadcasting news and sharing information, enabling users to voluntarily report on events they have experienced. Examples include short-text Twitter posts and photographs posted to Facebook in response to political debates, earthquakes, concerts, and other real-world events. With this growing user-driven evidence, new efforts arise in the large-scale mining and tracking of events derived from social media, e.g., [1, 2, 3], leading to new services that support intelligent emergency monitoring, finding nearby activities (e.g., concerts, rallies), and improving access to online content.

These social media "footprints" when coupled with extremely granular spatio-temporal information (e.g., timestamps and GPS-style geocodes), offers the tantalizing promise of a minute-by-minute and region-by-region account of a real-world event as it unfolds. Indeed, recent work has illustrated this promise through automated methods to aggregate Twitter posts for detecting the epicenter and trajectory of an earthquake [4], for detecting earthquakes and building a predictive system to notify people at-risk [5], and for constructing "theme cycles" from geo-located blog posts for assessing the public's response to Hurricane Katrina [6].

This thesis addresses the modeling of events in social media, with an emphasis on three critical factors for successful event modeling: (i) event detection; (ii) event tracking; and (iii) event analytics.

- *Event Detection* refers to the discovery of a specific activity that happens at a certain time and in a certain place. While there has been a long history of *event extraction* from traditional media like news articles, e.g., [7, 8], the growth of user-contributed and often *on-the-ground* reaction by regular social media users provides new opportunities to identify events as they arise. For example, in one exciting direction, researchers have shown how Twitter users may be treated as social sensors [9], the signal (in this case, a tweet) posted by the sensors can be used to estimate when and where an earthquake or a tornado happens. Similar efforts have demonstrated the potential of social media event extraction, for example [10], which uses user-contributed tags from the Flickr image sharing site to extract events based on temporal and geographical features.

- *Event Tracking* identifies how users behave in or respond to a certain event over time. Examples include tracking the mood changes of users over time, determining the number of users posting about an event in different periods, detecting the evolving topics users focused on in this event, and so on. Tracking event-related crowds is important for many domains. For example, companies and investors could adjust their marketing strategy, and allocate their limited resources based on the population of users drawing attention on their products; and political groups could estimate the percentage of people voting for or protesting against a new policy, with the help of population modeling for crowds.

- *Event Analytics* refers to the application-specific analysis of events that have been detected and tracked in social media. This thesis explores event analytics in the context of earthquake damage assessment. Damage assessment with social media is an important step for providing responders with rapid insight into the extent of damage to be expected in the field and the locations of greatest damage, which are both necessary for deciding how to best deploy the limited emergency response and recovery resources during the initial moments

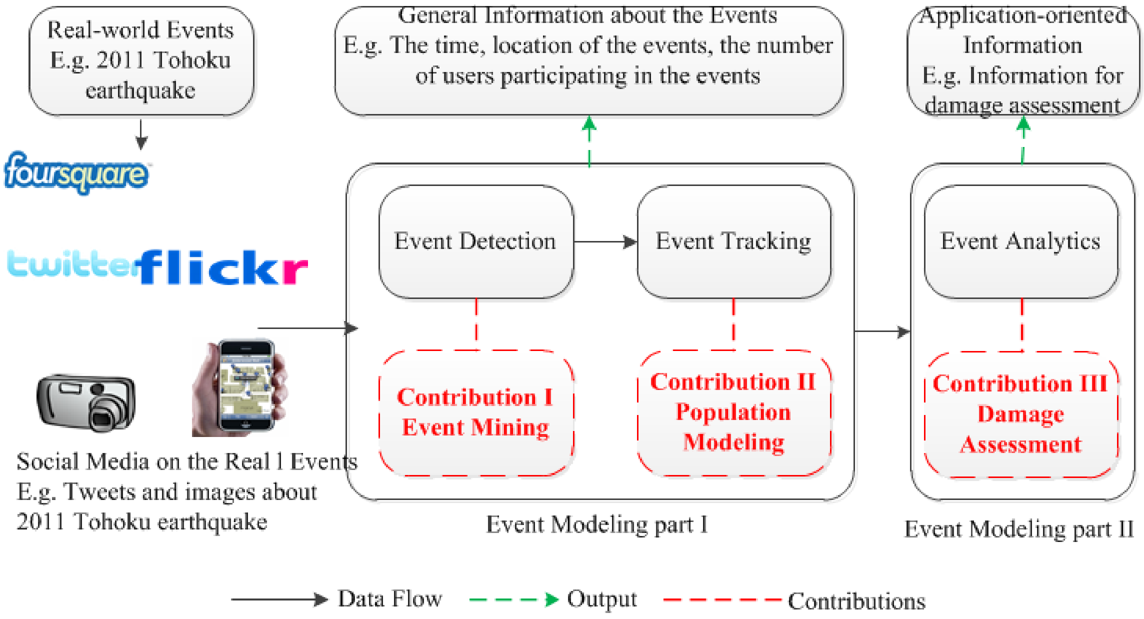

Real-world Events
E.g. 2011 Tohoku earthquake
foursquare
twitter
flickr
Social Media on the Real l Events
E.g. Tweets and images about 2011 Tohoku earthquake
General Information about the Events
E.g. The time, location of the events, the number of users participating in the events
Event Detection
Event Tracking
Contribution I Event Mining
Contribution II Population Modeling
Event Modeling part I
Application-oriented Information
E.g. Information for damage assessment
Event Analytics
Contribution III Damage Assessment
Event Modeling part II
Data Flow
Output
Contributions

Figure 1.1: Thesis Contributions

of an earthquake.

1.2 Contributions

This thesis proposes a framework for event modeling, which is compromised of event detection, event tracking and event analytics components (as shown in Figure 1.1). Take the 2011 Tohoku earthquake as an example. When the disaster happened, there are tens of thousands of text tweets and images posted on-line describing this event. From these messages, the event detection component extracts related contents, event tracking component tracks the number of affected people, and event analytics component mines the application-oriented information such as epicenter and damage degree. Together, these general and application-oriented information are collected and provided to the damage assessment application.

Concretely, this thesis makes three contributions:

1.2.1 Event Detection — Spatial-temporal Event Mining

The first contribution is a signal processing-inspired approach for *event detection* from social media. While existing event detection work like [11, 12] have provided many effective methods over long-form documents like news articles, which typically provide a rich source of context for event detection. Social media content, in contrast, often provides only a short description, title, or tags, (and thereby little textual narrative) limiting the effectiveness of semantic similarity based event detection techniques, and thus adds new challenges on event detection.

In this work, an iterative spatial-temporal event mining algorithm is designed for identifying and extracting topics from social media. One of the key aspects of the proposed algorithm is a signal processing-inspired approach for viewing spatial-temporal term occurrences as signals, analyzing the noise contained in the signals, and applying noise filters to improve the quality of event extraction from these signals. The iterative event mining algorithm alternately clusters terms and then generates new filters based on the results of clustering. Multiple filters are explored, including Gaussian band-pass filters, Ideal band-pass filters, and others. The proposed approach is evaluated through experiments on several event data sets; the results indicate that the proposed method can effectively remove event noise, improving event mining effectiveness from social media.

1.2.2 Event Tracking — Population Modeling

Second, this thesis models the *population dynamics* of event-related crowds as they first form, evolve, and eventually dissolve. Crowd modeling is challenging since 1) user-contributed data in social media is noisy and oftentimes incomplete, in the sense that users only reveal when they join a crowd through posts but not when they depart; and 2) the size of short-lived crowds typically changes rapidly, growing and shrinking in sharp bursts. Toward robust population modeling, a duration model is proposed to predict the time users spend in a particular crowd. And then a time-evolving population model is designed for estimating the number of people departing a crowd, which enables the prediction of the total population remaining in a crowd. At last, the crowd models is validated through extensive experiments over 22 million geo-location based check-ins and 120,000 event-related tweets.

1.2.3 Event Analytics — Disaster Damage Assessment

Finally, this thesis designs a set of methods for leveraging social media for the earthquake damage assessment application. The potential of social media is investigated to provide rapid insights into the location and extent of damage associated with two recent earthquakes – the 2011 Tohoku earthquake in Japan and the 2011 Christchurch earthquake in New Zealand. Firstly, the difference between text tweets and media tweets is investigated (containing links to images and videos), and then three features – tweet density, re-tweet density, and user tweeting count – are extracted to model the intensity attenuation of each earthquake. The observation that the relationship between social media activity vs. loss/damage attenuation suggests that social media following a catastrophic event can provide rapid insight into the extent of damage.

1.3 Thesis Overview

The remainder of this thesis is organized into four parts, of which the first three are about the contributions and the fourth is for conclusions. The outline is:

Section 2 — Event Detection : Spatial-temporal Event Mining — designs a new signal-inspiring method to detect events from short-text posts from social media.

Section 3 — Event Tracking : Population Modeling — proposes a population model based on duration to estimate the size of the on-line crowds.

Section 4 — Event Analytics : Disaster Damage Assessment — discusses the capacity of social media for providing quick assessment for earthquakes.

Section 5 — Summary and Future Research Opportunities — concludes with a summary of the thesis contributions and a discussion of future research.

2. EVENT DETECTION : SPATIAL-TEMPORAL EVENT MINING

This section describes a new spatial-temporal features based method to detect events from short-text social media. An event is defined as a specific activity that happens in a specific time and place [13]. Therefore, given a group of terms, if it represents an event, the group of terms should satisfy three constraints: 1) the terms are semantically consistent, 2) the terms should happen in the same time period, and 3) the terms should appear in similar locations. Hence, event detection can be defined as: given a set of terms S, to detect subsets from S so that each subset $S_k \in S$ is a set of terms satisfying the constraints.

2.1 Introduction

In general, existing event detection methods can be categorized into two types: *document-pivot* approaches and *feature-pivot* approaches [13]. Document-pivot approaches identify events by clustering documents (e.g., news articles) based on semantic similarities, and then treating each cluster as an event. A series of works like [11, 12] have shown the effectiveness of this method over long-form documents like news articles, which typically provide a rich source of context for event detection. Social media content, in contrast, often provides only a short description, title, or tags, (and thereby little textual narrative) limiting the effectiveness of semantic similarity based event detection techniques. As a result, many social media event detection algorithms have relied on *feature-pivot approaches*, which group similar event-related terms, for example by finding terms with a similar temporal distribution or spatial footprint. In this way, event-related terms may be clustered together based on these common signals (treating each term as a frequency function over either time or space). These feature-pivot approaches, e.g., [10, 13], have shown the potential of this approach for scaling to event detection over user-contributed social media posts.

While encouraging, these feature-pivot based approaches may be susceptible to *noise* in both the temporal and spatial signals they use, which can hinder the quality of event detection. Specifically, three potential sources of noise are identified:

- *Background-topic noise* is noise introduced by topics unrelated to an event. For example, the term "apple" has a periodic and strong background signal (as illustrated in Figure 2.1(a)) which may be unrelated to a specific event like Apple introducing a new iPad. This background noise may obscure the "apple" signal for the iPad introduction, leading to poor event detection.

- *Multi-event noise* is noise introduced by events related to an event of interest. For example, an event of interest like a tornado in Texas may occur at the same time as multiple tornados striking in different regions of the country, all triggering tornado-related social media posts. These other tornado posts potentially introduce noise for detecting the event of interest.

- *Random noise.* The final source of noise is Random noise, which can be introduced through the sparsity of data, in-correct timestamps or locations, mislabeled geo-coordinates, term extraction error, and so on.

Each of these sources of noise may impact the quality of the spatial and temporal term-based signals, leading to poor event detection. Hence, the thesis explores a new approach for event detection from social media explicitly designed to target these sources of noise.

Concretely, this section proposes a *signal-processing* inspired event detection framework for social media. This signal processing-inspired approach views spatial-temporal term occurrences as signals, analyzes the noise contained in the signals, and applies noise filters to improve the quality of event extraction from these signals. The proposed approach incorporates this noise-filtering approach into an iterative spatial-temporal event mining algorithm for identifying and extracting events from social media. The iterative event mining algorithm alternately clusters terms using their filtered signals, and then generates new filters based on the results of clustering. Multiple filters are explored – including Gaussian band-pass filters, Ideal band-pass filters, and others – in the comprehensive experimental setting. Over two tasks – Event Identification and Event Retrieval based over two Twitter-based datasets – it is found that the noise filtering based approach results in improved clustering results over several baseline methods. The results also show that the proposed method increases by around 7% than baseline methods regarding the clustering purity in the Event Identification, and improves about 2% to 4% in the Event Retrieval based on the metrics of 2nd Moment and Entropy.

2.2 Related Work

Event detection refers to the discovery of a specific activity that happens at a certain time and in a certain place. Event detection is typically categorized into two types: retrospective detection and on-line detection [8]. The former is to detect events from collected historical documents [14, 15], and the latter tries to extract events from real-time documents [7]. This thesis focuses on retrospective detection methods.

Early retrospective detection approaches usually adopt clustering methods based on semantic similarities of documents, e.g., [7] uses a modified version of TF/IDF to measure the distance of documents, and cluster documents based on the estimated distance. [8] uses a similar technique plus a time window and a decay factor for the similarity measurement between documents. Many other efforts focus on detecting events from terms, focusing on measuring the semantic similarities of terms. [16] and [12] use co-occurrences to measure the closeness of tags for landmark detection and tag recommendation. [17] builds a tag graph using co-occurrence, but considers multiple hops in the graph to compute the distance of tags.

Recently, many new features have been introduced for event detection, like temporal-pattern, geo-location, user calendars, clicks and queries. For example, the work in [18] detects events from click-through web data by considering each event as a set of query-page pairs. [19] examines features first by using Discrete Fourier Transformation (DFT), and classifies features into important and un-important categories for event extraction. [20] explores domains for events, discovers important event categories and classifies extracted events based on latent variable models. [1] applie multiple features including tag, time stamp and authors, and proposed a weighting method to combine them for event extraction.

Temporal and spatial patterns also have been studied to discover relationships including terms, posts in social network, and news articles. [21] utilizes the time information to determine a set of bursty features which may occur in different time windows, then it detects bursty events based on the feature distributions. [22, 23, 2] use geo-location information integrated with a statistical model to detect the geographical topics. [13] observes the spatial-temporal patterns for tags, and adopts a wavelet transform-based method to find tags with significant peaks in spatial-temporal distribution, and then cluster these tags using DBSCAN to extract events. [24] detects event by finding tags with bursts in temporal and spatial patterns. [10] compare the 3D spatial-temporal distributions between terms to measure the closeness of different terms, and cluster the terms based on the distances to extract events.

Building on this related body of literature, the work presented here focuses on the challenge of noise in the spatial-temporal term signals for event detection.

2.3 Problem Statement

Given a collection of user-contributed social media documents $D = \{d_1, d_2, ..., d_T\}$, each document d_i can be presented as $\langle W, t, l \rangle$, where W is a list of terms from vocabulary V, t is the published time of d_i, and $l = (la, lo)$ is the associated geo-location, consisting of latitude and longitude coordinates. It is assumed that there are K events $\theta = \{\theta_1, ..., \theta_K\}$ hidden in D and each document belongs to one of these events. The goal is to detect these K hidden events from the observed documents.

Event: As described before, an event refers to a specific activity that happens in a specific time and place [13]. Is defined with a group of terms satisfying three constraints: 1) semantically consistent, 2) happen in the same time period, and 3) appear in similar locations. Given a set of terms S, event detection is the process of detecting subsets from S so that each subset $S_k \in S$ is a set of terms satisfying the constraints.

Term Signals: Term signals refer to the sequence of numbers representing the bucketed occurrence counts in the temporal, spatial or spatial-temporal domains (usually normalized). They can also be regarded as samples of temporal, spatial or spatial-temporal distributions of the term. Given a term $w_i \in V$, $D_i = \{d_{i,t_1}, d_{i,t_2}, ..., d_{i,t_M}\}$ is used to denote the set of documents contains w_i. A time sequence $\{t_1, t_2, ..., t_M\}$ can be got from D_i. Bucketing them into bins gives a temporal sequence of occurrence counts for w_i: $F_{t,w_i} = \{f_{i,1}, f_{i,2}, ..., f_{i,T}\}$, where T is the maximum time bin index. F_{t,w_i} (*temporal term signal*) is the one dimensional term signal in the temporal domain. Similarly, given the geo-locations for w_i, the two dimensional signal F_{l,w_i} (*spatial term signal*) can be derived by bucketing the location sequences of D_i into a $P * Q$ grid. Given both the temporal and spatial sequences, a three dimensional signal F_{t,l,w_i} (*spatial-temporal term signal*) can be derived by gridding spatial-temporal stamps for w_i into a $T * P * Q$ space.

Event Signal: Event signals for θ_k are the aggregation of the signals of terms belong to θ_k. They can be regarded as samples of temporal, spatial or spatial-temporal distributions of the event. Given a set of terms S_k for event θ_k, the event signals can be computed using:

$$F_{t,l,\theta_k} = \sum_{E(w_i)=\theta_k} F_{t,l,w_i} \lambda_{w_i,\theta_k} \tag{2.1}$$

where $E(w_i)$ refers to the corresponding event of w_i and $\lambda_{w_i,\theta}$ is the weight of w_i.

2.3.1 Chanllenges

Given a set of terms S and their spatial-temporal signals, to detect events from S or to find the subsets of S, the key question of the method is how to estimate the distances between each pair of terms. In [23], the authors have shown how terms associated with an event usually share a similar temporal-spatial pattern; in a related direction, [10] finds that the spatial-temporal based distance can be used to measure the closeness of terms.

However, when measuring the similarities between terms by comparing the patterns of their spatial-temporal signals, one problem is that these signals can be easily polluted by *noise*. This noise may obscure the actual spatial-temporal pattern for terms and thus affect the similarity measurement. An ideal way is if for a term w_i associated with event θ_k $((E(w_i) = \theta_k))$, the method can filter all the noise contained in the signals of w_i and extract only the signals belonging to event θ_k. So if the signals belonging to the event θ_i are treated as the signals of the Region-of-Interest (ROI) and other non-relevant signals are regarded as noise, the goal of proposed method is to remove or reduce the noise and estimate the (ideally) noise-less Region-of-Interest signals for terms.

