Scholarly Paper:
Modeling Content and Trends in Social Media

Patrick Trinkle

April 19, 2013

Abstract

This work explores the state of the art modeling of social media data for forecasting trends. It explores the current state of the art, which primarily focuses on minimal feature selection and targeted searches. This work achieves limited success with entropy for trend detection, however the data set spareness introduced some difficulties.

Contents

Chapter 1

Introduction

As broadband access and smartphone use has spread over the past several years, more and more people are interacting online transforming the internet to a more user-content oriented environment. This move towards Web 2.0 has spawned several websites providing free online storage for sharing photos and videos. Also, with users accessing the web on their phones; there has a been a shift from traditional online journals or blogs to micro-blogs. The root goal of most of these online services is to encourage and support social networking. These sites include Google+[1], Facebook[2], Twitter[3], and formerly Myspace[4]. Google+ is Google's fourth attempt at starting a social network. Its three previous services are Buzz, Orkut, and Wave. These services not only provide an outlet for online interaction and sharing, they are also a multi-billion dollar industry that leverages personal information for targeted advertisements.

The users of these services post their feelings about movies and products, their interests, their photos, and often their locations. With the increasing usage of smart phones, a large quantity of posts contain global positioning system (GPS) coordinates. These are used by certain services to identify to your friends where you are, to go along with the post saying what you're doing.

At first glance this information seems useless to anyone who doesn't know that person, especially since the goal of this information is to share with friends and family. However, this wealth of information can readily be data-mined as a resource. This includes identifying traffic, not necessarily by word search but possibly by watching the distance between GPS coordinates in posts.

More directly, a user will post what is happening to them in life in near real-time. These posts include photos, videos and text. They act as an intelligent mechanical sensor. In the literature, users are sometimes referred to as social sensors, because of this reporting, which can include earthquakes, rain, bad traffic, or even military action.

Three recent events stand out as interesting uses of social networking services. News of the protests in Iran [35] was heavily publicized by Twitter users experiencing it. The citizens were broadcasting in near real-time what was happening on the ground. A similar experience occurred in Egypt [41]. Also during the US raid on Osama bin Laden in Pakistan, a local citizen tweeted the entire experience [33].

The focus of this work is leveraging information from a specific online social service, Twitter, by modeling user traffic.

[1] http://plus.google.com
[2] http://www.facebook.com
[3] http://www.twitter.com
[4] http://www.myspace.com

1.1 Twitter

What is *Twitter*? Twitter is a rapidly growing micro-blogging[5] service used by hundreds of millions world-wide. It was recently passed in usage by Google Plus[6]. At its most basic level it is a service by which users post information for others to consume. People can subscribe to posts or feeds from each other; similarly to an RSS[7] service.

The Twitter system is comprised of users and messages. The users post messages, known as statuses or tweets. Twitter is a micro-blogging service, which is different from a blogging service. The goal of a status message is to answer the following question: "What's happening?" This is a considerable divergence from the typical long-winded diatribes of regular blogs. Twitter is an Internet based service, which is large scale and multi-lingual. The data is streaming and follows a directed graph structure. Unless specifically disabled, all status messages are publicly accessible. This format tied to the millions of users makes it a strong point of information diffusion [28].

Posting, reading and searching the service is handled through a published API that is leveraged by dozens of third-party applications.

Twitter also provides a trending topic service, by which users can query what topics are considered "hot" by Twitter.

1.1.1 Anatomy of a Twitter User

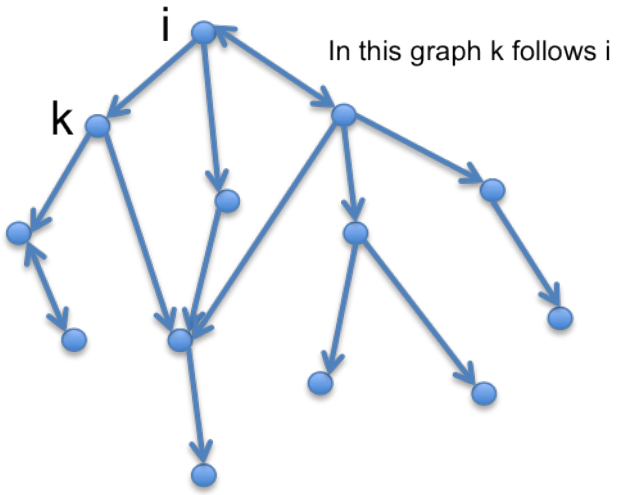


Figure 1.1: Twitter User Graph

Users of Twitter can "follow" other users. When a user_k is following another user_i; user_k receives tweets from user_i. Therefore Twitter users can be viewed as a directed graph, see Figure 1.1. Users can follow each other as "friends," allowing for certain edges to be bi-directional. This directed graph identifies the specific flow of information with the caveat that a user can read tweets from users they are not following. Recent research has identified that homophily, a bi-directional follow relationship, is prominent in Twitter [55].

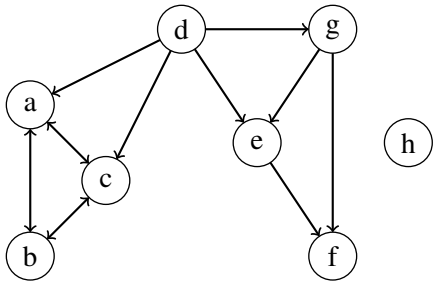
[5] http://en.wikipedia.org/wiki/Micro_blog
[6] http://soshable.com/google-surpasses-twitter-at-least-on-paper/
[7] http://en.wikipedia.org/wiki/RSS

Twitter users can also re-tweet what another user has tweeted previously.

Given the popularity of Twitter, many celebrities and government agencies, like the CDC (Center for Disease Control), have user accounts for information dissemination. To aid in a user's ability to identify real people and agencies, Twitter has a "verified" user badge that can be assigned to an account. Accounts with this badge have been somehow verified to represent the real person or group.

Another way to interpret this graph of users is as a set of agents using a variation of the gossip protocol. Given a graph $G = (V, E)$, such that V is the set of users and E is the set of directed edges between users. The direction of the edge indicates the follow (follower, followed) relationship. The graph of users is also a spanning tree throughout the network of users, whereby the followers receive information from the followed. There is however the possibility of entirely disconnected nodes, where a user follows no other users and no user follows that user, see Figure 1.2, node h. Cliques can also occur within the network, see Figure 1.2, nodes: a, b, c.

Figure 1.2: Twitter Follow Graph



1.1.2 Anatomy of a Tweet

```
id: 10903892834321234
timestamp: Thu Aug 18 2011 +0000 15:12:00
text: I'm walking my dogs today #activities
location: [ 45.678, -35.000 ]
```

A Twitter post, also known as a tweet, can be at most 140 Unicode characters in length. This length limit is severely restrictive in the quantity of information that can be posted at a given time. To overcome this, the user is forced to choose words carefully. This forced deliberation can lead a user into using specific patterns. Even casual users who do not post anything that would be typically considered valuable information should fall into a pattern. Tweets are very similar to SMS messages and as such are highly ungrammatical and fraught with spelling errors.

Each Tweet has a 64-bit ID value, a timestamp, an optional location, and the text of the message itself. Tweets can also contain hashtags[8]. A hashtag is a searchable word or spaceless phrase such as "#activities" or "#winningelections." Messages that contain the same hashtag are automatically clustered together by the Twitter service. Often popular news topics have related hashtags with which users can publicly post tweets attached to the news trend. For instance, Charlie Sheen[9] started a trend of tweeting with the hashtag "#winning." Different hashtags are more popular and stay longer, while others rapidly dissipate. The longer the hashtag stays popular, the more the information seems to diffuse and certain hashtag subjects are persistently

[8] http://support.twitter.com/entries/49309-what-are-hashtags-symbols

[9] http://www.huffingtonpost.com/2011/03/01/charlie-sheens-first-twee_n_830048.html

more popular [47]. Users can post media to Twitter, but this action injects a URL into the message contents that points to a website storing the media.