To begin the effort, three common types of noise that may impact event detection in social media are identified:

- *Background-topic noise* refers to the signals caused by the daily topics like eating, shopping, traveling, and so on. The background noise varies among different words. For example, in Figure 2.1(a), the term 'apple' is a common word which has a periodical and strong background signal. Consider the event "Steve Jobs resigned as Apple CEO". Since 'steve jobs' is not as common as 'apple', its background noise power is much lower than that of 'apple'. Therefore when comparing the closeness of them based on their term signals, the background noise will push the two signals apart.

- *Multi-event noise* refers to the burst signal caused by other un-related events. Given a word w_i, it may belong to multiple events, so its spatial-temporal signals are actually the combination of the signals belonging to multiple events: $F_{t,l,w_i} = \sum_k F_{t,l,w_i,\theta_k}$. For example, in Figure 2.1(b), two different tornados happened around the 24th and 27th of August. To capture the signals belonging to a tornado on the 24th of August, the noise associated with the tornado on the 27th should be filtered.

- *Random noise* refers to the random signals introduced by the sparsity of data, in-correct timestamps or locations, mislabeled geo-coordinates, term extraction error, etc. For example, when the terms are extracted from the text of the documents, it might introduce noise due to typos, abbreviations (very common in social media), and so on.

Since Background-topic can be treated as a special event with periodical bursts (Figure 2.1(a)), so the Background-topic noise and Multi-event noise are put into one category – Event noise.

2.4 A Noise-Filtering Approach

The section proposes a *signal-processing* inspired event detection framework for social media that views spatial-temporal term occurrences as signals, analyzes the noise contained in the signals, and applies noise filters to improve the quality of event extraction from these signals. Additionally, this noise-filtering approach is incorporated into an iterative spatial-temporal event mining algorithm for identifying and extracting events from social media. The iterative event mining algorithm alternately clusters terms using their filtered signals, and then generates new filters based on the results of clustering.

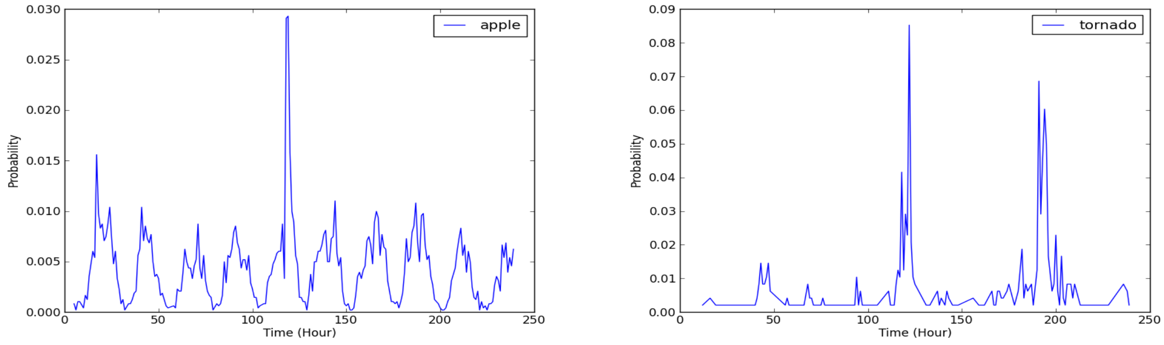
To begin, based on the above analysis, term signals F_{t,l,w_i} for w_i are viewed to be comprised of three components: (i) Random noise F_{t,l,w_i,θ_r}; (ii) the Region-of-Interest signals F_{t,l,w_i,θ_e} belong to a specific event; and (iii) Event noises $F_{t,l,w_i,\theta_{S-e}}$, where S is the set of all the events.

$$F_{t,l,w_i} = F_{t,l,w_i,\theta_e} + F_{t,l,w_i,\theta_{S-e}} + F_{t,l,w_i,\theta_r} \tag{2.2}$$

Toward the goal to better measure the similarities between words, a noise filter-based approach is proposed for estimating the F_{t,l,w_i,θ_e} from polluted signals. The approach contains two types of filters: 1) the first filter aims to reduce Random noise by smoothing the signals; 2) the second filter is based on a band-pass filter which aims at keeping only Region-of-Interest signals, and removing background-topic and multi-event noise $F_{t,l,w_i,\theta_{S-e}}$.

2.4.1 Filtering Random Noise

The first filter adopted in this work is used for reducing Random noise from the term signals (which, recall is a key step toward event detection). In speech and



(a) Temporal Distribution of Term “apple”

(b) Temporal Distribution of Term “Tornado”

Figure 2.1: Noise in Term Temporal Signals

image processing, the *mean filter* is an effective way to smooth the signal and reduce un-correlated Random noise [25]. It is assumed that the Random noise contained in the term signals are un-correlated, and therefore the method can directly apply the mean filter to the signals. The key point of a mean filter is using the neighbors to average the signal values. For every point in the signals, the value is smoothed with the Equation 2.3.

$$F^{'}_{t,l,w_i} = \sum_{t^{'}\in N(t), l^{'}\in N(l)} F_{t,l,w_i} Q(t^{'}, l^{'}) \tag{2.3}$$

For mean filter, $Q(t^{'}, l^{'})$ is set with $1/M$, where M is the number of neighbors, $N(t)$ refers to the set of neighbor points of t. A neighbor here is the point with adjacent time unit to t and closed location to $l = (la, lo)$. For example, if the boundary is defined as $N(t) = [t - 2, t + 2]$ and $N(l) = [l - 2, l + 2]$, then all the points which locate in the cubic owning length=2 and centered at (t, la, lo) are regarded as the neighbors of the unit of (t, l).

2.4.2 Filtering Event Noise

While the first filter may reduce Random noise, there is another challenge — filtering the Event noise. Toward reducing the impact of Background-topic and Multi-event noises, the method considers a *band-pass filter*. The idea of a band-pass filter is to pass the signals in a Region-of-Interest, but filter or reduce the signals in other regions. The key issues are how to find the Region-of-Interest for a particular event, and how to estimate the band-pass filter $Q(t, l|\theta_k)$ based on the detected Region-of-Interest. Once the filter $Q(t, l|\theta_k)$ is estimated, the F_{t,l,w_i} and $Q(t, l|\theta_k)$ can be used to retrieve the signals belonging to θ_k with the Equation 2.4.

$$F_{t,l,w_i,\theta_k} = F_{t,l,w_i} Q(t, l|\theta_k) \tag{2.4}$$

where $Q(t, l|\theta_k)$ is the band-pass filter for θ_k in the spatial-temporal domain.

To detect the Region-of-Interest for a certain event θ_k, it is proposed to aggregate all the signals of the terms belonging to event θ_k, and then label the region which contains the strongest signals as the Region-of-Interest. The idea behind this method is to use the neighbors to filter un-correlated noises and strengthen the signals belonging to θ_k. In signal processing, mean filtering is used to sum multiple polluted signals. For example, if $s_1, s_2, ..., s_K$ is K different samples of the signal s polluted by noises, then the mean filter uses $\lambda_1 s_1 + \lambda_2 s_2 + ... + \lambda_K s_K$, $(\lambda_1 + \lambda_2 + ... + \lambda_K = 1)$ to approach the un-polluted signal s. If the noise and signal are un-correlated, then with increasing K, the strength of the noise will be reduced to $1/\sqrt{K}$ [25]. Here, since individual terms can be polluted by some event noises which are usually un-correlated, by averaging the signals of term w_i with the signals of its neighbors, the noise introduced by different events will be reduced.

Unlike the neighbors in Section 2.4.1, which are found based on the adjacent time unit or spatial grid, the neighbors here refer to the terms belonging to the same event. However, since the method lacks access to a ground truth of which terms belong to which events, first it uses a clustering method to find the neighbors for term w_i, then the signals belonging to the same cluster are averaged using Equation 2.1 to arrive at the the estimated event signals. Regarding the clustering method, k-means is adopted in this thesis if the number of actual clusters is already known, and Affinity Propagation is used if it is unknown.[1] Next, based on the estimated event signals, the band-pass filter for events is further built up.

In particular, several different band-pass filters are considered to explore their appropriateness for event detection from social media:

Gaussian Band-pass Filter: In the Gaussian filter, it assumes that $Q(t, l|\theta_k)$ for θ_k can be represented as a single Gaussian. Then the event signals F_{t,l,θ_k} is used to train the parameters of $Q(t, l|\theta_k)$.

$$Q(t, l|\theta_k) = \frac{1}{\sigma\sqrt{2\pi}} exp\{-\frac{(x-\mu)^2}{2\sigma^2}\} \tag{2.5}$$

where x is the vector of $\langle t, l \rangle$.

Ideal Band-pass Filter: In the Ideal filter, it assumes in a cubic region of the filter (the center is the point with the strongest signal), each point has an identified weight, which is much larger than the other points outside the region.

$$Q(t, l|\theta_k) = \begin{cases} \frac{\lambda}{r} & x \in [x_l, x_r] \\ \eta\frac{1-\lambda}{R-r} & else \end{cases} \tag{2.6}$$

[1] Affinity Propagation identifies the high-quality set of exemplars among the data points and forms corresponding clusters of points around these exemplars, through exchanging messages between the points [26].

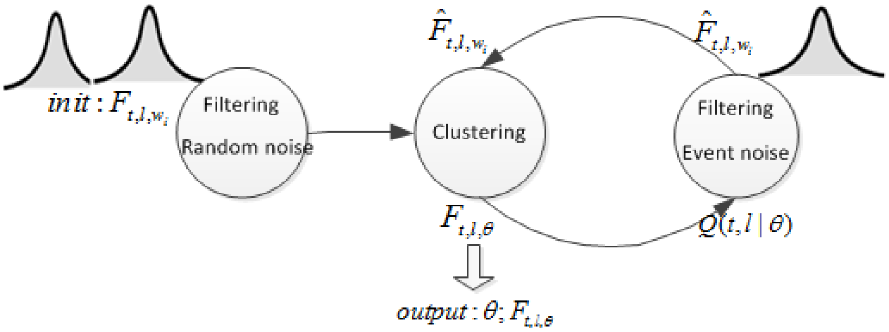


Figure 2.2: Structure of Proposed Method

where λ is the cumulative frequency probability of the region $[x_l, x_r]$, x_l and x_r are the left-up and right-down coordinators respectively. r is the area of the region, R is the whole area of the boundary, and η is the penalty factor (set 0.1 in this work).

Average Band-pass Filter: In the Average filter, $Q(t, l|\theta_k)$ is assigned with event signals directly. And for Average Filter, the λ_{w_i,θ_k} in Equation 2.1 is set with $1/N$, N is the number of terms belonging to θ_k.

Weighted Average Band-pass Filter: Similar to the Average Filter, the Weighted Average filter $Q(t, l|\theta_k)$ is also assigned with F_{t,l,θ_k} in Equation 2.1, but the λ_{w_i,θ_k} is assigned unevenly in the Weighted Average filter. The Equation for Weighted Average filter is modified from Equation 2.1 to Equation 2.7.

$$\hat{F}_{t,l,w_i} = \lambda F_{t,l,w_i} + \frac{(1-\lambda)}{N_\theta} \sum_{E(w')=\theta and w' \neq w_i} F_{t,l,w'} \tag{2.7}$$

where λ is the weight for the original signals, N_θ is the size of cluster θ.

Experimentally, these different band-pass filters are compared to evaluate their effectiveness at removing event-based noise.

2.4.3 Iterative Event Extraction Method

Based on the noise filters discussed above, the thesis proposes a new method to detect events from a set of terms. The framework of the proposed method is shown in Figure 2.2, which is mainly compromised of two parts: clustering and filtering.

The clustering part, implemented in the *clustering component*, is used to cluster terms using the distances computed with the filtered signals, and to generate the new noise filters with the clustering results. For distances based on spatial-temporal signals, the Manhattan distance is used, and different distances like temporal distance, spatial distance are tried. More details about the distance measuring are given in Section 2.5.2. For clustering, K-means (if the number of events is known) or Affinity Propagation (if the number of events is un-known) is adopted to cluster the terms. Then based on the clustering results, the event signal $F_{t,l,\theta}$ is computed with Equation 2.1 and passed to the filtering component.

The filtering part is comprised of two components: *Random noise filtering component* which is used to filter the Random noise in the initial signals, then passes the filtered signals to the clustering components. *Event noise filtering component* that is used to filter the Event noise. It generates the filter $Q(t, l, |\theta)$ based on the the $F_{t,l,\theta}$ from clustering component, and then applies $Q(t, l|\theta)$ to the F_{t,l,w_i} to generate the new $F_{t,l,w_i,\theta}$ for w_i using Equation 2.4.

An iterative method is used to integrate the clustering and Event noise filtering components. The details of proposed method are shown in the Algorithm 1. First the mean filter is used to reduce the Random noise, and pass the filtered signals to the clustering component. For the initial clustering, the co-occurrence based distance is used to cluster the terms, then the band-pass filters is generated based on the clustering results using the method in Section 2.4.2, next the signals for the

Algorithm 1 Noise Filter Based Event Detection

```
Input: $termSig$, $initC$
Init: $C \longleftarrow initC$; $isChanging \longleftarrow$ True; $iter \longleftarrow 0$
while $isChanging$ and $iter$ ¡ MaxIter do
  Clear($eventSig$, $filter$)
  for each cluster $c$ in $C$ do
    for each term $w$ in $c$ do
      $eventSig[c] \longleftarrow eventSig[c] + termSig[w]$
    end for
  end for
  for each cluster $c$ in $C$ do
    $filter[c] \longleftarrow$ GenBandFilter($eventSig[c]$)
  end for
  for each term $w$ in $c$ do
    $termSig[w] \longleftarrow$ GenNewSig($termSig[w]$, $filter[c]$)
  end for
  for each pair of terms $w_i$ and $w_j$ do
    $termDis[w_i, w_j] \longleftarrow$ Dis($termSig[w_i]$, $termSig[w_j]$)
  end for
  $newC \longleftarrow$ Clustering($termDis$)
  $isChanging \longleftarrow$ TestChanging($C$, $newC$)
  $iter \longleftarrow iter + 1$
  $C \longleftarrow newC$
end while
return  $C$, $termSig$, $termSignal[w]$, $filter$
```

terms are filtered using estimated band-pass filters and pass the filtered signals to the clustering component. The clustering component re-estimates the distances between terms based on the filtered signals and re-clusters the term using estimated distances. The clustering and Event noise filtering iteratively proceed until the clusters of terms do not change anymore or the iteration count reaches the threshold. At last, the clustering results are outputted as the detected events.

In Algorithm 1, the input term signals $TermSig$ is the signal filtered by Mean filter. The cluster C is initialized with the clustering results using Co-occurrence based distance. With the clusters C, firstly event signals $eventSig$ are computed for each cluster, then band-pass filters $filter$ are generated with $eventSig$. Next the new term signals are generated by applying $filter$ to the $TermSig$, and distances between terms $termDis$ are estimated with the new term signals $TermSig$. Then the terms are re-clustered based on the estimated distances. This process repeats until the clusters do not change or reach the max iteration count $maxIter$.

2.5 Experiments

To evaluate the effectiveness of the proposed filter based method for event extraction, two sets of experiments are designed: Event Identification (EI) and Event Retrieval (ER). Event Identification refers to identifying the clusters from pre-labeled terms whose hidden topics are already known. Event Retrieval is to retrieve clusters from selected terms without knowing the hidden topics. Two data sets are collected for the experiments: Event Identification data set which contains 4 sub sets of manually labeled terms and corresponding tweets; Event Retrieval data set that has 2,000 selected terms occurred in March 2011, and related tweets from February to August 2011. In the EI experiments, the effects of different filters are firstly evaluated including Event noise filters (like Gaussian, Ideal band-pass filters) and Random noise filters (Mean filter). Based on the evaluation results, the best filters are picked and applied to the proposed Algorithm 1. The identification results are compared with two baseline methods – Co-occurrence based method [27] and Co-occurrence-Spatial-Temporal (CST) based method [13]. In the ER experiments, the proposed method is evaluated with the selected 2,000 terms to detect their underground topics, and the results are evaluated with different metrics including 2nd Moment and Entropy. Manual evaluation is also applied to the results by studying the Relevance of extracted events.

2.5.1 Data Collection

Two sets of data are collected for the two sets of experiments respectively.

Table 2.1: Event Dataset

Dataset	Events	Period	Selected Terms	# of Tweets
IRENE	Irene Hurricane Steve Jobs resigns Earthquake in US	08/20/2011 -08/30/2011	hurricane, irene, tornado steve, ceo, apple earthquakepocalypse	234,785
JPEQ[1]	Fire Transportation Asylum Nuclear General Information	03/11/2011 -03/20/2011	smoke, fire, crack train, bridge, traffic refugee, asylum, ground nuclear, fukushima magnitude, epicenter	123,502
March	Japan Earthquake Arab Spring New Zealand Earthquake Government shut down Background topic	03/01/2011 -03/30/2011	earthquake, epicenter syria, civil war new zealand, earthquake federal, shutdown @, are, rt	312,021
August	Irene Hurricane Steve Jobs resign Earthquakepocalypse Arab Spring background topic	08/01/2011 -08/30/2011	hurricane, irene, tornado steve, ceo, apple earthquakepocalypse libya, rebel, gaddafi @, are, rt	723,943

[1] The selected terms in JPEQ contain 17 English words and 58 Japanese words. The terms here are translated ones.

Event Identification (EI) Data Set: The EI data set is collected for Event Identification. In this data set, the hidden topics in the set of documents are pre-identified, and each selected term is pre-labeled with one of the topics. Specifically, 4 data sets are collected, in which each includes 3 to 5 events selected from Wikipedia. For each event, a two-steps method is applied to get related terms and messages: (i) determining keywords that best describe the event using word association; and (ii) using keywords selected in the previous step and other event specific constraints to retrieve tweets for the event. In the first step, to select keywords for an event, one or two obvious keywords are identified for an event, like ‘Irene’ for Hurricane Irene. Using the keyword(s) the method goes through the dataset and find other terms that appear together with the keyword(s). 1,000 top most frequent terms are selected and then their tf values (i.e., term count per day) are calculated during the time span $T = T1 + T2$ (period $T1$ plus $T2$ days before the starting day of $T1$). In addition, the words with $tf_{T1} < 3 * tf_{T2}$ are filtered, where tf_{T1} and tf_{T2} denote the average tf during $T1$ and $T2$ ($T2$ is set to 10). Then from the rest of the words, about 15 terms with the highest tf are selected for each event. In the second step, the tweets containing the selected words are retrieved, and posted in the specific time frame and geographical region listed in Table 3.1.