Starting a tweet with an '@' symbol "directs" the tweet to that user. Having '@' symbols within the tweet is a method of simply referring to other users. If a user is mentioned in a tweet this user is notified, regardless of their relation to the user tweeting. This forced announcement is one of the ways spam has started working its way into Twitter. Users can re-tweet what another user has tweeted. This can be done by reposting their content preceded by "RT" or through an API mechanism. A re-tweet is a re-post by a user of a previously posted tweet [9]. Users can also reply to other user's tweets.

The tweets are stored in unicode, it allows users to post in their native language provided their language is supported in the unicode character set. This ability to post non-English text has grown international support for Twitter, by increasing accessibility and usability.

Users often wish to include a URL in their post. However, URLs are often exceedingly long, especially if pointing to a specific article. In these cases redirectors are commonly used. This shortened URL still occupies a large portion of the available characters in the tweet. This length restriction reduces wordiness, leaving a tweet that may only be "check out LINK."

Also due to the conciseness constraint, proper grammar is often ignored and clever wordings become more prominent. There is even a system which Twitter users can utilize that determines how many of their tweets are exactly 140 characters in length.

In early 2011, a snow and freezing ice storm struck Maryland. The following were posted by the Maryland State Highway Administration (user id 18917699). The list starts with most recent and goes backwards chronologically.

- "Lanes along westbound I-70 at MD 27 are open. Eastbound lanes remain closed." – around 0800 on the 18th
- "Lanes along westbound I-70 are open– eastbound I-70 remains closed."
- "All lanes are closed in each direction along I-70 at MD 27 for a crash. Motorists should use MD 144 as detour."
- "All lanes along I-70 in each direction are closed for an overturned tractor trailer." – around 0700 on the 18th
- "Possible lane closures along eastbound I-70 near MD 27 in Mt Airy. Prepare for delays."
- "The State Highway Administration is treating roads and highways with salt. Log onto www.roads.maryland.gov and click CHART for information"
- "SHA and contractor crews are out patrolling the highways spreading salt. Pavement temperatures are hovering near freezing."
- "At 11 pm SHA shops in Hereford, Hagerstown and Laurel each report over an inch of snow/ice. Easton has seen a changeover to rain." – around 2300 on the 17th
- "SHA Emergency Operations Center is active at 9pm. Snow Emergency Plans are now in effect in Carroll, Frederick and Harford counties."

The highway administration posted this data to help commuters. This is not the first time that useful information was posted for public consumption by the government.

In a similar vein to the detailed traffic information posts provided by the state agency, other users can act as a set of sensors spread over a geographical area that can track weather or local events. Albeit, there are better ways to track meteorological events. Alternatively, consider that the users or sensors could report on their political opinions. If this is the case, pollsters can track local opinions and issues in a more candid and inexpensive fashion. An interesting example of users as sensors was an earthquake in 2011 that was read about in some locations on Twitter moments before it was felt [17].

1.2 Applications of Social Networking (Twitter)

Users around the world are actively sharing information with complex metadata at high speed in a very simplistic manner. This sharing with social networking has allowed a wealth of data to accumulate. This information is mostly peoples' opinions and social activities. It is this growing wealth of data to which I attribute the strong academic interest.

The massive data provides an information set to improve algorithms for handling large indexes and parallel processing. Querying, clustering, and trend detection also benefit from social media. Natural language processing (NLP) can utilize this data to improve trope, idiom processing, and named entity detection. The data itself contains opinions that can be harvested for products and political polls.

There are several interesting applications based in leveraging social networking. For instance, this information can be mined for other users who wish to find other users with similar interests. Also, if a tipping-point can be determined whereby some topic or idea tends to spread at a faster rate, this might be one lead to something becoming popular among Twitter users [19], also known as a trend. Provided this can be accurately modeled; then it might be valuable with other data sets. Given a set of document streams and the model one could predict a point at which some action should be taken.

Parsing the graphs and the data can find authoritative sources of information. The real-time nature of the tweets themselves can detect large-scale events around the world. Tweets about movies and books can provide information to marketers on how well a product is doing. And the tweets provide interesting data on what is popular at the moment, allowing some entity to act.

Four Billion tweets were posted in the first quarter of 2010 alone[10]. Assuming that each tweet is half the maximum data size of 140 characters and 16-bits to represent a character, then in that time frame approximately 560,000,000,000 bytes were posted (521 GiB).

This massive data is considered valuable by enough researchers and historians that the Library of Congress is building a database to store some of it. Specifically they've commissioned an effort to store Twitter data [45]. Tweets from the President, tweets from revolutionaries, are of historical interest and the tweets from the general populace are especially interesting to researchers studying human language. Building a database from the tweets to answer interesting queries is difficult enough that a year later, it is nowhere near ready for access [54].

Jave et al [25], and Zhao et al [58] have investigated why people use Twitter, and in effect how powerful a resource it is. Understanding why people use Twitter explains how it has become so popular and accumulated so much data.

1.2.1 User Behavior

Java et al [25] addressed the problem of the social structure of Twitter and about what people tweet. They detected various communities. Among the communities, a community of individuals with interests in gaming was examined. A large portion of their tweets were related to gaming systems and games, while the remainder was personal feelings and life experiences.

Within communities of interest in Twitter, the users discussed the topic of interest as well as what was taking place in their lives and their personal feelings – even if those feelings are unrelated to the topic of interest.

Zhao and Rosson [58] evaluated the potential usefulness of micro-blogging as a method of information communication in the workplace. They evaluated the usefulness of "water-cooler" conversations among co-

[10] http://en.wikipedia.org/wiki/Twitter

workers in developing collaborative relationships and surveyed a group of Twitter users who work together in a large IT firm.

The surveys revealed that tools such as Twitter allowed the user to "keep a pulse" on what others were doing and feeling. This tool is especially helpful for people those surveyed did not see regularly. The results from the surveys suggested that micro-blogging may help professional and inter-personal relationships. It can improve the overall feeling of connectedness [58].

Twitter users posting what is taking place in their physical vicinity can be leveraged for situational awareness. The users themselves also act to diffuse information.

1.2.2 Information Diffusion

Twitter is also useful for studying information diffusion.

Situational Awareness Vieweg et al [53] collected and evaluated tweets during two concurrent natural disasters for possible contributions to situational awareness. The two events were the Oklahoma Grassfires of April 2009 and the Red River Floods from March to April 2009. They collected data from March 8th to April 27th. Their system only gathered public broadcast tweets containing their search terms through the Twitter Search API. The terms they eventually used were: red river; redriver; oklahoma; okfire; grass fire; and grassfire. Their searches identified 4,983 unique authors for the Red River Floods and 3,852 authors for the Oklahoma Grassfires.

The data set was reduced by requiring at least three tweets with matching terms per author for inclusion of that specific data stream. All the remaining tweets were manually reviewed for relevance and the locations of the Twitter users determined by looking at the user's profile. Their focus was only on those users in the vicinity of the events. They found 49 authors with 19,162 tweets for the Red River Floods and 46 users with 2,779 tweets for the Oklahoma Grassfires. From these tweets they found a significant amount of useful information was being broadcasted by individuals on the ground, who were concerned with the events. Within the Oklahoma Grass Fires data set, 40% contained geo-location information and 18% within the Red River Flood data. These tweets contained information, including the spread of the danger and evacuation information. Tweets that contained situational updates were re-tweeted greater than 12% of the time [53].

This study clearly demonstrates the value of social networks, such as Twitter as information diffusion networks.

Not only is the information diffused through social networking related to situational awareness, it is often news related. Most news services, local and global, have a face in Twitter for spreading information to followers.

Detecting when news stories first appear in Twitter falls under "first story detection." Petrović et al [42] researched a new method specifically for Twitter, which maps the idea of email threads to tweets. This is done by "linking" similar tweets based on their cosine similarity values.

Finding Authoritative Users Lee et al [26] examined the order of information adoption in Twitter to determine which users were authoritative. Similar work in this area has attempted to fit the PageRank [40] algorithm. This algorithm however relies on time to define an authority, through more links to the source existing. This method disregards the breaking edge of the information wave. Also, users with many followers are not necessarily diffusing information, but are merely celebrities. The more users an individual follows, the less likely they are to read all the tweets.

They crawled Twitter from June 3rd to September 25th, 2009 and found 41 million users. The user graph contained 1.47 billion directed edges. Once they had the user graph they collected tweets mentioning

one of the top 10 trending topics (as defined by an internal Twitter algorithm) in five minute intervals. This provided 4,262 unique topics with 223 million tweets. The tweets were then clustered by temporal proximity and topic.

To identify the probable information diffusion in the Twitter graph they defined three types of users. An effective reader is someone who is newly exposed to the information. A potential reader is someone who receives a tweet on a topic. A writer tweets on a particular topic. The influence of a user is the sum of the magnitude of the set of effective readers for each tweet. This is calculated by counting the number of users who transition from potential readers to effective readers given each tweet.

Their [26] model found that the most influential users were news media outlets and that there was very little overlap between their model and the PageRank approach.

Bakshy et al [3] also studied information diffusion within Twitter. Their approach was different in that they identified diffusion events over a two month period as tweets that included bit.ly URLs, and were a single seed point. They observed users who posted these event seeds in the first and second month and then used the first month as the training corpus to try to predict the second month. The goal of their work was to determine whether users could be identified as good seed points for injecting information into the data stream. The user follow graph was used to detect the path of the information that was diffused in the timeframe. Their results maintained that word-of-mouth strategies for triggering information diffusion by injecting data from certain users was not necessarily feasible.

Weng et al [55] developed a new approach to identifying authoritative users also based on PageRank, called TwitterRank. Each user's tweet streams were evaluated for topics using LDA allowing the researchers to locate the range of topics about which a user is spreading information. Given the prominence of a topic within the global topic stream of Twitter, a user's authority is the weighted measure of their topics weights within this stream. Their approach outperformed other similar approaches, but does have room for improvement. Weng et al describe one such improvement as leveraging user mentions and replies.

There has been other research in identifying user influence, especially with the prevalence of homophily in follow relationships and the presence of celebrities. Cha et al [12] explored the notion that having a million followers does not make the user important for information diffusion.

1.2.3 Scientific Linguistics

Twitter data is natural language text with optional metadata. Therefore, it is ripe for textual analysis. Specifically there has been interest in identifying dialects, gender, and context extension. Recent research into Twitter has been able to demonstrate that dialects are apparent in social media, specifically Twitter[11]. There has also been recent work in identifying men vs. women users by the terms they use when posting as well as the subject matter [20].

Due to their short size, tweets often carry insufficient context for normal information retrieval processing. However, there has been work in leveraging other information sources [4]. A large information heavy source, such as Wikipedia, can be coupled with a short text article to assist in querying and clustering. Work with this technique has been applied to among other data sources, to Google News[12] RSS feeds. Google News posts hundreds of articles a day. Users of this massive stream of data suffer a similar overload as Twitter users.

Twitter users who are friends with users of other languages could be identified as boundary crossers or likely bilingual. Users who post messages in multiple languages might bridge interesting region gaps, such as English and Arabic or Arabic and Farsi.

[11] http://www.cmu.edu/homepage/computing/2011/winter/twitter-dialects.shtml

[12] http://news.google.com

Identifying and tracking discourse is an important computational linguistic problem. Ritter et al [46] recently developed an unsupervised model for identifying dialogue structure. This was done with a combination conversation and topic model. The topic model used was based on LDA. Of the 1.3 million conversations in their data set 69% were simply posts with a single reply.

1.2.4 Commercial Uses

Beyond strictly "academic" applications there are significant commercial uses for this wealth of data.

> "Now thanks to Twitcher, morons voluntarily spew out every fact I need to know to exploit them."
>
> *from the Futurama episode "Attack of the Killer App"*

It is no secret that personal information is valuable to marketing firms. The episode of Futurama satired the reality that the personal information people post to social networking sites can be targeted for direct marketing.

As users publish their thoughts in near-real time about movies and products, corporations can harvest this information to rapidly identify trends and predict future earnings [1, 22, 24]. If most of the social networking posts indicate positive reviews for a film it may be worth opening it in more theaters, adding more showings, or leaving it in theaters longer. This may also be indicative of higher future DVD/Blu-ray sales. This analysis is typically referred to as sentiment analysis. Identifying products and locations within tweets falls under named entity detection [32].

Sentiment analysis of social networking traffic is currently being studied for this purpose. Bollen et al [8] have correlated the "mood" of posts with recent news using syntactic approaches instead of machine learning. They argue that small texts, such as Twitter, do not work as well for machine learning algorithms. Other Twitter based research typically concatenates tweets from a user into large time frames, which is likely to overcome any shortcomings in this area.

Other marketing leverages the viral nature of certain aspects of social media. The company Old Spice has had a very successful viral marketing campaign utilizing short videos on Youtube[13] [29].

If a firm wanted to identify whether their product had been recently mentioned in the Twitter stream, there is a public API. If a search operation is performed repeatedly over a period of time it can provide for a reasonable sample set. This sampling should provide an interesting image of the current trends. This method can provide better results than the standard Twitter output for trending topics. Twitter provides information from an internal algorithm, labeled as trending topics. Because of the global nature of Twitter, these topics are ones that tend to impact a larger audience.

A statistical model could be designed such that it identifies topics and terms before they hit a tipping point [2]. The model could learn where key nodes are within the user graph and use sensors on those users. When a topic hits those users it might indicate a new trend. Alternatively, regular queries into the live feed could provide early indications that a topic is becoming more popular. The immediate downside to this second approach is that the live feed queries only provide 20 tweets and it is cached in 60 second intervals. Previous research into trending topics, indicate topical shifts occurring within thirty minute intervals for short-term topics [14]. There is a new Streaming API; but the connections would likely be cut due to throttling.

Beyond social networking posts, users can still be identified through product reviews and other user generated online content [18]. It should also be possible to link users' accounts across many open access websites and further target them with advertisements.

[13] http://www.youtube.com

Grier et al [21] examined the use of spam on Twitter. Twitter is a passive messaging service, therefore spammers have to use varied approaches from email spam. Similar to other Internet spam it includes offers for dieting, free gear, and pharmaceuticals. The goal of the spammers is to lure a user to click on a malicious URL. Twitter has heuristics to fight spam that detect excessive friend requests. Grier et al enumerated five methods spammers use to generate traffic: call outs; re-tweets; tweet hijacking; trend setting; and trend hijacking. Call outs refer to mentioning a username in their tweet. This draws the attention of the user to that tweet automatically. Spam accounts also re-tweet other spam accounts in cahoots. A spammer can hijack another user's tweet, by re-tweeting it and appending or prepending their malicious URL. Spammers can also attempt to cause a trend by creating a significant number of tweets with the same hashtag. A more effective approach is for the spammer to simply post messages with a trending topic. Any users that search for that trend will find the spam messages mixing with benign ones. They determined that 8% of 25 million URLs were spam that redirected to webpages on known blacklists.

The quantity of spam within the Twitter framework has been steadily growing, as the service has become more popular. Therefore, research into identifying spammers within social media has gained interest, including Lee et al [27].

1.2.5 Government Uses

Not only can commercial entities leverage Twitter information for their benefit, governments can also benefit. Once a user of interest is identified, their social community can be enumerated. These other users may also be of interest to the government. These users may communicate regularly about uninteresting things, but within that noise might lay nuggets of information or coded messages.

There are uses of monitoring social networking focused less on domestic and foreign threats. These uses include passively polling public opinion. Currently to identify public opinion trends, phone interviews and in-person interviews are conducted on samples of the population. This active polling is expensive and invasive. Most of the called individuals are likely to decline comment. With a passive system it will only identify the opinions of the users who feel strongly enough to post [39]. This can be statistically weighed to account for the nature of social networking users.

Twitter was watched during the German federal elections and it was found that the number of political tweets was indicative of the likely winner. It was also demonstrated that social networking, specifically Twitter, is used for political deliberations [52].