All the chosen events are grouped into 4 sub sets according to when the events happened. The first set *March* includes 5 events in March 2011. The second one *August* contains 5 events in August 2011. The third data set *IRENE* contains 3 events on August 24th, 2011, and the fourth data set *JPEQ* contains only Japan Sendai Earthquake 2011 with 5 sub topics in the earthquake. The details of the events are listed in Table 3.1.

Event Retrieval (ER) Data Set: The ER data set is collected for Event Retrieval. In this data set, the ground truth about the hidden events is unknown, and the size of the data set is much larger than EI data set. The ER data set contains 2,000 terms extracted from the tweets in March 2011, and all the tweets in the sampled set with these terms from February 2011 to August 2011. Specifically, each tweet is tokenized in the sampled set according to blank space, and filter all the non-alphabetical words and stop words. For all the terms on March, all the terms whose term frequency (tf) do not satisfy $tf(March) > 3*tf(February)$ or $tf < 10,000$ are filtered out. For the rest of the terms, their timestamps are retrieved from the tweets, and the timestamps are bucketed into bins with a width of 1 day to get the vector $\langle p_1, p_2, ..., p_{180} \rangle$ for each term. p_i is the normalized tweet count for day i. Entropies are then calculated and used for sorting all the words. At last, 3 sub sets with top 500, top 1,000 and top 2,000 terms are collected for March 2011.

2.5.2 Parameter Setup

For each selected term in the EI data set and ER data set, the temporal, spatial, spatial-temporal signals, and the co-occurrence between each pair of them are first computed. And then the distance between each pair of terms are measured based on the extracted signals.

Temporal Distance: For temporal signals of terms, given the period and bounding box in Table 2.1, the timestamps of terms are bucketed into bins. The width of each bin is 1 hour in the Experiment 2.5.3. For Experiment 2.5.4, 1 day is used as the width. And then the F_{t,w_i} – the count of term w_i in each bin is calculated and normalized. The temporal distances based on F_{t,w_i} between w_i and w_j is defined with:

$$D_t(w_i, w_j) = \sum_t |F_{t,w_i} - F_{t,w_j}| \tag{2.8}$$

Spatial Distance: For spatial signals of terms, the bounding-boxes for terms are separated into $100 * 100$ mesh grids, and the normalized term count for each grid F_{l,w_i} is calculated. The temporal distance between any w_i and w_j is defined with:

$$D_l(w_i, w_j) = \sum_l |F_{l,w_i} - F_{l,w_j}| \tag{2.9}$$

Spatial-Temporal Distance: To extract the temporal-spatial signals of terms, for each day, the bounding-boxes for terms are gridded into $100 * 100$ mesh grids, and then the term count F_{t,l,w_i} for each time unit t and each spatial grid l are counted and normalized. The spatial-temporal distance between w_i and w_j is defined with:

$$D_{t,l}(w_i, w_j) = \sum_{t,l} |F_{t,l,w_i} - F_{t,l,w_j}| \tag{2.10}$$

Co-occurrence Distance: First, the co-occurrence count $o(w_i, w_j)$ of each pair of terms is counted, and the total co-occurrence count $O(w_i)$ for w_i is computed with $O(w_i) = \sum_{j,j \neq i} o(w_i, w_j)$. Then the distance between w_i and w_j is defined with Equation 2.11 [27].

$$D_o(w_i, w_j) = \frac{O(w_i)O(w_j)}{o(w_i, w_j)} \tag{2.11}$$

For the terms which never co-occur, their distance is set with an infinite value.

Co-occur-Spatial-Temporal Distance: To integrate the co-occurrence, temporal and spatial distances together, the Co-occurrence-Spatial-Temporal (CST) distance is defined with Equation 2.12 [13].

$$D_{t,l,o}(w_i, w_j) = (D_o(w_i, w_j) + 1)(D_t(w_i, w_j) + D_l(w_i, w_j)) \tag{2.12}$$

Different filters are tested on removing the Random and Event noises contained in the term signals. Mean filter is used to reduce the Random noise, band-pass filters including Average filter, Weighted Average filter, Gaussian filter and Ideal filter are tried to remove the Event noise.

Mean Filter: For the Mean filter, the number of neighbors for temporal distribution is set to 4, so the $[t-2, t+2]$ is used for temporal signals. A square with $width = 2$ is used for spatial signals, and a cubic with $width = 2$ is used for spatial-temporal signals.

Average Band-pass Filter: In the average band-pass Filter, the weight λ in Equation 2.1 is set to $1/N$, where N is the size of the cluster.

Weighted Average Band-pass Filter: For Weighted Average band-pass filter, the weight λ in Equation 2.7 is set to 0.5.

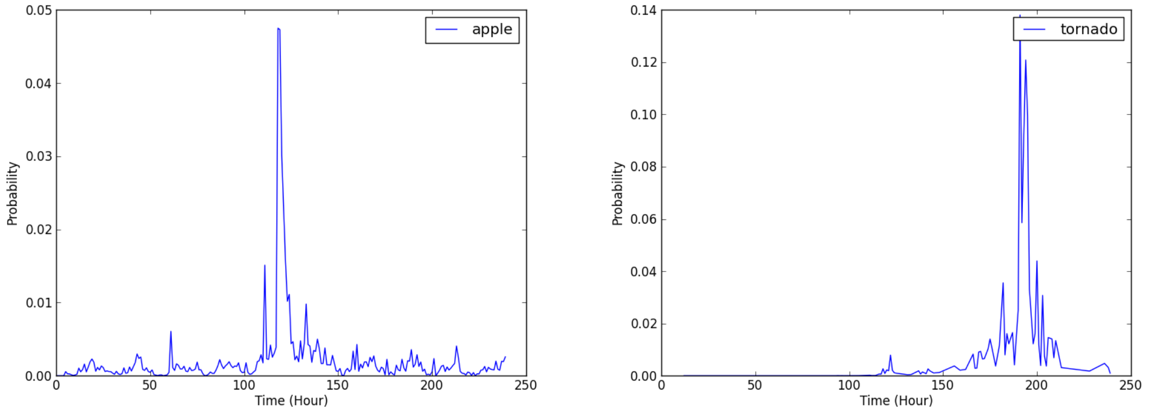
Gaussian Band-pass Filter: For Gaussian band-pass filter, the μ in Equation 2.5 is estimated with the t with the highest term frequency (for temporal signals). σ is estimated with the d where $P((t-d):(t+d)|\theta) = 0.68$. For spatial and temporal-spatial distributions, l and (t, l) are used instead of t respectively.

Ideal Band-pass Filter: For Ideal band-pass filter, the γ is computed with d where $P((t-d):(t+d)|\theta) = 0.68$, λ in Equation 2.6 is set with 0.1. l and (t, l) are used for spatial and temporal-spatial distributions respectively.

2.5.3 Event Identification

In the Event Identification experiment on EI data set, two sets of experiments are designed to evaluate the proposed method. The first set of experiments is to apply different filters in the temporal, spatial and spatial-temporal domains to evaluate the effects of filters using the proposed method. In the second ones, the proposed method are compared with the baseline methods including Co-occurrence based and Co-occurrence-Spatial-Temporal (CST) based methods. K-means is used as the clustering method, and the average results of 10 times experiments are used for evaluation. Purity is used as the metrics.

Band-pass Filter Evaluation: Band-pass filters are studied in different domains including temporal, spatial and spatial-temporal domain. The effects of the filter in the temporal domain are firstly observed, then in each domain, 4 kinds of band-pass filters including Average and Weighted Average band-pass, Ideal and Gaussian filters are compared on the. The comparisons are conducted between the filter based methods and the method without filters.



(a) Temporal Distribution of Term “apple”

(b) Temporal Distribution of Term “Tornado”

Figure 2.3: Filtered Temporal Signals

Take the term *tornado* in the event Irene as an example, as shown in Figure 2.1, before filtering with Gaussian band-pass filter, the term signals of *tornado* have two bursts, of which the second one belongs to the Irene Hurricane event. After filtering by the Gaussian windows, the first burst is diminished in Figure 2.3. Another example is the term ’apple’ in the event of CEO. Compared to the signals in Figure 2.1, the filter effectively reduces the noises generated from the background topics.

Table 2.2: Purity Results in Temporal Domain

data set	IRENE	JPEQ	March	August	Average
Temporal	0.750	0.683	0.400	0.429	0.565
Average	0.813	0.654	0.539	0.582	0.647
Weighted Average	0.881	0.735	0.548	0.504	0.667
Ideal	0.822	0.706	0.426	0.477	0.608
Gaussian	0.795	0.702	0.427	0.455	0.595

Experiments in Temporal Domain: The Average, Weighted Average, Ideal and Gaussian band-pass filters are used on the temporal signals for terms, and the clustering results using filtered signals and unfiltered signals are compared in Table 2.2. Table 2.2 indicates that generally the Event noise filters reduces the noises contained in temporal signal, resulting in better estimation of the distances, and thus achieves better clustering results. Compared with the method with un-filtered signals, the average purities on the 4 data sets using Average filter, Weighted Average filter, Ideal filter and Gaussian band-pass filter are increased by 14.47%, 17.97%, 7.51% and 5.24% on purity respectively. The probability-based filter including Weight-average and Average filter achieves the better results than the window-based filters (Gaussian and Ideal band-pass filter). It is probably because that Gaussian and Ideal band-pass filters put large weights on the detected ROI region, which dramatically change the power of the signals. If the ROI region is not detected correctly, it will incorrectly filter out the actual event signals, causing a severe damage which is different to recover. While in the Average filter, the weights of ROI regions are less than these of window-based filters. Thus it will moderately adjust the power of signals. Even when the ROI region is not detected correctly, there will not be a lot of lost in event signals.

In addition, the improvements on March and August data sets by the noise-filters are more substantial than those on Irene and JPEQ data sets. It is because that the common words in the data sets (March and August share 15 common words representing a general topic), like 'we', 'like'. There are more noises contained in their signals resulting from the widely and daily usage of these words. Therefore, the proposed methods with noise filters achieve better performance on these data sets.

Experiments in Spatial Domain: Table 2.3 shows the clustering results on the 4 data sets using the spatial signals of terms. Compared with the methods with un-filtered spatial signals, Average filter and Weighted Average filter improve the clustering results by 12.67% and 11.72%. While the window-based methods degrade the clustering performance. One possible reason is that assume the gaussian window and rectangle window in the Gaussian and Ideal filters have only one center. However in spatial domain, there are usually multiple centers for some events. For example, for the Irene event, there might exist multiple topic centers due to the transition of the center of hurricane. Therefore single gaussian and rectangle will incorrectly filter the real event signals, and thus degrade the clustering purities. Another possible reason could be that the performance of spatial signals are largely affected by the population density of different regions. If the ROI regions is incorrectly detected due to the population-affected tweet density, the single gaussian and rectangle window will mistakenly filter out the actual event signals.

Table 2.3: Purity in Spatial Domain

data set	IRENE	JPEQ	March	August	Average
Spatial	0.681	0.6623	0.375	0.378	0.524
Average	0.818	0.727	0.338	0.479	0.590
Weighted Average	0.750	0.733	0.433	0.425	0.585
Ideal	0.590	0.246	0.352	0.391	0.395
Gaussian	0.727	0.246	0.367	0.283	0.406

The clustering results in the spatial domain suggest that the spatial filters do not significantly improve the purity of clusters compared to using the temporal based filters, indicating that most events are not converged in a certain location but spread to multiple areas, while they are more likely to be centered at a certain period.

Experiments in Spatial-Temporal Domain: Table 2.4 shows the clustering results using spatial-temporal signals. Surprisingly, the results suggest that before noise filtering, among the temporal, spatial and temporal-spatial signals, spatial-temporal signals achieve the worst performance. That might be caused by that the sparsity of data makes spatial-temporal signals insufficiently represented. And due to the inaccurate representation with sparse data, there are more noises contained in the spatial-temporal signals. Therefore the noise filters achieve the largest improvement in this domain comparing to the results in the temporal and spatial domains. Averagely, the increase on purity for the Average, Weighted Average, Ideal and Gaussian band-pass filters reach to 32.96%, 28.10%, 26.73% and 34.04% respectively.

Table 2.4: Purity in Spatial-temporal Domain

data set	IRENE	JPEQ	March	August	Average
Spatial-temporal	0.636	0.545	0.257	0.256	0.424
Average	0.727	0.532	0.441	0.554	0.563
Weighted Average	0.727	0.538	0.426	0.479	0.543
Ideal	0.727	0.532	0.470	0.418	0.537
Gaussian	0.727	0.532	0.500	0.513	0.568

Integrating Random Noise Filters: Next, the Random noise filter is integrated with Event noise filter, and compare this method with the method using unfiltered signals. For Random noise reduction, Mean filter is applied, and for Event noise removing, the Average band-pass filter is used. The method using both Random noise and Event noise filters is compared with: 1) the method only use Event noise filter (Average band-pass filter), and 2) the method without any filters. The average purities on the 4 data sets in different domains is shown in Figure 2.4.

In Figure 2.4, it is observed that by applying Mean filter, in both temporal and spatial domains, the clustering purities decrease. And the purity only increase slightly in the spatial-temporal domain. It is likely that the grain of the signals is

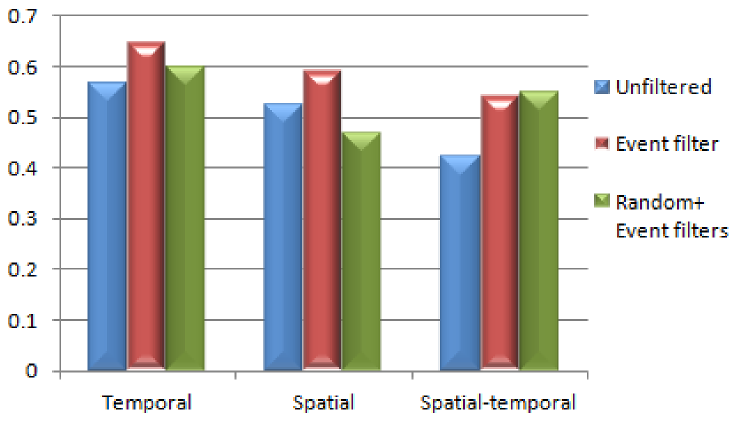


Figure 2.4: Average Purity of the Methods Using Random and Event Noise Filters

large in this work, e.g., in the spatial domain, the grid is 1*1 degree, so the neighbors for each grid are not closed. Therefore the mean filter, which uses the neighbors' signals to smooth the grid's signals, does not reduce the noises but introduces new noises. And the same reason applies for the temporal signals. But for the spatial-temporal domain, due to its higher dimension, the grain is smaller and thus the neighbors are closer than those in the temporal and spatial domains, therefore the Mean filter performs better.

2.5.3.1 Comparison with Baselines

Table 2.5: Average Purity Comparison

Methods	Co-occur	CST-based	filtered CST-based
Purity	0.5875	0.5803	0.6244

Based on the results in the last section, Average band-pass filter is adopted in the proposed method to filter noises in temporal and spatial signals, and the

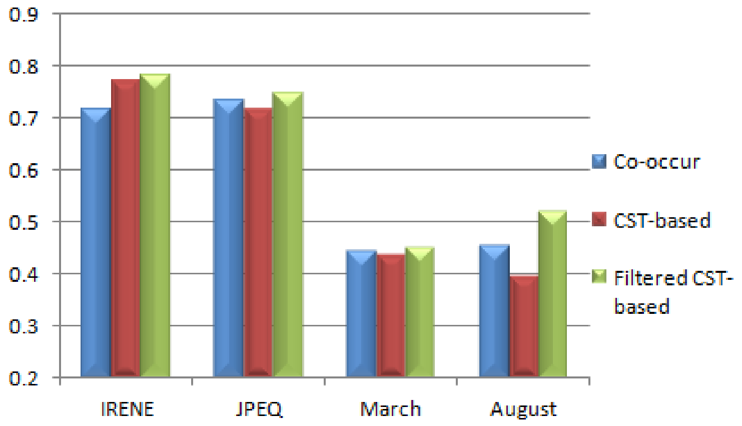


Figure 2.5: Comparison with Baseline Methods

Equation 2.12 is used to measure the closeness between terms. This method is named Filtered CST-based method here, and compared with two baseline methods: (i) Co-occurrence based method [10] using Equation 2.11 to computer the distances between terms; (ii) CST-based method [13] which also uses Equation 2.12 to compute the distances with term signals.

From Figure 2.5 and Table 2.5, it is observed that among three methods, the co-occurrence based and CST-based methods achieve comparable performances. the proposed Filter CST-based method perform the best over all the four data sets. Averagely, the Filtered CST-based method has an improvement of 6.26% and 7.60% over the co-occurrence based and CST-based methods. The results indicate the proposed method is effective in filtering the noises in the signals and improves the event identification.