Along with other commercial news organizations around the world, Twitter is censored in certain countries and has been censored or blocked during political events, such as protests. It has not only been a tool for organizing protests, but also for exporting information out of a locked down region[14].

1.2.6 Nefarious Uses

In the news recently more and more attention is focused on public information people post on social network sites. This includes photos of your home, your vacation plans, and possibly where you live. These services, including Twitter and Foursquare[15] typically allow a user to restrict access to the information, but a lot of users do not bother to use the security features and in ignoring these, they post their information to the public.

Posting when a person is on vacation is an invitation for criminals to burglarize. This is especially true of a recent post that includes a photo of a new television, or a description of an expensive item. For instance,

[14] http://en.wikipedia.org/wiki/Censorship_of_Twitter

[15] http://www.foursquare.com

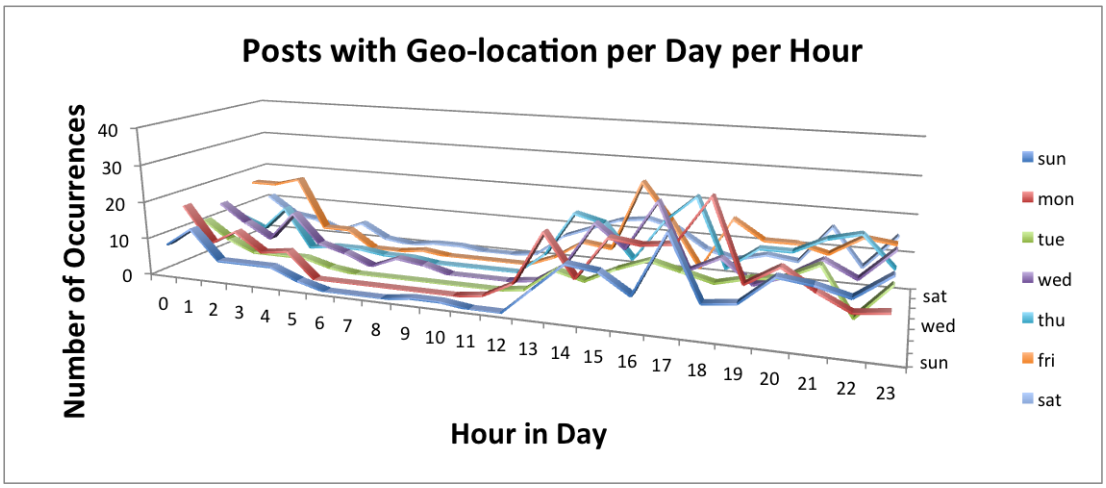


Figure 1.3: Number of occurrences of tweets with geo-location information mapped to weekday and hour.

"just bought the new Samsung 55 TV fantastic."

The danger of posting such information grows by a magnitude when the user has been posting with their geo-location information. Facebook, Google+, and Twitter all allow a user to attach geographic coordinates with their posts. The prevalence of users accessing these services via their smart phones increases the probability that a post will contain geographic markers. Therefore, if the user posts mostly from home, their home address can be exposed. If the user posts from work, this information can be publicly read.

To take full advantage of these geographically marked posts, one could develop a model for identifying specific details about the target user. Starting with identifying a user_0 within their geographic range of interest, who posts with geographic markers, the criminals could start collecting their posts.

Posts with geographic markers exist in time and space. Therefore, they can be used to identify the movements of the user. If most of the posts within a small area occur during the day and a different area in the morning or evening, it is probable that you have identified two locations of interest. These locations are likely user_0's home and office. Textual features within the posts themselves can also aid in determining whether they are from the user's work or home. Some geographic markers identify with a place name, such as a restaurant or business. These features in the traffic can add statistical significance to that address being either work or home. If the place-name information is absent, a query can be performed to see if it comes back with a residential address. These locations may be friends' homes. However, the model could take this into account.

Table 1.1: Possible Model Output

Location	**Occurrences**	**Time of Day**	**Place**	**Probability**
1234 Elm St	253	M-F, Su 5-7, 1800-2000	Home	73.67%
1256 Fake St	53	Sa 5-1200, 1600-2300	Home	26.00%
Unknown	0		Home	0.33%
1 Business Lane	110	M-F 8-1730	Work	89.54%
Unknown	0		Work	10.46%

The criminals could also enumerate the users this user_0 follows. Some of these users are likely real-

world friends who live within the vicinity. This can be determined rather straightforwardly by checking if their user profile lists their location, or if any of their tweets are geographically tagged. There is a possibility any posts that are within the vicinity could have been posted during a visit and the user is actually located outside the search radius. This can be identifying by cross-referencing any listed location with the locations in the posts.

Table 1.2: Possible Friends Locations

User	**Location**	**Occurrences**
aaaa	1234 Elm Ct	35
	4 Infinity Drive	200
	5 11th Street	16
aaab	6 Big Circle	350
	17 Fake Address Ln	1700
	1234 Elm Ct	35
aaac	1234 Elm Ct	60
	12 Business Group Cir	1700
	Prague, Czech Republic	16
	Jersey City	23

Once the target user's work schedule has been identified; further investigation can determine if the user is actually a good target. Similarly, if their posts' geographic information suddenly indicates a foreign country; this may be indicative of a vacation or trip. This can be further verified by the text of the post itself.

Given this model for this user or set of users, inferences could be made about where they were likely to head next throughout their day or their week. Figure 1.3 displays the number of occurrences of geo-location information attached to a tweet mapped by time of day and day of week. This data is a specific user and all times are coordinated universal time (UTC). Although, because the tweets are geo-location tagged, the UTC time could be converted to local time. The figure clearly indicates that during certain hours of the day or night the user is more active.

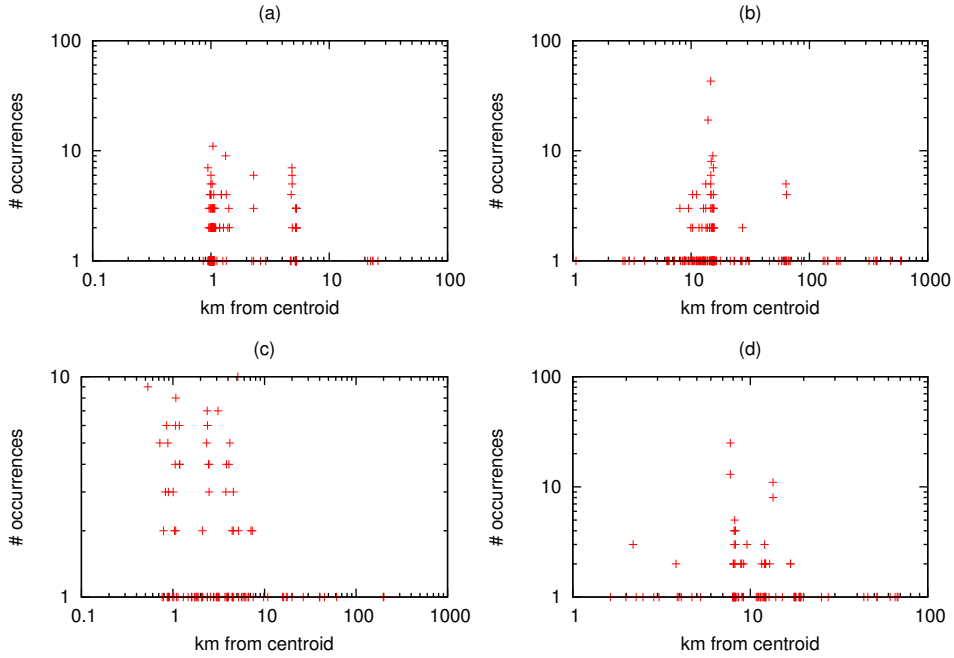


Figure 1.4: Nefarious Geo-location Plots

Figure 1.4 is a 4-plot grid of tweet locations as offsets from a centroid location. Plots a, b, c and d indicate that most posts are within 10 kilometers of their central location. Once interesting users are mapped, further information can be located by plotting a user's tweets' locations directly as gps coordinates, to locate multiple clusters of interest.

An improvement to any basic model for geo-location inference should consider more than the day of the week. It should also consider the month within the calculations. The month in the year can very much impact the location, for instance college students often spend several months of the year in a different location. If the location for that month and day of the week is the same as the location for the day of the week, then there's even more evidence for the location.

$$location(date) = max(\psi(date) \times \theta(date))$$

where $\psi(x)$ returns a matrix of locations and probabilities given month in date and $\theta(x)$ returns a matrix of locations and probabilities given the day of the week and the time therein. If there is no data for a given month or day of the week or hour, the nearest time-wise neighbor is used.

1.3 Problem

> ...imagine that the gods are playing some great game like chess. Let's say a chess game. And you don't know the rules of the game, but you're allowed to look at the board, at least from time to time. And in a little corner, perhaps. And from these observations you try to figure out what the rules are of the game. What the rules [are] of the pieces moving.
>
> *- Dr Richard Feynman*

Dr Feynman's analogy for understanding the physical world is analogous to modeling complex dynamical systems. Social media data is dynamic, massive, has low context, and contains named entities. Modeling the rules and interactions within the Twitter dynamical system is the most promising approach towards addressing the problems with the data set as well as provide an interface to the applications.

Given a massive quantity of data and a list of interesting applications, the challenge is analyzing this data for use addressing these applications. These applications include document classification, trend detection, and streaming data modeling. Social media services have grown very popular over the past few years.

Twitter is one of the top tier social media services. The data is massive, streaming and includes very useful metadata. Modeling this rich data runs into many challenges.

Topic modeling of tweets is difficult due to the short, low-context documents, as well as the constant growth and aging of the corpus. Traditional information retrieval approaches for clustering are either memory intensive by a vast vector space filled with thousands of new documents per second or fall flat. The multilingual nature of the users impacts text processing. The ungrammatical and spelling error laden nature of the tweets interferes with processing.

Modern approaches attempt to build topic models from the documents. However, many models assume the order of the documents is unimportant, including LDA.

Tweets can be thought of as a time-series data set. Each tweet can be represented as a tuple `(who, what, when [, where] [, reply_to])`. Given these time-series, the goal is to effeciently forecast trends. There are many approaches to this, including modeling users in the system in a deep hierarchical model or modeling groups of users or areas at a higher level.

Chapter 2

Current Work

Given the goal of modeling concept drift and hot topics in streaming massive text data, there are a few approaches. A naïve approach would be keyword focused. The approach would simply build a history buffer of keywords sampled from the Twitter stream at regular intervals. The goal would be to identify these "new" terms within the text stream.

A more advanced approach would likely leverage vector space modeling or probabilistic modeling, or possibly a hybrid.

The basic goal of document modeling is to correctly classify documents into clusters under topics. This goal can be extended if you treat tweets or groups of them as documents.

There are a variety of modeling approaches and document representations.

2.1 Vector Space Models

Vector Space Models attempt to map a document into a vocabulary-sized vector space. A vocabulary can be composed of features, which can be words or terms or pieces of words, referred to as n-grams.

2.1.1 Document Representation

Documents are typically represented in a vector space as multidimensional vectors such that each dimension represents a term or feature. This is the bag of words model, where the order of the words in a document is disregarded. The documents are vectors, however the meaning of the distance between the vectors is misleading. To reduce the vector space or term space, stop words are not included in a document vector and only the top m terms or keywords are used. Choosing the top m terms is a problem for feature selection. Stop words are terms that do not provide assistance in distinguishing a documents topic within a language. If these terms are included, under some models they will have very high term weights within a corpus for every topic. Typically, term weight is defined as a value related to the term's occurrence within the set of documents.

For n documents the following matrix represents the vector space model where tw is the term weight for a given term i in document j [34]:

$$\begin{bmatrix} tw_{i,j} & \cdots & tw_{i,n} \\ \vdots & \ddots & \vdots \\ tw_{m,j} & \cdots & tw_{m,n} \end{bmatrix}$$

2.1.2 Binary Vectors

The binary representation of a document is considerably simpler than the Tf-idf variant. Given some the top m keywords, the binary representation of a document is sufficiently described with the following, given t_i is is the term:

$$\forall t_i \in m, tw_i = \begin{cases} 1 & \text{if } t_i \text{ is in document;} \\ 0 & \text{otherwise.} \end{cases}$$

Each document vector is defined as the following:

$$\begin{bmatrix} \{1,0\}_i \\ \{1,0\}_{i+1} \\ \{1,0\}_{i+2} \\ \vdots \\ \{1,0\}_{m-l} \\ \{1,0\}_m \end{bmatrix}$$

To improve upon simply marking terms as present or not in the vector, one could weigh the terms. Functions from linear algebra can be mapped into this context for comparing documents, and other operations. Terms that are found together within documents are likely correlated within the topic and language. Terms that appear topic-free are referred to as stop words in this framework.

Deerwester et al [15, 16] developed a method of text analysis known as Latent Semantic Indexing or Latent Semantic Analysis. This framework leverages the correlation of related terms within a document and the corresponding term weights to answer queries. This framework is considerably more powerful than previous keyword focused information retrieval tools.

2.1.3 Tf-idf Vectors

The Term Frequency Inverse Document Frequency weighting is a method that can be used to build a vector representation of a document and assign weights to the terms. The term frequency used in Tf-idf is the normalized raw term frequency within the document. There are a variety of proven normalization methods, including but not limited to the square root of the sum of squares of term frequencies, or dividing by the length of the document. This normalization is done to balance out the term frequencies in the event of a large document with few occurrences of the term versus a small document with many occurrences. Often the Tf-idf representation of the document as a vector is actually the logarithm of the values for the term frequency and the inverse document frequency. This reduction is used to dampen spikes in the data. Given a term i in a document in a corpus of size N the term weight, tw, is defined as follows [34]:

$$\begin{aligned} tw_i &= tf_i * idf_i \\ tf_i &= \log(termfrequency_i + 1) \\ idf_i &= \log\left(\frac{N}{documentfrequency_i + 1}\right) \end{aligned}$$

The addition of 1 to the value provides that you will never divide by 0 or attempt to calculate the logarithm of 0.

2.1.4 Related Research with Vector Space Models

Vector space models have been leveraged for text categorization and more specifically topics.

Text Categorization Sriram et al [51] investigated a method of automatically classifying incoming tweets to reduce the raw data overload. This is similar to automatically labeling emails or blogs, but with significantly less data.

Users are overwhelmed with the raw data in Twitter. Therefore, it would be useful if incoming tweets could be automatically categorized. Their approach more specifically categorizes the tweets into a set of pre-defined groups. Because of the brevity of tweets introduced by the format, they are difficult to automatically classify – they lack extra context. The pre-defined categories are as follows: news (N); events (E); opinions (O); deals (D); and private messages (PM). Each tweet is evaluated for 8 features: the author; and seven binary features. The following binary features indicate the presence of something: shortened words or slang; time-event phrases; opinion words; emphasized words; currency and percentage symbols; @*username* in the beginning; @*username* somewhere within the text. Their classifier is greedy in that it classifies the tweet as whichever category it fits into first, or as whichever category it fits into most and each tweet only falls into one category. It is unclear which greedy approach was used. They leverage that authors' tweets tend to fall into the same category over a period of time. For instance, most tweets from the State Department are likely news. More specifically news tweets likely do not contain shortened words or slang. Time-event phrases, such as "around 6 o'clock" are indicative of a tweet related to an event. A pre-defined word list of approximately 3,000 words is used to identify opinion words. Words can be emphasized through the use of uppercase letters or extraneously repeated letters. A tweet advertisement likely contains currency or percentage symbols, such as "5% off" or "100£." By starting a tweet with @*username*, this tweet is more or less directed at that user. If a user does this, the mentioned user will receive notification of the tweet regardless of their relationship in the graph.