2.5.4 Event Retrieval

To test whether the proposed method works on large data set, it is applied to the event retrieval on the ER data set. ER data set contains about 2,000 selected terms and their related tweets from March 2011 to August 2011. Affiliation propagation clustering is applied on ER data sets, and 2nd moment and Entropy are used as the metrics to evaluate the cluster results. [10, 22, 1] suggest that better clustering results will approach larger 2nd moment value and smaller entropy value in the temporal domain, because larger second moment or smaller entropy indicate the internal terms in the cluster share more similar and spiky patterns. The average second moment and entropy definitions for clusters are given in Equation 2.13 and Equation 2.14.

$$mo_{average} = \sum_{\theta} \frac{N_\theta}{N} \sum_{t} F_{t,\theta}^2 \tag{2.13}$$

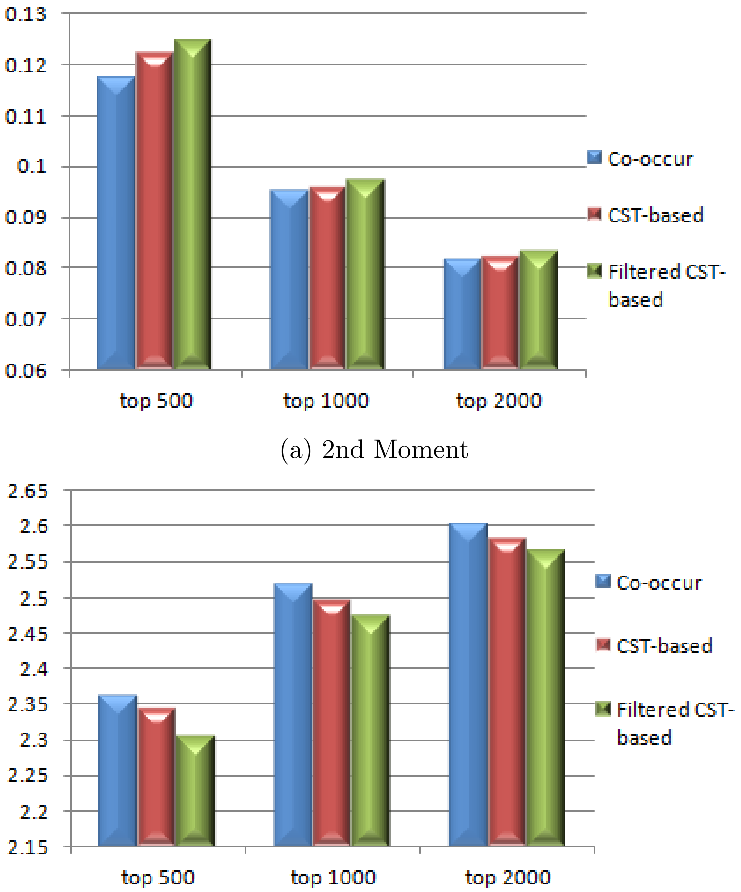
$$entropy_{average} = \sum_{\theta} \frac{N_\theta}{N} \sum_{t} -F_{t,\theta} log(F_{t,\theta}) \tag{2.14}$$

where N_θ is the number of terms belonging to θ, N is the total number of terms.

Based on the clustering results, the temporal event signals $F_{t,\theta}$ is first computed using Equation 2.1 for each cluster, then the moment and entropy are calculated with Equation 2.13 and Equation 2.14 to evaluate the clustering results.

For the 3 data sets containing the top 500, top 1,000 and top 2,000 terms according to the entropy values, in each data set the terms which never happen with others are further filtered. Finally, 3 data sets with 345, 864, 1922 terms are collected respectively. Three methods including Co-occurrence based, CST-based and Filtered CST-based methods are applied on these data sets. The evaluation results are shown in Figure 2.6.

Figure 2.6 shows that the Filtered CST-based method achieves the best results according to both 2nd moment and entropy, indicating the positiveness of proposed method. Averagely, Filter CST-based method improves by 3.75% and 1.69% than Co-occurrence based method and CST-based method on 2nd moment, and improves by 1.87% and 1.04% on entropy. T-test results in Table 2.7 shows that the improvement



(a) 2nd Moment

(b) Entropy

Figure 2.6: Event Retrieval Evaluation

Table 2.6: The Number of Detected Clusters

Method	top 500	top 1000	top 2000
total terms	345	864	1922
co-occurrence	91	186	336
CST-based	75	170	325
filtered CST-based	85	192	368

Table 2.7: T-test Results

Method	v.s. Co-occur	v.s CST-based
p-value (2nd moment)	0.0876	0.0317
p-value (entropy)	0.0074	0.0258

of the proposed method is significant comparing to the other two baseline methods (with the significant level $\alpha = 0.1$).

The top 8 events are selected based on the 2nd moment values, and listed in Table 2.8. Manual evaluation is applied to the extracted events according to two rules: Are they related to each other (Closeness)? Are they related to an event (Relevance)? The first question is answered with Yes or No; the second question is scored with 0, 1, 2 for each cluster, where 0 means“not an event”, 1 stands for “hard to tell”, 2 represents“is an event”. 8 people from three different majors evaluate these events, and the average scores are listed in Table 2.8.

For the Co-occurrence based method, all the terms are correctly clustered, but the average score of the Relevance for the 10 events is the lowest with only 1.150. For CST-based and Filtered CST-based method, the clusters are also correct regrading the Closeness. According to the Relevance, the Filtered CST-based method reaches the best Relevance scores with 1.575. Also it is also shown that for some clusters extracted by all of these three clusters, Filtered CST-based method achieves the better results than the other two methods. For example, the cluster in Co-occurrence

Table 2.8: Extracted Events

Method	Event No.	Top 8 Detected Clusters	Close.	Rele.
Co-occurrence Based	1	“geiger”,“tehachapi”,“fukishima”,“fukushima”,“clicks”	Y	1.375
	2	“japon”, “alerta”	Y	1.625
	3	“gears3beta”, “gearsviking”, “seaside”	Y	1.625
	4	“united”, “manutd”	Y	1.250
	5	“bernabeu”, “realmadrid”, “malaga”, “halamadrid”, “realmadrid”	Y	1.750
	6	“iremos”, “sortear”, “autografado”, “concorrer”, “aew”, “cfmaritrindade”, “participem”	Y	1.375
	7	“chernobyl”, “radiation”, “nuke”	Y	1.500
	8	“pipefitter”, “pipefitter”	Y	0.125
MST Based	1	“geiger”,“tehachapi”, “fukishima”, “fukushima”, “clicks”	Y	1.375
	2	“alerta”, “japon”	Y	1.625
	3	“united”, “drogba”, “manutd”	Y	1.500
	4	“tsunamis”, “usgs”, “devastating”, “swept”, “naruto”, “tornados”	Y	1.625
	5	“lisalampanelli”, “chernobyl”, “radiation”, “nuke”, “radioactive”	Y	1.500
	6	“iremos”, “sortear”, “autografado”, “concorrer”, “lojacdbrasil”, “cfmaritrindade”, “participem”	Y	1.375
	7	“messi”, “lionel”, “4-1”, “sesimbra”, “alves”, “ucl”, “penal”	Y	2.000
	8	“provas”, “geografia”, “portugues”, “filosofia”, “biologia”, “muuuito”, “quimica”, “matematica”, “materia”,	Y	0.500
Filtered CST Based	1	“earthquake”, “finder”, ”90999”	Y	1.750
	2	“geiger”, “tehachapi”, “fukishima”,“fukushima”, “clicks”	Y	1.375
	3	“alerta”, “japon”	Y	1.625
	4	“tsunamis”, “usgs”, “devastating”, “swept”, “naruto”, “tornados”	Y	1.750
	5	“iremos”, “sortear”, “cfmaritrindade”,“participem”	Y	1.250
	6	“mcilroyrory”, “united”,“drogba”, “manutd”	Y	1.375
	7	“messi”, “lionel”, “alves”, “ucl”	Y	2.000
	8	“bernabeu”, “realmadrid”, “realmadrid”, “malaga”, “halamadrid”, “sorteia”	Y	1.625

based method – “united”, “manutd”, the Relevance score is 1.250, because it can only be inferred that this cluster is related to the soccer team Manchester United, but cannot be specified which event they are related to. Filtered CST-based method both approach a better score 1.375, because they detect more terms including “Drogba”, and thus it can be concluded that it is an event about Didier Drogba with Manchester United in March 2011. The reason that Co-occurrence based method only detects two term is because of the lack of co-occurrence caused by the short content in tweets. Since the proposed method considers both occurrence and spatial-temporal distance, therefore it can find more similar terms with “united” or “manutd”. Another example is the cluster in CST-based method – “mess”, “lione”, “4-1”, “sesimbr”, “alve”, “ucl”, “pena”, the CST-based method in-correctly groups the terms “4-1” and “sesimbr” to the cluster containing the terms “ucl”, “messi”, “lione” and “alve”. But actually they are related to different games: “4-1” and “sesimbr” refers to the game Pinhalnovense 4 - 1 Sesimbra on 13th March 2011, and the other 4 terms are related to the game Barcelona 3-1 Arsenal on 8th March 2011. The error is probably cause by that “4-1”, “messi” and the others are popular terms that could occur in different events. Therefore there exists Event noise in their signals which result in the mis-clustering. By applying band-pass filter, the proposed method effectively reduces the Event noise and correctly separates these two events.

2.6 Summary

This section addresses the problems of event detection from social media, focusing on better estimating the distances between terms based on their spatial-temporal signals for a specific event. The spatial-temporal term frequencies are treated as signals, and introduce noise filters to filter different sources of noise. To remove the Event noise, an iterative method is designed to cluster the terms based on the filtered signals, and the band-pass filters are generated to filter noise based on the clustering results. Experiments on a series of collected EI data sets from Twitter indicate that the proposed method can effectively remove the Event noises for terms, improving the event identification performance. Also experiments on 6-months tweet data set from Twitter show encouraging results for the proposed event-retrieval methods.

3. EVENT TRACKING : POPULATION MODELING

3.1 Introduction

The previous section described how to detect events in social media based on a signal-inspired method. Once an event is extracted, the next step is to track the event. For example, how do subtopics evolve over time, how does the mood of users affiliated with an event change over time, how large is the crowd associated with an event, and how does this crowd size change over time. This section focus on tracking event-driven crowds as they first form, evolve, and eventually dissolve through the development of new population models for capturing the dynamics, duration, and population of these crowds.

Toward the goal of modeling and tracking crowds in social media, this section focuses on modeling the *population dynamics* of these crowds as they first form, evolve, and eventually dissolve. The goal is to study the potential of social media for building crowd population models that can estimate the dynamics, duration, and lifecycle of crowds that may form in these systems. In this way, population models may reveal crowds that will continue to grow and those that are on the decline, as well as providing the basis for new advances. For example, robust population models built over user-contributed posts could predict future population density of restaurants, bars, and other local hotspots; urban planners and local governments could have access to real-time population maps, reflecting the current movements of people through space (rather than reflecting stale census estimates or relying on expensive sensors); companies and investors could adjust their marketing strategy, and allocate their limited resources based on the population of users drawing attention on their products; and political groups could estimate the percentage of people voting for or protesting against a new policy, with the help of population modeling for crowds.

Concretely, the examination in this section are focused on two types of crowds:

- **Event-driven crowds:** reflecting a collection of people who are discussing or participating in a specific event, e.g., users posting about the superstorm Sandy or users participating in an anti-government protest in Syria.

- **Location-driven crowds:** reflecting groups who are bound to a certain place, e.g., people posting messages from Manhattan or from a Starbucks located in Pike Place, Seattle.

3.2 Preliminaries

This section describes the basics of population modeling, highlights several challenges to successfully developing models over social media, and presents the crowd-based datasets used in the following sections.

3.2.1 Challenges to Population Modeling

Intuitively, population can be modeled using the number of births, deaths, immigration, and emigration. In the basic population model, suppose a place has a population of N_t at time t. Denoting the number of newborns as B_t, the number of deaths as D_t, the number of immigrants as I_t, and the number of emigrants as E_t, then the population for this place at time $t + 1$ is defined as:

$$N_{t+1} = N_t + B_t - D_t + I_t - E_t \tag{3.1}$$

According to Equation 3.1, the population increase from times t to $t + 1$ is the difference between the number of births and deaths, plus the difference between the numbers of immigrants and emigrants. In the context of short-lived crowds, it can be assumed that the birth and death rates are close to 0, leading to a population model based purely on immigration and emigration:

$$N_{t+1} = N_t + I_t - E_t \tag{3.2}$$

Although seemingly straightforward, there are a number of challenges to model population from user-contributed artifacts in social systems:

- While it is natural to estimate immigration using check-in data observed from posts (reflecting users who newly joined a crowd), it is unclear how to estimate checkout behavior. Users typically do not explicitly indicate the time that they leave a crowd, meaning that population modeling via user-contributed posts alone is insufficient.

- Crowds in social media may suffer from data sparsity due to small coverage—since only a small percentage of people will post about their activities—and a low posting frequency—since the posts of a user may be too infrequent to capture fine-grained crowding behavior.

- Noise may also be introduced for many reasons including repetitive check-ins, incorrect location information, and misclassification of crowd-related posts.

3.2.2 Data Collection

For the analysis and experiments in the following sections, two sets of data are used: (i) a **Location Dataset** for analyzing location-driven crowds; and (ii) an **Event Dataset** for analyzing event-driven crowds.

Location Dataset: The first dataset contains user posts from several popular location-based services (e.g., Foursquare, Twitter, and Gowalla). Every post includes a timestamp (i.e., creation time) and the location (i.e., geographical co-ordinates) where the message was posted. This dataset was collected between October 2010 and January 2011 and it contains more than 22 million posts from 603,796 unique venues. About 62% of these posts are associated with a venue and every venue has about 23 posts on average [28].

Event Dataset: The second dataset contains event-related tweets that were collected using the Twitter API. Initially, around 1 billion tweets are collected from February 2011 to March 2012 with an average of around 3 million tweets every day. For the experiments using event-driven crowds focus are put on 6 major events in the dataset, which are listed in Table 3.1.

To identify event-related tweets, first a set of keywords associated with each event is determine, and then only tweets passing an event similarity threshold are kept. To determine the set of keywords associated with an event, first one or two obvious keywords are identified (like 'Irene' for Hurricane Irene). Using these seed keywords, 1,000 most co-occurring words are identified, and then their average term frequency (tf) is calculated per day during the time span of the event ($T1$) and during the ten days prior to the start of the event ($T2$), giving us tf_{T1} and tf_{T2}. Then, the words that occur relatively infrequently during the event are filtered: $tf_{T1} < \lambda tf_{T2}$, where λ is a threshold set to 3 in this case. This method yields about 20-50 event-related keywords, which are then used to represent an event as a vector. Next, the term vector for each tweet is computed by tokenizing the tweet using whitespace as a delimiter, and the cosine similarity of the keyword vector and the term vector is calculated. Finally, all the tweets passing a similarity threshold are collected in the dataset.

Noise Filtering: For the Location Dataset, the noises introduced by the incorrect location information are reduced by filtering out the check-ins from users whose successive check-ins imply a rate of speed faster than 1000 miles-per-hour [28]. For the Event Dataset, to reduce the noises caused by the misclassification of the event-related tweets, only the tweets within a pre-specified time frame and geographic boundary are kept. For example, Linsanity had a timeframe of the first two weeks of February 2012 and was constrained to the geographic boundary of United States. The tweets that are outside the time frame of the event or the corresponding boundary are filtered, and finally above 10,000 related tweets are collected for each event (Table 3.1).

Table 3.1: Event-driven Crowd Dataset

Event	Period $T1$	Box B	Top3 words	#Tweet
JPEQ	03/11/2011 -03/15/2011	US	tsunami, japan, earthquake	19281
Irene	08/20/2011 -08/24/2011	US	hurricane, irene, tornado	10352
SteveJobs	10/05/2011 - 10/09/2011	US	steve jobs, rip, apple	31738
Wedding	04/29/2011 -05/03/2011	UK	wedding, royal, kate	21551
Linsanity	02/04/2012 -02/14/2012	US	jeremy lin, linsanity, knicks	10369
Election	04/01/2012 -04/30/2012	US	obama, romney, president	30098

3.3 Crowd-Based Population Model

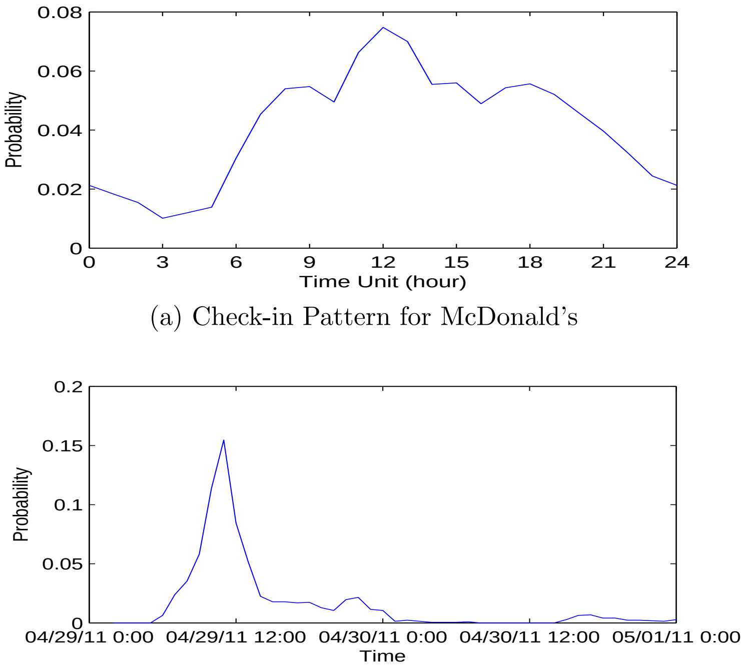
This section develops the crowd-oriented population model. It shows how to estimate the “immigration” (or check-in) population and the “emigration” (or checkout) population, before fully developing the population and emission-based models for understanding crowd dynamics.

3.3.1 Estimating "Immigration"

To estimate the number of "immigrants" for a crowd r, the temporal posting pattern is modeled for r using the timestamps of user-contributed posts, and use this pattern as check-in pattern to approximate the actual population of people checking in at r given a time.

Using the posts from the two data sets, check-in patterns can be generated for location-driven or event-driven crowds. For location-driven crowds, given a place l, the timestamps of l's posts are extract, and the timestamps are normalized into 24 time units representing the 24 hours in a day (if a user published multiple posts in l within 24 hours, only her first post is used). An example check-in pattern for McDonald's is displayed in Figure 3.1a, where x-axis represents the time unit and y-axis is the normalized count of check-ins, representing the check-in probability given a certain time. It can clearly be seen that there are three peaks around 8:00, 12:00 and 18:00, and 12:00 has the highest frequency of check-ins. The peak times are consistent with breakfast, lunch and dinner time, and indicate that McDonald's is most popular for a quick lunch. For event-driven crowds, given an event e and its period, the timestamps of associated posts per hour are normalized (for users who contribute multiple posts during the period of e, only the first one is taken into account). For example, Figure 3.1b is the check-in frequency of the Royal Wedding 2011 in the UK between April 29th and April 30th. It is found that the check-ins burst at April 29, 2011, and the peak hour is around 11:00 local time, which is exactly the highlight of the whole wedding.

Similar check-in patterns have been studied in [28], [29] and [10], where the check-in patterns associated with locations or event were shown to reveal semantic



(a) Check-in Pattern for McDonald's

(b) Check-in Pattern for Royal Wedding 2011

Figure 3.1: Examples of Check-in Patterns

information, for example for automatically grouping related locations based on the similarity of their check-in patterns (e.g., reflecting that coffee shops tend to have similar "immigration" patterns).

3.3.2 Modeling Duration to Estimate "Emigration"

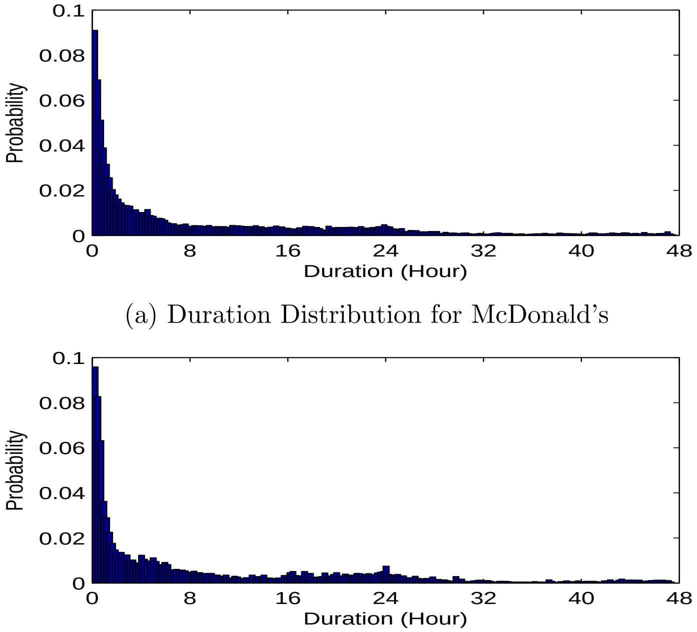
In analogy to check-ins, checkouts are used to estimate the number of "emigrants" checking out from a specific crowd. However, since users only publish posts when they join a crowd – like tweeting when they arrive at a place or participate in an event, and do not explicitly post announcements when they leave – the checkout number cannot be measured directly. To solve this problem, tho work proposes to model the duration of time that users spend in crowd r to estimate when users will check out from r. The duration d here refers to the time a user stays in crowd r. The probability of d is formally defined in Equation 3.3, where R is a crowd, and the subscript indicates the time period (e.g., the crowd at time t).

$$\begin{aligned} P(d|r) = P(&R_{t+d+1} \neq r, R_{t+d} = r, R_{t+d-1} = r, \\ &..., R_{t+2} = r | R_{t+1} = r, R_t \neq r) \end{aligned} \tag{3.3}$$

Location-based durations: For location-driven crowds, it is assumed that once someone checks in at location l, they will spend duration d at location l. From a notation standpoint, the r in Equation 3.3 can be replaced by l to reflect a location-based crowd. Hence, the duration d given a location l can be estimated using the time span between every two posts from the same user, where the first post is from l and the second post is the first one from a different location.

Event-based durations: Different from location-driven crowds where a user can only be in one crowd at a specific time t, event-driven crowds may attract participants who express interest in multiple events since there is no physical requirement of being present at a particular location. As a result, these users can follow multiple events at the same time, and they may leave and return to a particular event e over time. Therefore, the duration for event-driven crowds can be estimated in a different way: from the posts related to an event, all the posts $R_{t_1}, R_{t_2}, ..., R_{t_n}$ (t_i is the index of posts) that belong to a user u are identified, and then the interval of her first and last post is used as the duration d. $t_i (i = 1, 2.., n)$ do not need to be successive (like the posts for location-driven crowds do). If there is only one tweet for a user in e, then the duration is assumed 0, which means this user does not stay in e.

Example: Taking the crowds at McDonald's and Best Buy as examples, the duration distributions for these location-driven crowds are plotted in Figure 3.2. To illustrate event-driven crowds, the duration distributions for the people involved in the Japan earthquake and the Royal Wedding are shown in Figure 3.3.



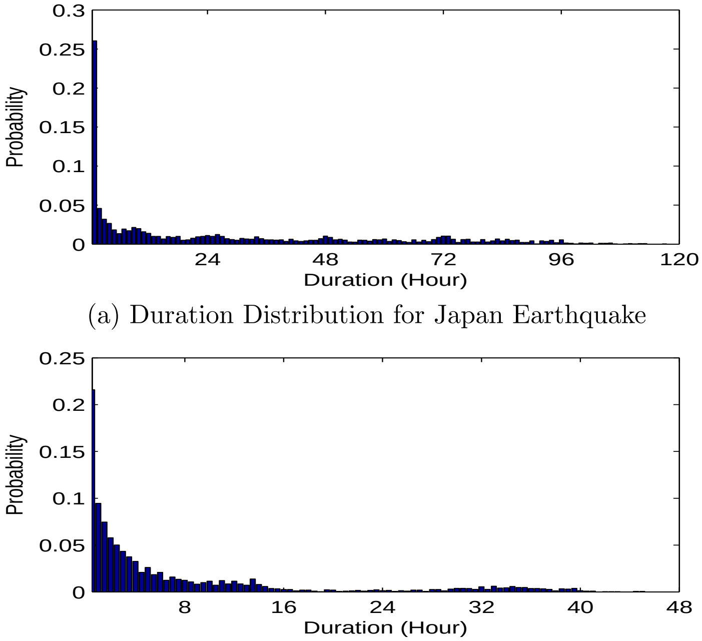
(a) Duration Distribution for McDonald's

(b) Duration Distribution for Best Buy

Figure 3.2: Example of Duration Distributions for Location-Driven Crowds

For McDonald's and Best Buy, the probabilities peak in the first half an hour and decrease following a power decay law when the duration increases, which appears intuitively reasonable. However, It is also observe that the durations derived from measuring inter-posts times display some anomalies. Since many users may post only infrequently, the Figure 3.2 shows that there are a number of people apparently with a duration of 24 hours or more at McDonald's. Similarly, it shows a spike after 24 hours at McDonald's and Best Buy, most likely capturing people who posts only for these location and nowhere else (resulting in a one day "duration").

Figure 3.3 shows that the duration probability for event-driven crowds also follows a power decay law and has a long tail (Figure 3.3 only plots the duration distribution of the users who spend $d > 0$ on the events). Compared with the location-driven crowds, people tend to spend less time on events. For example, the probability



(a) Duration Distribution for Japan Earthquake

(b) Duration Distribution for Royal Wedding

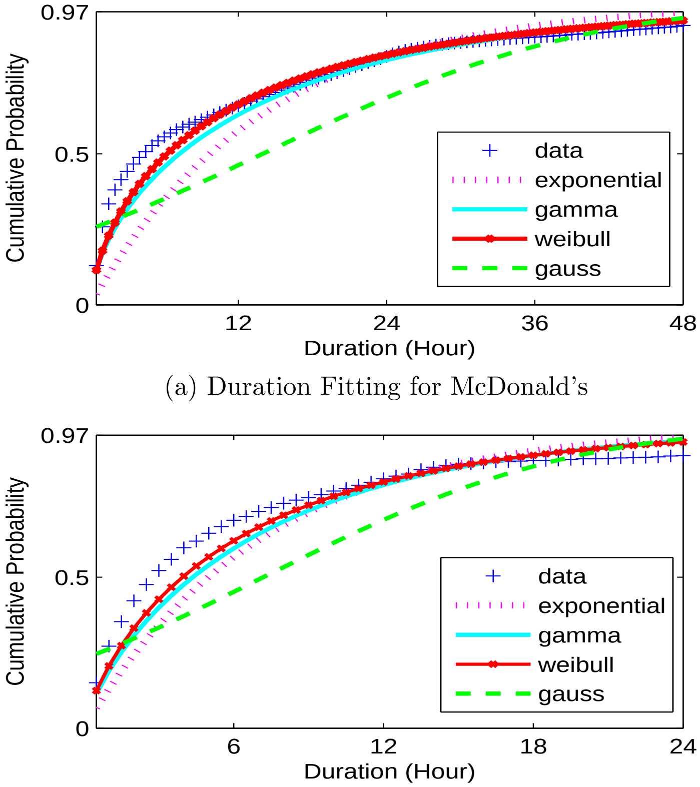
Figure 3.3: Examples of Duration Distributions for Event-Driven Crowds

$P(d = 0|JPEQ)$ is 77.61%, whereas $P(d = 0|Wedding)$ is 56.72%, which means that fewer users stay associated with an event. That is consistent with the reality that most people are just transient viewers for an event but not long-term participants. And comparing to location-driven crowds, they have a longer tail, because events usually last longer.

3.3.2.1 Duration Distribution Fitness

To better understand the duration distribution, given a crowd r, this section examines a series of distributions which are commonly used for duration modeling. In different applications, different probability density functions have been adopted for duration modeling, e.g. [30] proved that the contact duration follows an Exponential pdf in a mobile ad-hoc network, [31] used a Weibull pdf to estimate the response time for traffic incidents. Here, four alternatives are considered: Gaussian, Exponential[30], Gamma[31], and Weibull[32].

Taking two crowds as examples – McDonald's and the Royal Wedding – their cumulative distributions of duration is fitted using different pdfs, and the best-fit results are illustrated in Figure 3.4.



(a) Duration Fitting for McDonald's

(b) Duration Fitting for Royal Wedding

Figure 3.4: Duration Cumulative Distribution Fitness

In Figure 3.4, it is observed that the Weibull pdf fits the data best, followed by the Gamma pdf, and then the Exponential pdf. The Gaussian pdf achieves the worst fitness. Applying the Kolmogorov-Smirnov test also indicates that the Weibull pdf fits the data better than other possibilities. Usually Weibull is more flexible than the Exponential distribution in which the probability of users staying an additional period may depend on their current duration. Therefore it's quite suitable for modeling the duration for these two crowds.

3.3.3 Building the Population Model

Given the duration model defined in the previous section, now this section proposes a time-evolving population model that estimates both when users will depart a crowd and how many users are remaining in the crowd.

Given a crowd r and a time t, the number of people who will check out at time t is the sum of the number of people who checked in at t_0 and stay at the location for $t - t_0$ and the people who checked in at t_1 and stay there for $t - t_1$ and so on. The people who checked in d hours ago before time t check out with probability $P(d|r)$. So denoted $Q_{out}(t|r)$, the population checking out from r at t is in Equation 3.4, where $Q_{in}(t|r)$ is the check-in population given a crowd r at time t, and $P(d|r)$ is the duration probability for people to stay in r for duration d.

$$Q_{out}(t|r) = \int_{t'=0}^{t} Q_{in}(t'|r)P(t - t'|r)\mathrm{d}t' \tag{3.4}$$

Equation 3.4 is a convolution of check-in function and duration functions, the effect is that the volume of check-ins is smoothed and shifted backward by the duration function.

Given a crowd r and a timestamp t, the remaining population is the difference of the total number of people who have checked in before t and the total number of people who have checked out before t. And people who checked in d hours ago before time t would remain with the probability $1 - C(d|r)$, where $C(d|r)$ is the cumulative distribution function for d. Therefore, the population remaining in r at t, denoted by $Q_{rem}(t|r)$, can be estimated as:

$$Q_{rem}(t|r) = \int_{t^{'}=0}^{t} Q_{in}(t^{'}|r)(1 - \int_{d=0}^{t-t^{'}} P(d|r)\mathrm{d}d)\mathrm{d}t^{'} \tag{3.5}$$

where $\int_{d=0}^{t-t^{'}} P(d|r)\mathrm{d}d$ is the cumulative probability $C(t-t^{'}|r)$.

In the experiments described in Section 3.4, given a crowd r, a series of distributions including Gaussian, Exponential, Gamma and Weibull will be tried for $P(d|r)$.

3.3.4 Emission-Based Modeling

Based on the population model, the emissions of the crowds can be further estimated. An *emission* here refers to the products that a user "emits" during their stay in a crowd. To illustrate, for a crowd of people attending the 2012 London Olympics, some will "emit" videos and photos by uploading them to social media sites like Facebook. For a crowd of Apple iPhone 5 fans, some will actually purchase the iPhone, resulting in a crowd emission. For tourists visiting Manhattan, their associated traffic volume can be viewed as an involuntary crowd-based emission.

The goal in this section is to develop an emission model for capturing these crowd-based products. Depending on the application domain, the emission model could be useful across a number of settings including web service providers, product review systems, and hotspot detection applications. Concretely, this work focuses the proposed model on the tweets emitted by a crowd based on the event-based crowd population model.

To estimate the number of tweets, Equation 3.4 is modified to Equation 3.6 by considering the post count per user when they stay in crowd r. Given a time t, the users staying in r are those whose check-in time $t_s \leq t$ and checkout time

$t_e > t$ (their duration is $d = t_e - t_s$). The number of those people can be estimated with $Q(t|r) = \int_{t_s=0}^{t} \int_{t_e=t}^{\infty} Q_{in}(t_s|r) P(t_e - t_s|r) \mathrm{d}t_s \mathrm{d}t_e$. And for each user, the expected number of the posts is $\lambda = \sum_{n=1}^{MaxN} n\eta(n)$, where n is the number of posts, $\eta(n)$ is the probability mass function (pmf) for n. And given a time $t \in [t_s, t_e)$, let the probability that a user posts an annotation at t is $\gamma(t|t_s, t_e)$. Then the number of posts at t can be computed with the user count $Q(t|r)$ at t times the posts count per user $\lambda\gamma(t|t_s, t_e)$ at t. The formula is shown in Equation 3.6:

$$Q_{post}(t|r) = \lambda \int_{t_s=0}^{t} \int_{t_e=t}^{\infty} Q_{in}(t_s|r) P(t_e - t_s|r) \gamma(t|t_s, t_e) \mathrm{d}t_s \mathrm{d}t_e \tag{3.6}$$

The posting probability function $\gamma(t|t_s, t_e)$ is modeled with three distributions: 1) Uniform distribution; 2) Exponential distribution; and 3) U-shaped distribution. These functions are used to check whether the posts of users are evenly distributed or concentrated on the beginning or ending during the event duration $[t_s, t_e)$.

Uniform distribution: In this function, it is assumed that during period $[t_s, t_e)$, each time point has the same probability to emit a post.

$$\gamma(t|t_s, t_e) = \begin{cases} \frac{1}{t_e - t_s} & t_s \leq t < t_e \\ 0 & else \end{cases}$$

Exponential distribution: In this approach, it is believed that it is more likely for people to post when they just check in, and then the chance for posting decreases exponentially when time passes. In the following equation, $\alpha = \int_{t=t_s}^{t_e} e^{-(t-t_s)} \mathrm{d}t$ is the normalizing factor.

$$\gamma(t|t_s, t_e) = \begin{cases} \alpha e^{-(t-t_s)} & t_s \leq t < t_e \\ 0 & else \end{cases}$$

U-shaped distribution: This function assumes that people tend to post when they check in and check out, so the chance is high at their check-in time, then decreases exponentially until at the middle of the period $[t_s, t_e)$, then the chances increase exponentially until at the checkout time. It is a combination of two Exponential function, constituting a U-shape probability function.

$$\gamma(t|t_s, t_e) = \begin{cases} \alpha e^{-(t-t_s)} & t_s \leq t < \frac{t_s+t_e}{2} \\ \alpha e^{-(t_e-t)} & \frac{t_s+t_e}{2} \leq t < t_e \\ 0 & else \end{cases}$$

where $\alpha = \int_{t=t_s}^{\frac{t_s+t_e}{2}} e^{-(t-t_s)}\mathrm{d}t + \int_{t=\frac{t_s+t_e}{2}}^{t_e} e^{-(t_e-t)}\mathrm{d}t$ is the normalizing factors for the function.

3.4 Experiments

Three sets of experiments are designed to verify the proposed duration, population and emission models respectively.

- In the first set of experiments, the goal is to analyze duration and determine if it is informative. The interests lie in evaluating if for a given venue the duration patterns modeled with data reflect the actual time users spend in the venue.

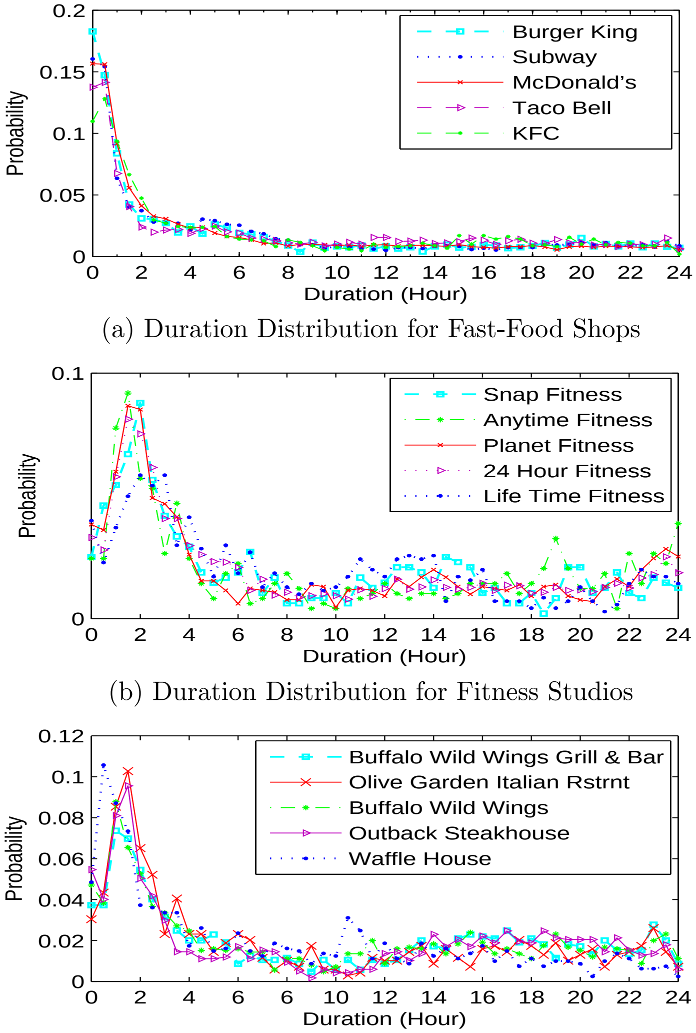
- The second set of experiments analyzes if the population model described in this thesis can be used to predict traffic volume. To verify this, the traffic information is used on Manhattan's bridges as an example. Given the incoming traffic volume to Manhattan, first the duration model is trained for Manhattan using Location Dataset. Then the checkout volume is estimated using proposed population models and compare it with the actual outgoing volume to verify these models.