To build the training set for the classifier they collected recent tweets from randomly chosen users. Tweets that were not in English, contained fewer than three words, or too few words other than a URL, were removed. This left 5,407 tweets from 648 users. The tweets were manually categorized. Stop words were pruned, leaving a vocabulary of 6,747 terms. Sriram et al, performed the experiments with the Naïve Bayes classifier in WEKA[1] with 5-fold cross validation. In other words, they partitioned the data into five sets, four of which are used as training sets for five runs. The results of the runs are averaged.

Their system was an improvement over using a strictly bag-of-words model and classifier. However, because news tweets may contain opinion words, there was error here. The authors felt that an improvement could come in the form of allowing a tweet to be assigned to multiple categories.

This work was interesting and fairly effective at clustering tweets in a user's inbox. However, the use of set partitions limits its growth. Also, the wealth of features available that are not utilized could provide better results. Because their goal was not necessarily to cluster the tweets, but rather to categorize them, they did not need to worry about growth. An improvement would be to sub-classify from within the categories. Given the set of tweets that have been identified as likely news, they could be further classified into types of news such as sports, entertainment, and more generally world news.

Emergent Topics Identifying what topics are becoming important to a community is useful. These topics may relate to local politics and opinions. Public opinion often steers funding. If the community is a set of would-be terrorists; their current topics and interests could be very informative of their intentions.

[1] http://www.cs.waikato.ac.nz/ml/weka/

Twitter is a very low level source of news and information. At its best it is a source of the most current events. Therefore it is useful to identify what topics are emerging as important to users given some interval. Cataldi et al's [11] approach assigns energy values to terms based on the term's weight within the corpus; the freshness (or newness) of the term (how much it appears now, versus in the past); and the authority of the user who used the term. A topic is composed of semantically related terms. These topics are eventually placed into a graph to visually represent the energy of the topics based on the energy values of the comprising terms.

Each tweet is represented as a sparse vector.

$$t\vec{w}_j = \{w_{j,1}, w_{j,2}, ..., w_{j,v}\}$$

The weight of the term x in the k^{th} tweet: $tf_{j,x}$ is the frequency of the term in the k^{th} tweet; tf_j^{max} is the maximum term frequency in the tweet.

$$w_{j,x} = 0.5 + 0.5 \cdot \frac{tf_{j,x}}{tf_j^{max}}$$

The authority value for a user is based on the PageRank [40] algorithm. The general notion is that the more people follow a user, the more authority the user has. Albeit more celebrity is more likely accurate. They calculate the authority value as follows:

$$auth(u_i) = d \times \sum_{u_j \in follower(u_i)} \frac{auth(u_j)}{|following(u_j)|} + (1 - d)$$

d is the dumping factor, or the probability that a user will move from one user to another. Typically set to 0.85 [40]. The authority values are initialized to $\frac{1}{|users|}$.

Each term within the time interval is given a nutrition value:

$$nutr_k^t = \sum_{tw_j \in TW_k^t} w_{k,j} * auth(user(tw_j))$$

The change in nutrition of a term given a specified interval s is used to calculate the energy value of the term. They define a term as emergent if it is used extensively in a given time interval, but not previously. This factors into the following equation:

$$energy_k^t = \sum_{x=t-s}^{t} \left(((nutr_k^t)^2 - (nutr_k^x)^2) \cdot \frac{1}{t-x} \right)$$

where $0 < s < t$.

Given the energy values for the terms, a threshold can be defined by which any term whose energy value is below the threshold is dropped. This value can be determined without user intervention.

The goal of their work however, was to identify emergent topics. Therefore, they build a correlation vector to determine which terms comprise the topic. This correlation vector attempts to use co-occurrence as evidence that the terms are related. Negative evidence of a relationship is the terms not appearing together.

$$rank_{ET_z^t} = \frac{\sum_{k \in K_z^t} (energy_k^t)}{|K_z^t|}$$

Their work is novel in assigning energy values to terms based on the authority of the user posting the message. This is an interesting approach, however many trending topics are likely tweeted by many users whose authority values are very low. Equally, a user with a high authority value may be someone who follows a trend instead of starting one. Also, the lack of co-location of terms within a tweets of the same topic may be disadvantageous as negative evidence.

2.2 Topic Models

Beyond vector space models for document indexing, there are also topic models.

A topic model is a probabilistic model focused on identifying the topics in documents in a corpus. "Introduction to Probabilistic Topic Models" by David Blei [7] is an excellent introduction to this subject.

> "if you have a document that you believe is exchangeable, i.e. if you move the right words around, i.e. bag-of-words model. you don't mean i.i.d. alright, think if I see the first word is an Italian word, I think it's very likely the second word is italian too, and if they were i.i.d. that wouldn't be the case"
>
> *Michael Jordan on De Finetti*

2.2.1 Related Research with Topic Models

Phan et al [43] addressed the problem of short text classification by leveraging a universal data set. Specifically, they theorized that a classifier could be trained with a large data set that is a superset of future topics in the short text corpus. By manually identifying the large data set to use with the classifier it is a semi-supervised technique. To identify the topics within the universal data set they used LDA with Gibbs sampling. This classified data is fed into a MaxEnt [5] as a training set. For evaluation Wikipedia was used as the universal data set for Google web search results and MEDLINE was used for disease classification from medical abstracts. Their technique is limited by the required relevance of the universal training data set. Requiring a priori training is also a disadvantage to any good topic classifier.

Event Tracking Lin et al [30] attempted to tackle the problem of statistically modeling the popularity of a given event in Twitter, and DBLP. They defined a graph for each time period of users where the edges represent connections between users in the same period. They also define a stream of documents as all tweets published in the same time frame. The documents are represented with the bag of words model and are built by concatenating all tweets published by user$_i$ on each day. An event is defined as a stream of topics and at each point users have a level of interest in each event. To verify their resulting model it was compared against a few others including the contagion model. An important assumption they make is that a user becomes interested in an event and then generates discussions about the event. The contagion model states that each user is infected if the number of friends who are infected goes beyond a threshold. The researchers concatenate the tweets into full-sized documents per user per time period.

More specifically their research defines a unified model for measuring an event, interest, and documents. For the following definitions, $k \in [1...T]$. Given a network stream, $G = \{G_1, \cdots, G_T\}$, such that $G_k = \{V_k, E_k\}$, where each user is a vertex and each edge is a connection between two users. It is a complete graph, and $g_k(i, j)$ is the strength of the connection. Documents are similarly modeled in a stream, $D = \{D_1, \cdots, D_T\}$ such that $D_k = \{d_{k,1}, \cdots, d_{k,N}\}$. D_k contains all documents published in from $k-1$ to k, more specifically $d_{k,i}$ is the document published by node $V_{k,i} \in G_k$. $d_{k,i}$ is a bag-of-words frequency based representation of the days' tweets for the user, such that $d_{k,i} = \{c(d_{k,i}, w_1), \cdots, c(d_{k,i}, w_M)\}$ where

$c(x, y)$ is the count of term y in document x. The vocabulary is a fixed set $W = \{w_1, \cdots, w_M\}$. Topics are represented similarly to LDA, as a multinomial distribution of words. However, the topic distributions, θ, can vary over time similarly to the other streams represented. $\{p(w|\theta)\}w \in W, \sum_{w \in W} p(w|\theta) = 1$. Events are represented as a stream of topics, $\Theta^E = \{\theta_0^E, \theta_1^E, \cdots, \theta_T^E\}$. θ_0^E must be provided, and each event, $[1...T]$ in the stream is just a version of the event. Interest is modeled as a real value $h_k(i) \in [0, 1]$, $H_k = \{h_k(1), \cdots, h_k(N)\}$ such that i is also an index in V_k.

Their work leverages three assumptions about users' interests: users with strong connections influence each others' interests; interest does not change dramatically in a short time; and the higher the interest, the more likely the user will post on topic. These assumptions are factored into the factorization of the probability equation to solve for the posterior, $P(H_k, \Theta_k|G_k, D_k, H_{k-1})$. It is also assumed the previous interest levels (H_{k-1}) have been provided as an input to the model. Leveraging the assumptions, you can split the posterior into two portions: the interest model and the topic model:

$$P(H_k, \Theta_k|G_k, D_k, H_{k-1}) = P(H_k|G_k, H_{k-1}) \cdot P(\Theta_k|H_k, D_k)$$

They compared their models design to the state automation model and the contagion model and found them comparable. For complexity analysis, they compared their work to pLSA and also found the complexity within similar bounds.