- The third set of experiments verifies the population and emission models with respect to event-driven crowds. It is evaluated if for a given event and its posts, the population model can estimate the checkouts from that event and the emission model can accurately predict actual number of posts written by the crowd corresponding to that event.

3.4.1 Is Duration Informative?

Even though the duration as measured through inter-post gaps is clearly noisy and not immediately informative, perhaps there are interesting patterns in this duration distribution across crowds? Examining the duration patterns for venues of different types, the duration distributions are plotted in Figure 3.5 , which include three categories of venues: fast food restaurants, fitness centers, and casual restaurants. To reduce the noise of incorrect large durations caused by infrequent posts, all the d are removed with $d > \eta$, where η is set with 24 hours here. Interestingly, it is observed that the duration distribution agrees with the expectation that venues in the same category share very similar patterns. Across different categories, the duration patterns of retail stores and fast-food restaurants are dramatically different from the ones of fitness centers and restaurants, indicating that people tend to spend less time on fast-food venues and retail stores than fitness centers and restaurants.

To further investigate whether the duration pattern can reflect the actual time users spend in a location, this work checks the duration patterns of venues of different types and proposes to analyze the semantic correlation between them, by grouping related venues based purely on check-in and derived durations revealed through lo-



(a) Duration Distribution for Fast-Food Shops

(b) Duration Distribution for Fitness Studios

(c) Duration Distribution for Restaurants

Figure 3.5: Comparison of Duration Distributions

cation sharing services. it is believed if the duration model suggests the real pattern of users' behaviors in physical world, it can be used as a feature for semantic analysis of locations.

114 venues are sampled with the largest number of posts (posts of all distinct venues owning the same name are aggregated, e.g., combining all distinct Starbucks into a single "Starbucks" location) and retrieve their features including: check-in

feature in the form of 24 dimension vector $(c_1, c_2, ..., c_{24})$, the $i^{th}(1 \leq i \leq 24)$ dimension in the vector stands for the probability of check-ins at i^{th} hour during 24 hours; and a duration feature containing 24 dimensions $(d_1, d_2, ..., d_{24})$, the $d^{th}(1 \leq d \leq 24)$ component is the probability of duration is equal to d hours.

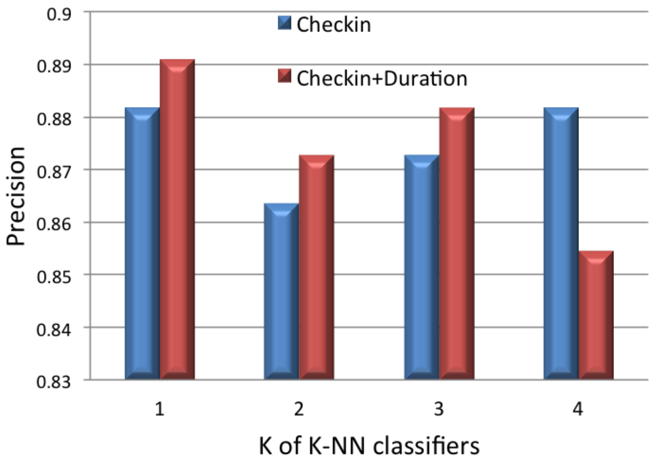


Figure 3.6: Venue Classification

Given the semantic category information ("Food", "Shop", or "Home, Work and Other") retrieved from Foursquare for each of the 114 venues, the set of 114 venues are considered as ground truth data, and the distribution of duration is considered as another feature in addition to the check-in pattern to predict the category label for the venues. A kNN classifier is applied (using Euclidean distance as similarity measure between venues) on the set of 114 venues, and use 10-fold cross validation to evaluate whether the use of duration distribution can improve the identification of semantically-related venues. The classification results for kNN (k = 1, 2, 3, 4) is shown in Figure 3.6. As shown in figure, augmenting a baseline classifier with duration information yields positive results in most cases, suggesting that duration is an informative characteristic derived from location sharing services.

3.4.2 Traffic Prediction with Population Model

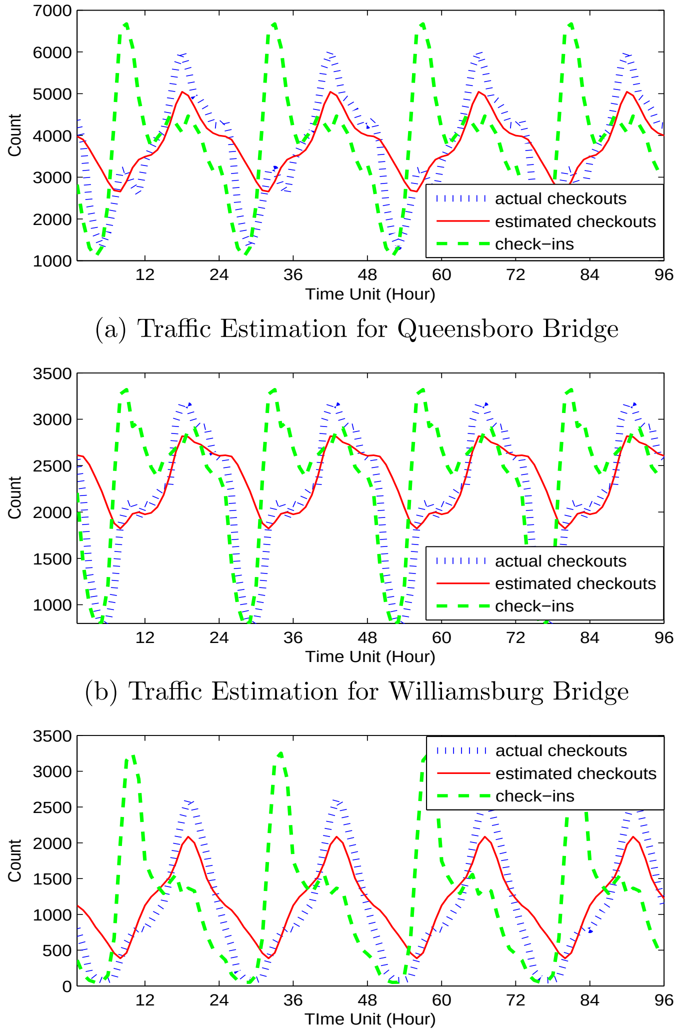
To verify the population model, this work proposes to use it to predict traffic conditions. With this goal, it is needed to build a traffic prediction model. Simply, an area is treated as an enclosed box with only several outlets, whereby people check in and check out using these outlets. Suppose the check-ins for this area is already known, using Equation 3.4, then the checkout population can be estimated. Take Manhattan as an example, which is an enclosed area, the bridges and tunnels connect Manhattan and the surrounding districts including New Jersey, Brooklyn, Queens, Bronx. The traffic volumes on these bridges and tunnels from the surrounding districts to Manhattan are treated as check-ins, and the traffic volumes from Manhattan to these areas are treated as checkouts. 19 two-way bridges and tunnels are considered in this thesis, e.g., the Manhattan Bridge, Brooklyn Bridge, Queensboro Bridge, Williamsburg Bridge, Gorge Washington Bridge, Holland Tunnel, Lincoln Tunnel, Washington Bridge, Alexander Hamilton Bridge, and so on.

The traffic volume data comes from the report "New York City Bridge Traffic Volumes 2010" [33]. It lists the average hourly traffic volumes importing to and exporting from Manhattan through the bridges in 2010, which are two 24-hour volume vector (in_1 ,in_2 , ... in_{24}) and (out_1, out_2, ... out_{24}). (in_1, in_2, ... in_{24}) is used as check-in data, and (out_1, out_2,
... out_{24}) as the ground truth. Equation 3.4 is used to estimate the number of checkouts. The duration pdf $P(d|Manhattan)$ in Equation 3.4 describing how long people stay in Manhattan is trained with the data set containing 22 millions geo-located based posts. A duration is computed by considering the interval of two successive posts by the same user in the Manhattan (and neighboring) areas. For example,

if a user checks in at Brooklyn at t_1, then he posts in Manhattan at $t_2, t_3, ..., t_{n-1}$, and then checks in at Brooklyn again at t_n, $t_n - t_1$ is considered as a duration. With the prior knowledge that most traffic volume come from commuters shuttling between two districts, the durations larger than 12 hours are filtered. At last 1,673 durations are collected to train $P(d|Manhattan)$. Both discrete and continuous functions including Gaussian, Gamma, Exponential and Weibull pdfs are experimented for $P(d|Manhattan)$.

In Figure 3.7, the green dotted line is the repeat check-in frequencies of 4 days, the red solid line is the estimated checkout population using the incoming volumes and the $P(d|Manhattan)$, and the blue dotted line is the actual outgoing volume (checkouts). To illustrate, Figure 3.7a shows the check-ins and checkouts for the Queensboro Bridge. The peak time of check-ins is about 8 am, the peak time of actual checkouts is about 5 pm, which is intuitively consistent with habits arranged around a work schedule. The estimated checkouts display a similar trend with the actual ones and also peak at 5 pm. In Figure 3.7b and 3.7c, the estimated checkouts also fit the ground truth well. This suggests that the duration distribution can correctly capture the lag between check-ins and checkouts, and that the population modeling method with duration is effective in estimating the size of crowds, and correctly capturing the dynamics in population with time passing. It is also notice that the estimated checkouts do not exactly fit the actual checkouts. This difference is attributed to three main reasons: 1) in real traffic, not all incoming vehicles will also depart (check out) by the same route; 2) outgoing volumes may also be contributed by other crowds which are not shuttling between Manhattan and its four neighbor districts; and 3) incomplete coverage of users' trajectory may lead to inaccurate estimation of their duration.

Though the exact traffic volume is hard to predict, the relative volumes can be



(a) Traffic Estimation for Queensboro Bridge

(b) Traffic Estimation for Williamsburg Bridge

(c) Traffic Estimation for Brooklyn Battery Tunnel

Figure 3.7: Traffic Prediction of Bridges of Manhattan

predicted using the estimated checkouts. So the time units is ranked according to the number of checkouts, and compare the ranked list with the list ranked with actual checkout volumes. Thus the prediction problem may be viewed as a rank problem. And since the top ranked time units (rush hours) are what being concerned with most, therefore NDCG (normalized DCG - discounted cumulative gain) is used as the metric to evaluate the ranked results. The equation of NDCG is given in

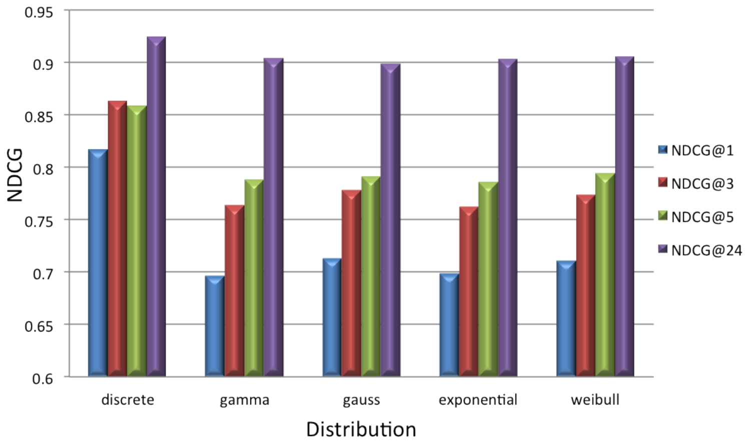


Figure 3.8: NDCG for The Rush Hours Ranked by the Estimated Number of Checkouts

Equation 3.7, where the rel_i is the score for a time unit, IDCG - ideal DCG - is the maximum possible DCG till position k. To calculate rel_i, first the number of checkouts $Q_{out}(t|Manhattan)$ is computed for 24 time units using the proposed models, then the units are ranked decreasingly with $Q_{out}(t|Manhattan)$ and assign each unit a score according to its rank, e.g. the 1st unit is given 24, 2nd is given 23. The ranked results are shown in Figure 3.8.

$$NDCG@k = \frac{DCG}{IDCG} = \frac{rel_1 + \sum_{i=2}^{k} \frac{rel_i}{\log_2 i}}{IDCG} \tag{3.7}$$

For different NDCG@k, the population models all achieve above 70% gains, especially the model using discrete duration distribution reach above 80% gains for all ks. This indicates that the checkout trends can be well estimated with the proposed population model. The different continuous duration achieves comparable performances, where generally Weibull slightly outperforms the other distributions, which is consistent with the hypothesis test result in Section 3.3.2.1. The discrete duration distribution is better than the continuous ones; one reason is the limited data undertrains the models, and the trained continuous model oversmooths the checkouts.

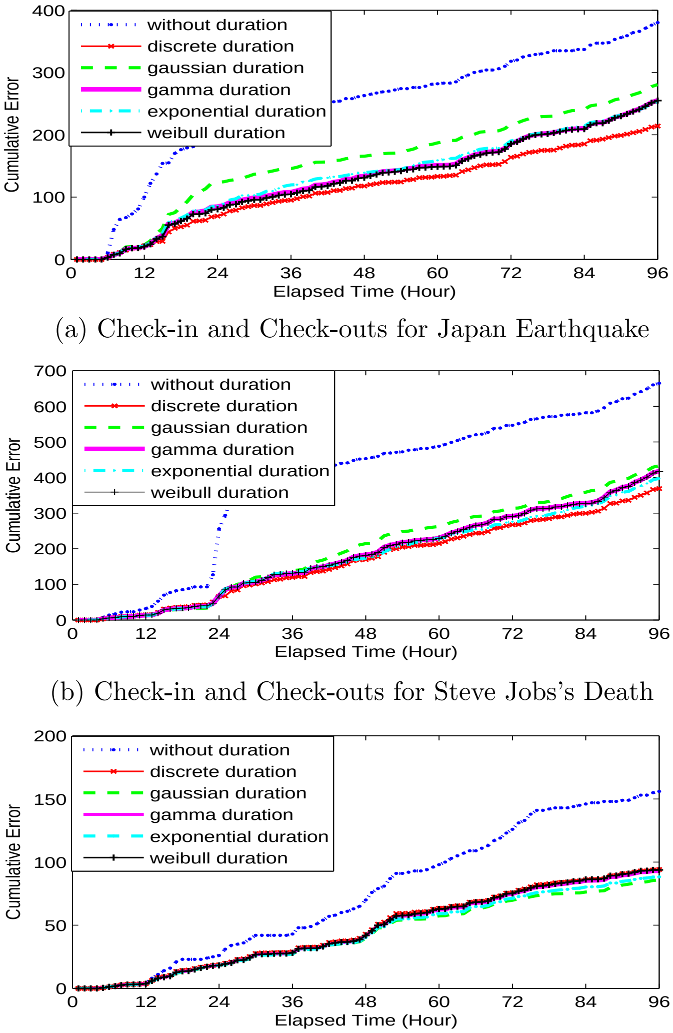
3.4.3 User and Post Prediction with Population Model

In this part, two sets of experiments are conducted to verify the population model and emission model respectively. In the first experiment, the population model is used to estimate the checkouts for the event-driven crowds. Given an event e, the ground truth of checkouts $Q_u(t|e)$ are collected using the number of users who publish a post about e at t and never tweet about e after t. The input data check-ins $Q_{in}(t|e)$ are collected with the number of new users per hour. The $P(d|e)$ is trained with all the pairs of two successive posts related to event e of the same user. Equation 3.4 is used to estimate the number of users who check out $Q_{out}(t|e)$ at time t.

In the second experiment, the emission model is applied to estimate the number of posts for events. Given an event e, $Q_w(t|e)$ - the number of tweets about e per hour is collected as the ground truth and check-ins $Q_{in}(t|e)$ as the inputs. Equation 3.6 is used to calculate the emitting posts. In Equation 3.6, the expected posts number λ and $P(d|t)$ are trained with Event Dataset. To verify the emission model, the estimated post is compared number with the actual post number $Q_w(t|e)$, and residual sum of squares (RSS) is used to evaluate the results.

In both of the experiments, for each event, the tweets are randomly divided into two parts: training data containing the tweets of 80% users and testing data containing the tweets of 20% users.

In Figure 3.9 and Figure 3.10, it is shown that the cumulative errors between the estimated checkouts with the actual checkouts. (The X axis in Figure 3.10 is the elapsed time since the start time of events.) The models are compared with the method with no durations. The cumulative error is calculated using Equation 3.8.



(a) Check-in and Check-outs for Japan Earthquake

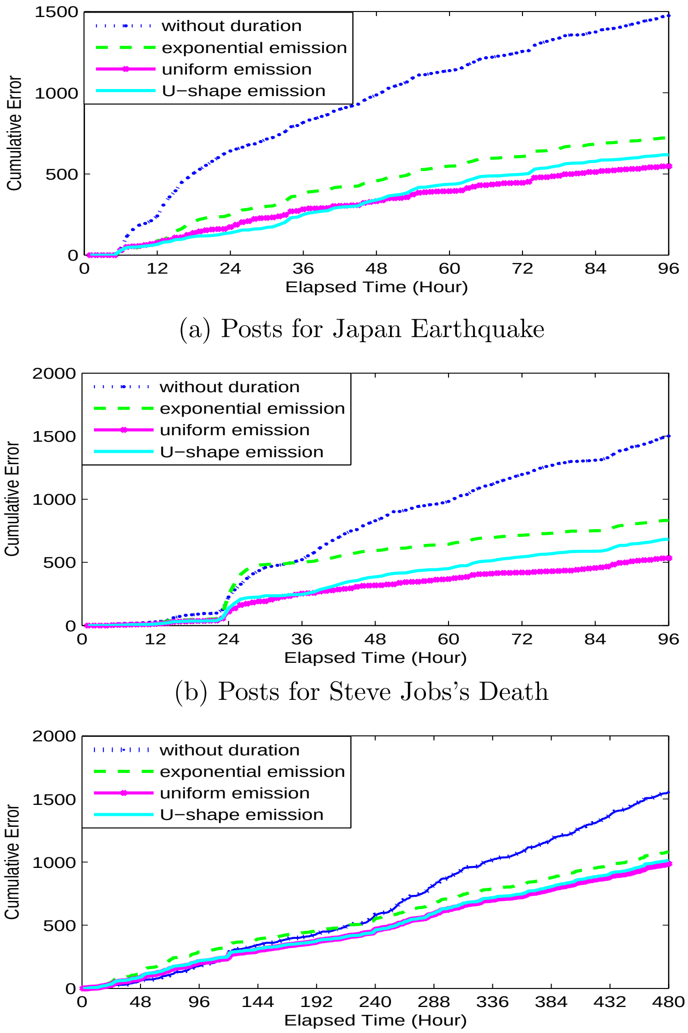
(b) Check-in and Check-outs for Steve Jobs's Death

(c) Check-in and Check-outs for US President Election

Figure 3.9: Estimating Check-outs for Events

$$Error(t) = \sum_{t'=0}^{t} |Q_{est}(t'|e) - Q_{real}(t'|e)| \tag{3.8}$$

In Figure 3.9, the blue dotted line is the error between the actual checkouts and the estimated checkouts using the method with no duration (without duration, the number of checkouts is the same with that of check-ins). In all three examples in Figure 3.9, the proposed models outperform the method with no duration. Among



(a) Posts for Japan Earthquake

(b) Posts for Steve Jobs's Death

(c) Posts for US President Election

Figure 3.10: Estimating Event-Related Posts

the models with different duration pdfs, the Gaussian pdf achieve the worst results in the short-term events, and approaches a good result in the long-term event. Weibull, Gamma and Exponential pdfs have very similar performance, and averagely they achieve the better results than Gaussian distribution. This fact again validates the analysis in Section 3.3.2.1. The discrete function performs the best in the short-term events, probably because that limited training data set does not perfectly display the properties of given pdfs, leading to that all the continuous pdfs do not well fit the data set.