Their novel unified probabilistic model is a step in the right direction for modeling streaming documents that are generated by social media. Some limitations of this work are its reliance on outside assistance for identifying events. Also, topics within Twitter data tend to change far more rapidly than daily and do not necessarily return – violating one of their assumptions.

Author-Topic Rosen-Zvi et al [48,49], made an interesting improvement to the modern LDA topic model by including author information. Their model captures the interests of the authors, based on the topics about which they've written. Most previous author modeling has been aimed at the authorship attribution problem. This author-topic model can assist in authorship attribution by the distribution of topics within a document correlating to the distribution of topics for each author. However, that was not the goal of their work. In this model each word in the document an author is sampled, uniformly randomly. Then a topic is chosen from the distribution of topics for that specific author. From that topic distribution of words, the word is chosen. The data sets they used for experimentation were a collection NIPS papers and CiteSeer abstracts. An application of this work is identifying reviewers for conference papers, as well as identifying possible future collaborative efforts.

Streaming Documents One of the immediate problems with most topic modeling algorithms is that they are computationally expensive to update. More specifically as new documents enter the corpus, especially with new vocabulary and unseen topics, the model built cannot accurately classify them. With the growing prevalence of small documents posted by mobile devices, algorithms need to handle online updates to the model. More pointedly, the model needs to evolve. Yao et al [57] focused on improving a common sampling method used for state inference, Gibbs sampling. They found that MaxEnt produced reasonable results, but for improved accuracy they built SparseLDA which is faster and has reduced memory usage.

Spatiotopic Mei et al [36] utilize geographic information in weblogs to associate locations to the topics of the blogs. Their process is to first identify themes (topic defined as a unigram language model) from the corpus of blogs. Then they compute a series of themes split into time intervals for each location. Three data sets were used: "Hurricane Katrina," "Hurricane Rita," and "iPod Nano." Their work is very interesting

in that it can identify weblog trends over time across a country. This includes identifying the different viewpoints pertaining to the same events. This is especially interesting when some of the viewpoints are from victims from a natural disaster, while other viewpoints relate to the fallout or aftermath.

Microblog Topics Hong et al [23] empirically compared traditional tf-idf against LDA and Rosen-Zvi et al's author-topic extension to LDA. Their work was the first study to compare topic models used against the Twitter data set. They found that although the models were very effective on short text documents, that given a sufficient corpus, tf-idf provided the best performance. Their tweet corpus was cleaned removing any non-latin characters, any user mentions, any URLs, and converted entirely to lowercase. Their work focused on capturing tweets from "verified" users and tied the category of the user within Twitter (their recommended user system) to the user's tweets.

Labeled LDA is a semisupervised version of LDA. Ramage et al [44] applied this to Twitter data, but the main focus of their work was to identify the essence of tweets. What makes up a tweet? They managed to break tweets into *substance, social, status, and style*. Their work also demonstrated that hashtags could provide a set of labels that rotates as Twitter traffic trends.

2.3 Language Models

A language model is a probabilistic model that attempts to represent a document as a probability sequence determined from the words within the document. For instance, given a document X that contains the terms W, the probability of this document can be determined against a language model.

In its most basic form it is a unigram model, see Equation 2.2. Although this representation is sufficient for many problem sets; an N-gram model is more powerful. An N-gram model represents the probability of a document as the product of the probabilities of each n-terms in the sequence. Leveraging the Markov property[2], the probabilities carry forward as the term pairings are evaluated in a Bigram model:

$$p(\mathbf{w}) = p(w_i|w_{i-1})p(w_{i-1}|w_{i-2})p(w_{i-2}|w_{i-3}) \cdots p(w_{i-n-1}|w_{i-n})$$

If a model is built from a learned corpus or an online process, any n-grams that have never been encountered previously will have 0 probability of occurring. To avoid this problem, smoothing is required.

Chen et al [13] evaluated several traditional smoothing methods for language models, unigram, bigram, and trigram. They found Katz and Jelinek-Mercer smoothing performed consistently with varied parameters. However, this study was done well before the take-off of social media which provides for a very different data set. Similar work has been recently performed with Twitter data [31].

To improve effectiveness of statistics in language models, metadata can be extracted with the term features. More specifically, the proximity of terms within a document can indicate how related the terms are. There has been research in exploiting term proximity to improve the effectiveness of language models without unduly increasing computational complexity [38,59]. Due to the short nature of tweets, the proximity of terms within them may not necessarily be important, unless it is a named entity. For instance, "Great Wall" likely is not similar to "Great! Let's build a wall."

2.3.1 Related Research with Language Models

Smoothing Lin et al [31] built an online language model to track topics in Twitter. Their primary focus however, was on evaluating various smoothing techniques. An effective smoothing technique is important in

[2] http://en.wikipedia.org/wiki/Markov_property

language modeling. They evaluated absolute discounting, Jelinek-Mercer smoothing, bayesian smoothing using Dirichlet priors, and stupid backoff [10]. They found that the simplest approaches were the best.

A disadvantage to their language modeling techniques were the assumptions they made, which included relying on identifying an initial topic by hashtag, and relying on a pre-defined set of hashtags. However, their use of both a foreground and background model was effective and was not memory intensive.

They *normalized* the "stupid backoff" approach. Let $P_B(w)$ be the probability of term w in the background language model; $c(w;h)$ is the count of term w within their history. The probability from the background model is scaled by α.

$$P(w) = \begin{cases} \frac{1}{1+\alpha} \cdot \frac{c(w;h)}{\sum_w c(w;h)} & \text{if } c(w;h) > 0; \\ \frac{\alpha}{1+\alpha} \cdot P_B(w) & \text{otherwise.} \end{cases} \tag{2.1}$$

2.4 Time-series Physical Events

Documents that are associated with timestamps and geo-location information are a time-series. If the users act as sensors, then the time-series data can be used to detect and track physical events. There has been research into identifying and tracking physical events with Twitter as the sensor base, but it has not moved into a possible augmentation of real sensors and RADAR. However, in 2011 Twitter users received earthquake tweets before they felt the ground shake [17].

Targeted Physical Event Detection

Sakaki et al [50] refer to Twitter users as social sensors, because the user's job is report what is happening to them in their lives. This idea that each user is a sensor is important for two reasons. Firstly, a user should tweet sensory information, what the user is seeing or feeling at that moment. And secondly, that sensors can report false data.

Tweets often have geographic information, which can be correlated with the tweets to locate and track large events. This work tracks earthquakes and tsunamis in Japan, because these are fairly common events and the density of Twitter users in Japan is very high. A training set of tweets that have been manually determined to refer to earthquakes was built. And a negative training set with tweets which may reference "shaking" was sent through a support vector machine. New tweets are fed into the SVM and have been proven to locate earthquakes in Japan. Their model was built on the assumption that at any instance there is only one event. More specifically there could only be one earthquake in Japan at a time. This limitation hampers their model's ability to identify and track smaller events that are more frequent (e.g. traffic jams).

They broke the tweets into three groups of features. They calculated statistical features (A): number of words, position of query word within the tweet. In this case the query word was the search term for the event, such as "earthquake" or "shaking." They also captured keyword features (B), the words themselves, and word context features. The word context features (C) include the words before and after the query term. This is how context of a word is determined in natural language. For Japanese tweets, they used Mecab[3] for morphological analysis. In English they used stop word elimination and stemming. Stemming is the process by which a word is broken down into its root by removing any suffixes. This allows multiple versions of the same word to map correctly to the same word and meaning (e.g. mapping the word swimming to the verb to swim).