Figure 3.10 shows that cumulative error for the estimated count of posts for different events. Based on the previous experimental results, the discrete duration distribution is adopted for estimating the post count. In Figure 3.10, the blue dotted line is the cumulative error between the actual counts of posts and the estimated ones of posts using the method with no duration (without duration, the posts are the ones contributed by check-in users). In all three examples, the proposed models are better than that with no duration. And among different emission distributions, the Uniform pdf achieves the best performance, Exponential pdf achieves the worst performance. This result indicates that for the people who would like to stay in an event (most users do not stay in an event, their duration is 0), they tend not to write all their posts at the beginning, neither only post at the beginning and end time of the event, in fact they are more likely to evenly distribute their annotations.

Table 3.2: Estimating Checkout User Count

event	no dura.	Discrete	Exp.	Gamma	Gauss	Weibull
JPEQ	68.000	33.382	40.436	40.411	46.033	39.963
Irene	32.665	38.199	39.814	40.767	41.863	40.619
SteveJobs	165.360	59.049	66.456	70.199	71.288	70.361
Wedding	169.505	84.436	103.266	100.767	121.192	98.166
Linsanity	56.665	28.165	29.143	28.917	28.948	29.008
Election	47.138	42.355	43.379	44.206	44.935	44.186

Next, the RSS is listed for the estimated number of checkout users and posts in Table 3.2 and 3.3. Table 3.2 shows that the root RSS of estimated checkouts and posts using the proposed models are much smaller than those of the method without considering duration. For each data set, the best model reduce the root RSS by 50.91%, -16.94%, 64.29%, 50.19%, 50.30% and 10.15% respectively. Among all the duration distributions, discrete function achieves the best performance. Exponential, Gamma, Weibull distribution approach very similar results, Gauss pdf perform worst averagely.

Table 3.3 shows that for each event, the best model reduce the root RSS by 59.55%, 26.66%, 56.03%, 77.97%, 60.00% and 37.05% respectively. Generally, Uniform distribution gets the best results, while for event Irene and Linsanity, the U-shape distribution is the best. The two events have a similarity that they both have multiple bursts, since Irene hurricane is a swift disaster and Linsanity erupted every time Knicks wins. Users' comments tend to concentrate in these bursts, and U-shape pdf has two bursts, so it might fit the data better than other pdfs. And for royal wedding, the Exponential performs the best, this might be due to the fact that the royal wedding is the shortest-term event in these events. People are likely to talk about the event intensively right at the wedding procession, but will not look back later after the wedding.

Table 3.3: Estimating Post Count

event	no dura.	Uniform	Exponential	U-shape
JPEQ	200.242	80.997	102.046	88.372
Irene	61.073	46.811	59.630	44.790
Steve Jobs	218.958	96.285	211.488	127.921
Wedding	862.541	293.299	190.059	335.976
Linsanity	121.737	48.691	74.615	43.432
Election	117.694	74.089	80.998	78.141

In conclusion, though the training data is very noisy and sparse, the duration model is informative for the location-driven crowds. The proposed population model based on duration modeling is effective in estimating the checkout population for both location-driven and event-driven crowds. And the emission model works well on estimating the post number for event-driven crowds.

3.5 Summary

This section proposes to model population given user-contributed posts for crowds. Opportunity for and challenges to population modeling in this noisy and incomplete domain are examined, and a novel time-evolving population model is proposed. To model the population, the distribution of duration derived from posting data is investigated, and it is observed that the durations for location-driven and event-driven crowds follow an power decay law. In addition, given the check-ins and duration models, it is able to estimate the checkout population for crowds at a specific time. This work further applies the population models to emission modeling for crowds to estimate the volume of their posts. Finally, the population model is applied to the traffic prediction and event-driven crowds' population estimation problems. Evaluation with the examples of Manhattan and a set of events shows effectiveness of the proposed models.

4. EVENT ANALYTICS : DISASTER DAMAGE ASSESSMENT

4.1 Introduction

The previous sections describe how to detect events and track the population of crowds in social media. In this section, this thesis explores *event analytics* in the context of earthquake damage assessment. The research goal in this section is to investigate the capacity of social media for conveying damage information, which is an important step for providing responders with rapid insight into the extent of damage to be expected in the field and the locations of greatest damage, which are both necessary for deciding how to best deploy the limited emergency response and recovery resources during the initial moments of an earthquake.

This initial study assesses the quality, coverage, and capacity of two types of social media: text-only tweets, which are typically short and require little effort to post, and media-containing tweets, which include links to either images or videos and are intuitively more expensive in the sense that the person posting must expend effort to capture the picture or video. The initial investigation is reported through an examination of the 2011 Tohoku earthquake in Japan and the 2011 Christchurch earthquake in New Zealand. The study suggests that media tweets provide more valuable location information than text tweets, and that both provide comparable evidence of the linear intensity attenuation function for earthquakes, indicating a similar ability to serve as the foundation of rapid damage assessment for earthquakes.

4.2 Related Work

Recently, with the thriving development of social network services, scientists have begun to study the use of social media on large-scale crises, and apply it to detect, track, summarize and assess them. For example, [34] and [35] examined the social life of micro-blogged information and show how social media can be used for summarizing hazards. By studying 106 million tweets generated, [36] found the majority (over 85%) of detected topics are headline news or persistent news, indicating Twitter plays a more important role as an information source.

Location sharing services have also attracted increasing attention in the last couple of years, and recently have been studied for emergency events. [34] analyzed the temporal, spatial and social dynamics of tweets during a fire emergency, and discussed how the location-based social network can be a source to collect information during emergencies. [4] treated every user as a sensor, and applied Kalman filter to the signals generated by these human-powered sensors to locate an earthquakes' epicenter and to predict the trajectory of the resulting typhoons. [37] explored the relationship between the spatial pattern of geolocated SMS (Short Message Service) messages and the building damage.

Images are increasingly playing a more significant role in disaster detection and summarization. [38] described the evolution of Flickr's role during disaster response and recovery efforts, and discussed this evolutionary growth pattern as a community forum for disaster-related activities. [39] extracted images semantic information under translation model, and use a time-line to summarize the 2011 Tohoku earthquake.

4.3 Data Collection

For the study, two Twitter-based datasets were collected, which are associated with the March 2011 Tohoku earthquake in Japan and the February 2011 Christchurch earthquake in New Zealand. For each, the earthquake-related tweets are identified from an ongoing crawl hosted in Infolab of TAMU that collects around 3 millions geo-located tweets per day. To identify tweets related to each event, several keywords were first selected with the largest counts co-occuring with the seed words “earthquake”, “”, and then filtered some of them according to tf-idf. For Japan, 75 earthquake-related keywords were identified (17 in English, 58 in Japanese); for New Zealand, 15 earthquake-related keywords were identified. Based on these keywords, all tweets, containing at least one of these keywords within the earthquake time-window, were selected. The details for the two collected data sets are listed in Table 4.1.

Table 4.1: Earthquake Data Sets

Event	Time Frame	Selected Terms	#Tweet
JPEQ	03/11/2011 -03/15/2011	earthquake, epicenter, eqjp, honshu, fukushima	207,876
NZEQ	02/20/2011 -02/30/2011	earthquake, christchurch, new zealand, victim, rescue	38,699

The Tohuku earthquake dataset (JPEQ) contains 207,876 tweets, of which 35.41% contain a URL (which links to an image, video, or webpage). The Christchurch earthquake dataset (NZEQ) contains 38,699 tweets, of which 32.55% contain a URL. Each URL-containing tweet is considered as a media tweet, whereas all other tweets are considered as text tweets. On inspection, a random sample of media tweets were found to overwhelmingly include on-site pictures of earthquake damage.

4.4 Approach and Findings

To begin the examination of these two kinds of geo-located social media, a series of investigations are constructed intended to assess the quality, coverage, and capacity of social media in the aftermath of the two earthquakes.

- Epicenter estimation: Firstly, the quality of the two kinds of tweets is assessed for estimating the actual epicenter of each earthquake. Three different features are considered as input to this task, including the tweet density, the re-tweet density, and the average tweets count per user.

- Intensity attenuation: Based on the detected epicenter, the three features are further modeled versus the radius from the epicenter to study the intensity attenuation pattern for earthquakes. It is well known that for any given specific earthquake there exists an "attenuation relationship" that relates the shaking intensity with respect to the distance to the earthquake's epicenter [40]. Are such a pattern in text and media tweets witnessed? And which factors model intensity attenuation the best?

- Spread speed: The temporal and spatial features of tweets are integrated to examine the speed of propagation of text and media tweets. This spread speed is important for understanding the influence of social media for disaster communication.

4.4.1 Epicenter Estimation

In the first study, the capacity of social media is investigated to estimate the epicenter of each earthquake. Three different tweet-based features are considered for epicenter estimation and compare across both text and media tweets. For this case, all tweets are bucketed by applying a grid overlaid on the bounding box of Japan and New Zealand. A grid width of 0.01 degrees is used, which corresponds to about 1.11 km. For each grid cell, the following three features are computed based on the geo-located tweets:

- Tweet density (TD). The first feature is the number of tweets for each grid cell divided by the area of the cell. In this way, the density of tweets is identified for each cell. Intuitively, this feature captures the assumption that people are more likely to post a text or media tweet in the regions that are more severely damaged.

- Re-tweet density (RD). The second feature is the number of re-tweets for each grid cell divided by the area of the cell. Perhaps severely damaged areas re-tweet more. Or perhaps areas outside of the most damaged regions re-tweet based on first-hand accounts from closer to the epicenter.
- User tweeting count (UC). The third feature measures the average number of tweets per user (number of tweets divided by the number of users) from a particular grid cell. The intuition here is that users who are in a damaged region may tend to be engaged with the event longer than those who are not so close, so they might emit more tweets than those who are outside the damaged region.

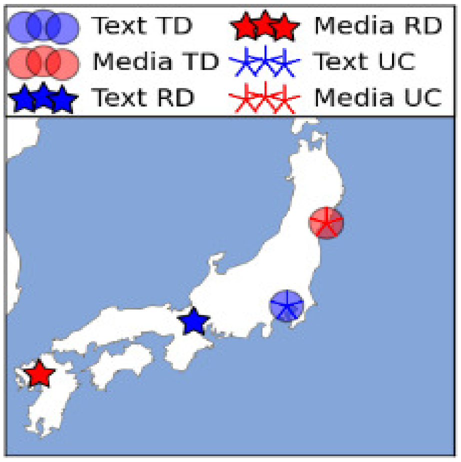


Figure 4.1: Location Estimation Using Different Features

For each feature and for each type of tweet (text versus media), the grid cell with the maximum value is identified as the detected epicenter. In Figure 4.1, the estimated epicenters using the text tweets density and text re-tweets density is located around Tokyo, which has the largest population density in Japan. This fact indicates that the count of text tweets can be easily affected by the population, so they are not good evidence for epicenter estimation. The re-tweet density feature locates the epicenter to the areas which are not badly damaged, which suggests that people who are in the safe places tend to re-tweet posts of others but not generate original content.

In contrast, media tweet density and users' media tweeting count perform the best; the epicenter detected by these features are located in most severe region in the earthquake, and are closest to the actual epicenter. These results suggest that media tweets perform better on epicenter estimation. This is probably due to that media tweets are more likely to happen in the local place of a region of crisis because the scene in the images should be actually observed, while text tweet can happen anywhere no matter where the scene it describes really happens. Therefore, the location information for media tweets is more credible than that of text tweets.

Table 4.2: The Euclidean Distance Between Estimated Epicenter and Actual Epicenter (Degree)

Event	TD		RD		UC	
	text	media	text	media	text	media
JPEQ	3.747	**1.080**	6.950	12.927	3.747	**1.080**
NZEQ	7.066	**0.100**	3.095	3.111	7.066	**0.100**

In Table 4.2, the distance between the estimated epicenter and the actual epicenter is measured (from Wikipedia) with Euclidean distance. It is found that for both JPEQ and NZEQ, the media tweets perform better than text tweets. And the tweet density and user tweeting count achieve the same good results. For JPEQ, since the epicenter is located in the ocean, the detected epicenter is located in the most severely affected region of Japan, so the smallest distance is about 1 degree. For NZEQ, the detected center is close to the actual center, with 0.1 degree difference.

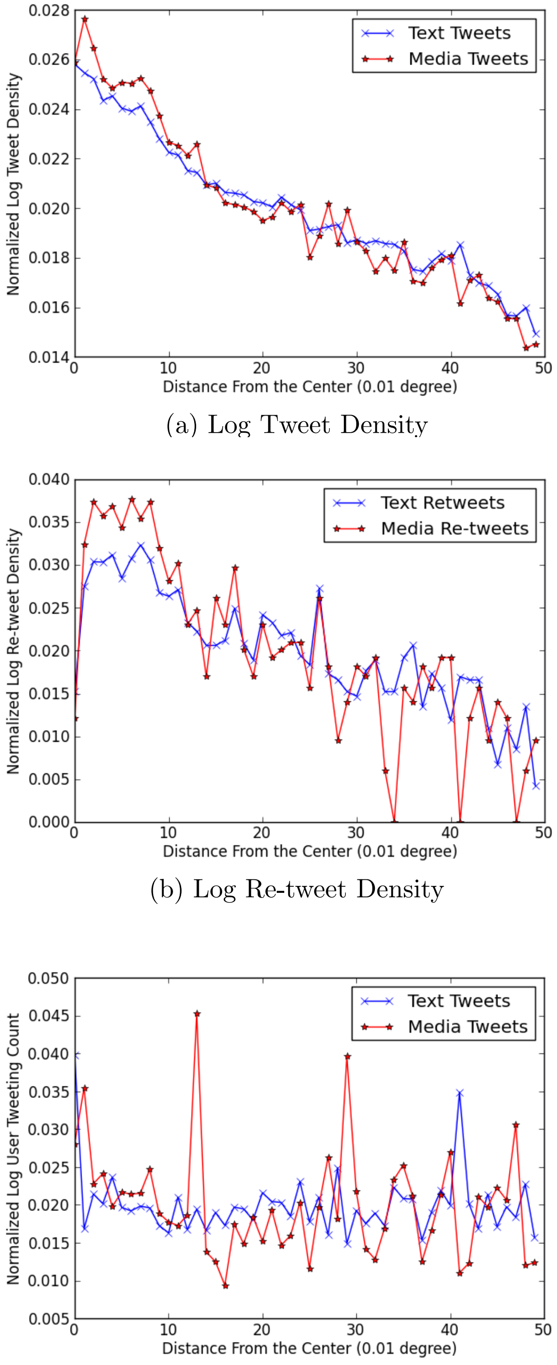
Together, these results show that tweet density and user tweeting count are the best features for identifying the epicenter of an earthquake, that re-tweets tend to happen in the regions that are less affected, and that location information of media tweet is more credible than that of text tweets.

4.4.2 Intensity Attenuation

Based on the detected epicenter, the relationship between text and media tweets on intensity attenuation is examined next. It is well known that for any given specific earthquake there exists an "attenuation relationship" that relates the shaking intensity with respect to the distance to the earthquake's epicenter. For this study, the same three feature are reconsidered as before – tweet density, re-tweet density and users' tweeting count – as well as the two different types of tweets (text and media).

Rather than consider a simple grid, increasing concentric circles are considered around the epicenter. Given an epicenter o for a certain region, the features for the circle region centered at o with radius r are extracted. Then r is increased by 0.01 degree (about 1.11 km), and the features are extracted for the ring area outside the inner circle. At last, the values of features against the radius r are observed. Given the detected epicenter, which has the largest tweet density, the values for the three features versus the radius from the epicenter are shown in Figure 4.2.

Figure 4.2a shows that in the areas close to the epicenter, the log density of tweets is linearly related with the radius r, which means the density decreases following a power law. This linear relationship is consistent with the previous seismic intensity



(a) Log Tweet Density

(b) Log Re-tweet Density

(c) Log Tweeting Count Per User

Figure 4.2: The Densities Versus the Radius in JPEQ

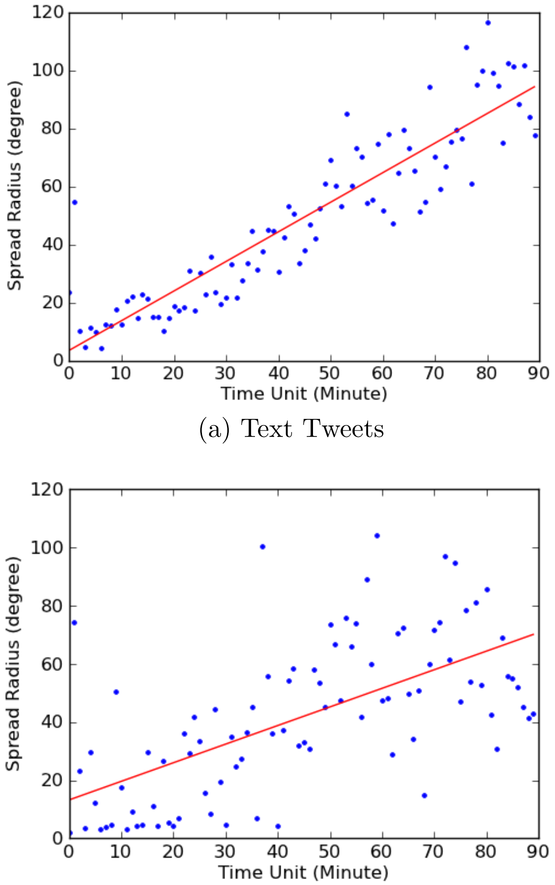
research [40] on the power law decay in intensity of earthquakes. This suggest that tweet density could be used as a proxy for actual seismic readings toward constructing rapid damage assessments based purely on social media content. It is shown that text and media tweets have similar trends, indicating that both are suitable for earthquake intensity estimation.

Figure 4.2b shows the relationship between re-tweet density and distance from the epicenter. Interestingly the re-tweet densities firstly stay stable, then decrease exponentially when the distance from the epicenter exceeds 10 km. With respect to the results from Figure 4.2a, it is known that in the nearest 10 km region, the re-tweet rate (re-tweet density/tweet density) increases with the distance from the center, because the re-tweet density stays constant and the tweet density decreases. This result is consistent with the previous finding in epicenter estimation that people tend to re-tweet more from the (more distant) less damaged area. These more distant re-tweeters are serving as a communications hub spreading the posts from the more direct observers.

Figure 4.2c shows the results that the tweeting count per user versus the distance from the center. Surprisingly, the tweeting count per user is not affected by the distances. For most regions they stay constant, but burst in certain regions. User tweeting counts appear to largely depend on particular population and media centers (e.g., where newspapers and government agencies are located), and so it is unrelated to the radius from the epicenter.

4.4.3 Spread Speed

Finally, the dynamics of tweets are examined: how fast does social media spread in the aftermath of an earthquake? The temporal and spatial features of tweets are integrated to examine the speed of propagation of text and media tweets. For each minute from the onset of each earthquake, the average distance of posted tweets is computed from the epicenter, allowing us to compute the average spread distance versus time. The spread distance in the first 90 minutes is shown in Figure 4.3.



(a) Text Tweets

(b) Media Tweets

Figure 4.3: The Spread Speed of Tweets in JPEQ

For text tweets, it is seen that the spread distance and time are linearly related. As time passes, tweets spread more rapidly in terms of distance. In contrast, that media tweets are less frequent and have a more chaotic spread. Applying linear regression to the two, the correlation coefficients is computed: $r = 0.626$, slope $l = 0.693$ for media tweet, and $r = 0.910$, $l = 1.020$ for text tweet. The correlation coefficient shows that spread distance of text tweets is linearly related to the passing time, indicating a constant spread speed for text tweets. And the spread speed (represented by slope) of text tweets is about 1.020 degrees per minute (about 113.34 km per minute), which is much faster than media tweets, which have a speed of 0.639 degree per minute (71.01 km per minute). Consistent with the results in Figure 4.3, the correlation coefficient of media tweets suggest they are more chaotic than text tweets.

4.5 Summary

Based on this investigation into geo-located text and media tweets in the 2011 Tohoku earthquake and the 2011 Christchurch earthquake, encouraging evidence has been observed. First, it is found that media tweets provide more valuable location information than text tweets, and thus play a more important role in epicenter detection. Second, they both provide comparable evidence of the linear intensity attenuation function for earthquakes, indicating a similar ability to serve as the foundation of rapid damage assessment for earthquakes. The findings of a relationship between social media activity vs. density attenuation suggests that social media following a catastrophic event can provide a rapid insight into the extent of damage to be expected in the field, and that this relationship can then be used to infer the locations of severest damage, as well as where to best deploy emergency response and recovery resources.

5. SUMMARY AND FUTURE RESEARCH OPPORTUNITIES

This section presents a summary of this dissertation and potential future research avenues in this area.

5.1 Summary

This thesis addresses the event modeling problems, focusing on event detection, event tracking and event analytics. Successful event modeling is critical for many services including information search, entity extraction, disaster assessment, and emergency monitoring. Hence, this thesis made three contributions:

The first work is designing a new signal processing-inspired approach for event detection. In this work, an iterative spatial-temporal event mining algorithm is designed for identifying and extracting topics from social media. One of the key aspects of the proposed algorithm is a signal processing-inspired approach for viewing spatial-temporal term occurrences as signals, analyzing the noise contained in the signals, and applying noise filters to improve the quality of event extraction from these signals. The proposed approach is evaluated through experiments on collected Events data sets; the results indicate that the proposed method can effectively remove Event noise, improving event mining effectiveness from social media.

Second, this thesis models the population dynamics of crowds driven by an event or other stimulus. In the proposed population modeling, a duration model is introduced to predict the time users spend in a particular crowd. And then a time-evolving population model is designed for estimating the number of people departing a crowd, which enables the prediction of the total population remaining in a crowd. At last, the crowd models is validated through extensive experiments over 22 million geo-location based check-ins and 120,000 event-related tweets.

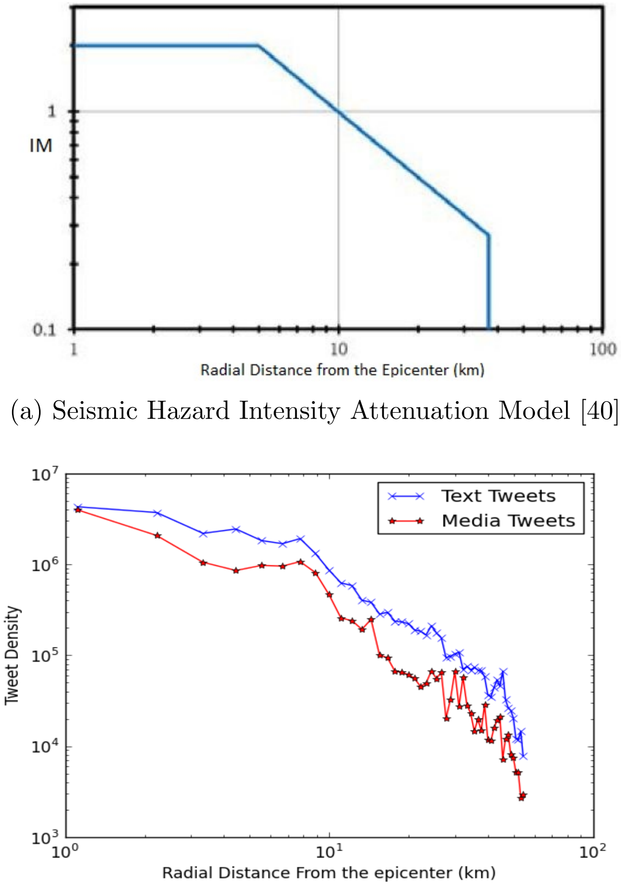
Finally, given the specific scenario — earthquake damage assessment, the potential of social media is investigated for providing rapid insights into the location and extent of damage associated with the earthquakes. Firstly, the difference between text tweets and media tweets is investigated, and then three features – tweet density, re-tweet density, and user tweeting count – are extracted to model the intensity attenuation of each earthquake. The observation that the relationship between social media activity vs. loss/damage attenuation suggests that social media following a catastrophic event can provide rapid insight into the extent of damage.

5.2 Future Research Opportunities

- **Multiple Duration Models**: Section 3 has discussed how to estimate the population for crowds based on duration models. There two types of crowds are examined: (i) event-driven crowds and (ii) location-driven crowds. Future research in this direction can observe other types of crowds like conversation-driven crowds and interests-driven crowds. Besides, although different pdfs have been tested for duration modeling, it was assumed that there only exists one duration pdf for a single crowd. In the future work, multi-duration models will be adopted for a single crowd. For example, when people have a dinner in a restaurant at 6 pm, the time they will spend there would probably be different from that they spend at 10 am, therefore the duration models adopted for these two periods should also be different.

- **Damage Assessment with Architecture Engineering**: Section 4 has discussed event modeling in damage assessment application. In the continuing work, this connection will be investigated through partnerships with structural engineers. For example, [40] builds a 3-d (damage, death and downtime) model to connect seismic hazard intensity attenuation models with loss models, and thus provides a rapid method to estimate the structure damage, downtime and deaths in the earthquake. By studying the relationship of seismic hazard intensity attenuation models [40] and tweet density model in Figure 5.1, can a function be found to map the y-axis values in Figure 5.1a to those in Figure 5.1b?(Figure 5.1a is the attenuation relationship between intensity measure (IM) and the radial distance from the epicenter [40]; Figure 5.1b is the attenuation relationship between tweets density and the radial distances.) If this mapping function can be found, then tweet density model can be connected to the 3-d model directly, hence the losses in the earthquake including damage, downtime and deaths can be estimated directly from social media.

Other future work could be: Can any holes can be found in the coverage of social media due to power outages, lack of population, and lack of access to social media tools? To what extend can the content of media tweets (e.g., images and videos) be incorporated into these models to refine their damage assessments? These and related questions motivate the ongoing investigation into the linkage between social media and traditional methods of post-disaster damage assessment.



(a) Seismic Hazard Intensity Attenuation Model [40]

(b) Tweets Density Attenuation Model

Figure 5.1: Connection between Social Media Models with Architecture Models.

REFERENCES

[1] H. Becker, M. Naaman, and L. Gravano, "Learning similarity metrics for event identification in social media," in *Proc. Web Services Distributed Management. (WSDM 10)*, pp. 291–300, 2010.

[2] Z. Yin, L. Cao, J. Han, C. Zhai, and T. Huang, "Geographical topic discovery and comparison," in *Proc. World Wide Web Conference. (WWW 11)*, pp. 247–256, 2011.

[3] Q. Mei and C. Zhai, "Discovering evolutionary theme patterns from text an exploration of temporal text mining," in *Proc. International Conference on Knowledge Discovery and Data Mining. (KDD 12)*, pp. 198–207, 2006.

[4] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes twitter users: realtime event detection by social sensors," in *Proc. World Wide Web Conference. (WWW 10)*, pp. 851–860, 2010.

[5] M. Okazaki and Y. Matsuo, "Semantic twitter: analyzing tweets for real-time event notification recent trends and developments in social software," in *Proc. Recent Trends and Developments in Social Software. (RTDSS 11)*, pp. 63–74, 2011.

[6] Q. Mei, C. Liu, H. Suz, and C. Zhai, "A probabilistic approach to spatiotemporal theme pattern mining on weblogs," in *Proc. World Wide Web Conference. (WWW 06)*, pp. 533–542, 2006.

[7] J. Allan, R. Papka, and V. Lavrenko, "On-line new event detection and tracking," in *Proc. Special Interest Group on Information Retrieval Conference. (SIGIR 98)*, pp. 37–45, 1998.

[8] Y. Yang, T. Pierce, and J. Carbonell, "A study of retrospective and on-line event

detection," in *Proc. Special Interest Group on Information Retrieval Conference. (SIGIR 98)*, pp. 28–36, 1998.

[9] T. Sakaki, M. Okazaki, and Y. Matsuo, "Earthquake shakes twitter users: real-time event detection by social sensors," in *Proc. World Wide Web Conference. (WWW 10)*, pp. 851–860, 2010.

[10] H. Zhang, M. K., E. Y., and D. J. C., "Beyond co-occurrence: discovering and visualizing tag relationships from geo-spatial and temporal similarities," in *Proc. Web Services Distributed Management. (WSDM 12)*, pp. 33–42, 2012.

[11] N. Garg and I. Weber, "Personalized tag suggestion for Flickr," in *Proc. World Wide Web Conference. (WWW 08)*, pp. 1063–1064, 2008.

[12] B. Sigurbjrnsson and R. V. Zwol, "Flickr tag recommendation based on collective knowledge," in *Proc. World Wide Web Conference. (WWW 08)*, pp. 327–336, 2008.

[13] L. Chen and A. Roy, "Event detection from Flickr data through wavelet-based spatial analysis," in *Proc. International Conference on Information and Knowledge Management. (CIKM 09)*, pp. 523–532, 2009.

[14] Z. Li, B. Wang, M. Li, and W. Ma, "A probabilistic model for retrospective news event detection," in *Proc. Special Interest Group on Information Retrieval Conference. (SIGIR 05)*, pp. 106–113, 2005.

[15] J. Kleinberg, "Bursty and hierarchical structure in streams," *Data Mining Knowledge Discovery*, vol. 7, no. 4, pp. 373–397, 2003.

[16] S. Papadopoulos, C. Zigkolis, and Y. Kompatsiaris, "Cluster-based landmark and event detection for tagged photo collections," in *Proc. International Conference on Multimedia. (MM 10)*, pp. 52–63, 2010.

[17] E. Khabiri, J. Caverlee, and K. Y. Kamath, "Predicting semantic annotations on the real-Time web," in *Proc. ACM Conference on Hypertext and Social Media.*

(HT 12), pp. 219–228, 2012.

[18] Q. Zhao, T. Y. Liu, S. S. Bhowmick, and W. Y. Ma, "Event detection from evolution of click-through data," in *Proc. International Conference on Knowledge Discovery and Data Mining. (KDD 06)*, pp. 484–493, 2006.

[19] Q. He, K. Chang, and E. Lim, "Analyzing feature trajectories for event detection," in *Proc. Special Interest Group on Information Retrieval Conference. (SIGIR 07)*, pp. 207–214, 2007.

[20] A. Ritter, M., O. Etzioni, and S. Clark, "Open domain event extraction from twitter," in *Proc. International Conference on Knowledge Discovery and Data Mining. (KDD 12)*, pp. 1104–1112, 2012.

[21] E. Moxley, J. Kleban, J. Xu, and B. S. Manjunath, "Not all tags are created equal: learning Flickr tag semantics for global annotation," in *Proc. International Conference on Multimedia and Expo. (ICME 09)*, pp. 1452–1455, 2009.

[22] Z. Yin, L. Cao, J. Han, C. Zhai, and T. Huang, "LPTA: a probabilistic model for latent periodic topic analysis," in *Proc. International Conference on Data Mining. (ICDM 11)*, pp. 904–913, 2011.

[23] Q. Mei, C. Liuy, H. Suz, and C. Zhaiy, "A probabilistic approach to spatiotemporal theme pattern mining on weblogs," in *Proc. World Wide Web Conference. (WWW 06)*, pp. 533–542, 2006.

[24] T. Rattenbury, N. Good, and M. Naaman, "Towards automatic extraction of event and place semantics from Flickr tags," in *Proc. Special Interest Group on Information Retrieval Conference. (SIGIR 07)*, pp. 103–110, 2007.

[25] R. C. Gonzalez and R. E. Woods, *Digital Image Processing.* Upper Saddle River, New Jersey: Prentice Hall, 3rd ed., 2007.

[26] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, no. 5814, pp. 972–976.

[27] G. Begelman, P. Keller, and F. Smadja., “Automated tag clustering: improving search and exploration in the tag space,” in *Proc. Collaborative Web Tagging Workshop at WWW. (CWTW-WWW 06)*, pp. 15–33, 2006.

[28] Z. Cheng, J. Caverlee, K. Kamath, and K. Lee, “Toward traffic-driven location-based web search,” in *Proc. International Conference on Information and Knowledge Management. (CIKM 11)*, pp. 805–814, 2011.

[29] M. Ye, D. Shou, W. Lee, P. Yin, and J. Janowicz, “On the semantic annotation of places in location-based social networks,” in *Proc. International Conference on Knowledge Discovery and Data Mining. (KDD 11)*, pp. 520–528, 2011.

[30] F. Bai, N. Sadagopan, B. Krishnamachari, and A. Helmy, “Modeling path duration distributions in manets and their impact on reactive routing protocols,” *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 7, pp. 1357–1373, 2004.

[31] D. Nam and F. Mannering, “An exploratory hazard-based analysis of highway incident duration,” *Transportation Research Part A: Policy and Practice*, vol. 34, no. 2, pp. 161–166, 1991.

[32] R. J. Butler and J. D. Worrall, “Gamma duration models with heterogeneity,” *The Review of Economics and Statistics*, vol. 73, no. 1, pp. 85–102, 1998.

[33] J. Sadik-Khan, “New York City Bridge Traffic Volumes 2010.” http://www.nyc.gov/html/dot/downloads/pdf/bridge-traffic-report-10.pdf, 2010.

[34] B. D. Longueville, R. S. Smith, and G. Luraschi, “Omg, from here, i can see the flames!: a use case of mining location based social networks to acquire spatio-temporal data on forest fires,” in *Proc. Workshop on Location-Based Social Networks. (WLBSN 09)*, pp. 73–80, 2009.

[35] K. Starbird, L. Palen, A. Hughes, and S. Vieweg, “Chatter on the red: what

hazards threat reveals about the social life of microblogged information," in *Proc. Computer Supported Cooperative Work. (CSCW 10)*, pp. 241–250, 2010.

[36] H. Kwak, C. Lee, H. Park, and S. Moon, "What is twitter, a social network or a news media?," in *Proc. World Wide Web Conference. (WWW 10)*, pp. 591–600, 2010.

[37] C. Corbane, G. Lemoine, and M. Kauffmann, "Relationship between the spatial distribution of sms messages reporting needs and building damage in 2010 haiti disaster," *Natural Hazards and Earth System Sciences*, vol. 12, no. 2, pp. 255–265, 2012.

[38] S. B. Liu, L. Palen, J. Sutton, A. L. Hughes, and S. Vieweg, "In search of the bigger picture: the emergent role of on-line photo sharing in times of disaster," in *Proc. Information Systems for Crisis Response and Management. (ISCRAM 08)*, pp. 140–149, 2008.

[39] S. Xu, L. Kong, and Y. Zhang, "A picture paints a thousand words: a method of generating image-text timelines," in *Proc. International Conference on Information and Knowledge Management. (CIKM 12)*, pp. 405–417, 2012.

[40] J. B. Mander and Y. Huang, "Damage, death and downtime risk attenuation in the 2011 christchurch earthquake," in *Proc. Annual Conference of the New Zealand Society of Earthquake Engineering. (ACNZSEE 12)*, pp. 16–23, 2012.