[3] http://mecab.sourceforge.net

Probabilistic models were built to detect events and then estimate their location or origin for earthquakes, and trajectory for typhoons. Given tweets classified as positive by their SVM, they were fed into the temporal model. The sensors are i.i.d and the the probability that a user detects an event at time $t = 0$ and posts from t to Δt is λ. n_0 is the number of sensors at time 0 and $n_0 e^{-\lambda t}$ sensors at time t. The probability that all n sensors return a false alarm is p_f^n, therefore $1 - p_f^n$ is the probability of an event occurrence. The expected number of sensors at time t is $n_0(1 - e^{-\lambda(t+1)})/(1 - e^{-\lambda})$. The probability of an event at time t is:

$$p_{occur}(t) = 1 - p_f^{n_0(1-e^{-\lambda(t+1)})/(1-e^{-\lambda})}$$

Once an event is detected by the temporal model, the geo-location information associated with the tweets is passed into their spatial model. For evaluation purposes they attempted this process with both Kalman, and Particle filters. The goal of the spatial model was to identify the epicenter of the earthquake or the trajectory of the typhoon.

After running various experiments they determined that of the features selected, categories B and C provided little to assist the SVM. The spatial models were compared against the baseline of the weight average and median of the latitude and longitude of the tweets as well as the ground truth. Kalman filtering was found less effective in their model, likely due to the non-linear non-Gaussian nature of the physical phenomenon involved. They also found it was difficult to locate the epicenter via tweets if the center was out in the ocean or in a low density population area – for obvious reasons.

Their SVM to temporal model worked very well in identifying the events, however the spatial model did not perform as well as would be required for an effective predictive system.

Similar to their research, during the evaluation of the comprehensive topic modeling in my research, the features extracted and measured will be compared for effectiveness. The fewer features used with the most impact provide the best results, without wasting computation and memory. The goal of my research is not to track or identify physical events, however because the data set also includes geographic information, either Kalman or Particle filters may become involved.

2.5 Probabilistic Models

A vastly different approach to modeling documents strictly as numeric vectors is based on probability theory and machine learning. In its most basic form, terms that often appear together are probably related. Vector space modeling attempts to assign a weighted value to this relation and has methods for exploring this theory. However, probabilistic modeling begins with this theory and builds from there.

The most basic probabilistic document model is a fully independent unigram model, such that each word within the document comes from a single multinomial distribution:

$$p(\mathbf{w}) = \prod_{n=1}^{N} p(w_n) \tag{2.2}$$

An application of probability modeling is classifying documents in a corpus. One approach to this is Naïve Bayes.

2.5.1 Naïve Bayes

The Naïve Bayes classifier follows Bayes theorem. Given a set of terms, the probability of them appearing in a specific class is based on all the conditional probability of any of the terms occurring. Naïve Bayes simplifies this by stating that all term probabilities are independent [56]. Without the assertion of independence

the probability calculations would complicate exponentially as terms are added to the feature set. Given a document d, a class c, a term t_k, and n_d is the term count for d, the probability is defined as:

$$P(c|d) \propto P(c) \prod_{1 \leq k \leq n_d} P(t_k|c)$$

2.5.2 Graphical Models

Graphical models are a method of representing a probabilistic model with a graph.

Given a joint probability distribution over random variables or groups of random variables, a graphical representation can assist in expressing conditional relationships. The factorization of Bayesian networks are represented as directed acyclic graphs, where the following holds [6, 37]:

$$p(\mathbf{X}) = \prod_{k=1}^{K} p(x_k|pa_k)$$

such that the graph has K nodes, $\mathbf{X} = \{x_1, \cdots, x_K\}$ and pa_k is the set of parents of node x_k.

To compactly represent repeated nodes a plate notation is used, see Figure 2.1. Nodes that are shaded are observed.

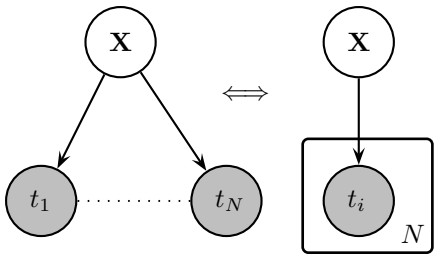


Figure 2.1: Compact Representation using plate notation.

Chapter 3

Methods & Results

3.1 Introduction

Academic interest in textual analysis and more specifically classification and trend detection has gained momentum since the movement of the Internet towards streaming documents. Work in this area has evolved over time from basic bookkeeping to complex graphical models. The massive quantity of this content has provided new challenges in categorizing the text as well as detecting trends.

This work includes basic text categorization approaches that leverage binary features within tweets [51]. Cataldi et al [11] attempted to leverage a calculated user authority value added to the weight of a term to detect new trending topics. Lin et al [30] approached the problem of tracking topics within a set of users by building a model that combined an interest model and a topic model. Another interesting approach was the spatiotemporal model developed by Mei et al [36], which hypothesized that different geographic regions would have different trending topics.

LDA inference has also been examined to better support streaming documents [57]. Several other models have been designed to work on this problem. Rosen-Zvi et al [48, 49] extended LDA by adding the notion that different authors had different distributions over the topics. To overcome the low-context nature of some social media there has been research in leveraging outside databases to provide context [4,43]. With the various models, an empirical study has been performed to identify the best performer when the approach is specifically targeted toward Twitter data [23]. This study included tf-idf, LDA, and the Author-Topic model.

Language models are another set of approaches for textual analysis. A significant portion of this work pertains to smoothing the model [13, 31]. This is necessary because any learned model will have gaps for either words of phrases not yet encountered. An interesting improvement upon the basic language model would be the addition of textual metadata such as term proximity [38, 59].

The significant quantity of previous work in the field of text categorization and trend detection has largely been motivated by fairly straightforward applications. These applications include identifying, tracking or following user opinions detailed in documents, blogs and social media [8, 25, 32, 39, 52, 58]. Identifying and following opinions is an application of the sub category of textual analysis known as sentiment analysis as well as the social science of political analysis.

Tracking information diffusion by identifying authoritative users is another application. This also includes tracking information diffusion as it pertains to geographic or geopolitical situational awareness [3, 12, 26, 28, 42, 53, 55]. Textual analysis can also be leveraged to identify topic trends. Focusing the topics queried onto products such as new films and aiming the text analysis to social media, it can be used

to predict future earnings [1, 2, 14, 22, 24, 29]. Identifying user gender, detecting conversations, and labeled users by probable dialects are additional applications [20, 46]. Analyzing social media text can also be used within to detect and track certain physical events [50].

For either intelligence or nefarious purposes, modeling the text generated by users, specifically from social media outlets can aid in modeling the user's life patterns.

Given the rich and massive streaming dataset of tweets, the text analysis problem of modeling topics is non-trivial. There have been several previous attempts using either language models or topic models. However, the topic models (excluding the author-topic model) focus on the text of the documents instead of the valuable metadata. Another challenge with social media is that posts are length limited. Due to this limitation however, an assumption carried from topic modeling is the likely effectiveness of the bag-of-words model.

The current approaches to the modeling of social media content fall short. Given the previous comparison that demonstrate the shortcomings of using LDA on this content [23], I conjecture that adding metadata beyond the author [48, 49] of the document will improve modeling.

Beyond adding the metadata to the model, the structure and amount of influence between the metadata needs to be learned. The metadata which is accessible across Facebook, Google Plus, and Twitter includes the geo-location information, the source, the time, and their friends.

If the source changes between posts, then the user has likely shifted from either a mobile position to a stationary one and this may indicate a high chance of a topic shift. Posts that are very close together in time may be more likely similar. Posts that rapidly follow a post from a friend in the social network may be a response or comment to that post and as such may share the topic. The location of a post, such as the gps coordinates may indicate a change in location, such as posting from work about work and then posting from home about more social matters. The recent posts of the user's friends in the network may also influence the user's posts in general. Also, Facebook, Google Plus, and Twitter support mentioning a user within a post. This draws the attention of the user mentioned. If the mention is from a friend it may influence their future posts in the near term. Users can also "reply" or post onto posts from other users. This mechanism is one form of online dialogue.

Given the previous research, the following hypotheses fall out. If a term appears in multiple locations, or models from multiple locations that are similar, are the sensors behaving similarly? As varied sensors report the same data, is a trend forming? Can these trends be detected by measuring variations in the model through entropy? Can metadata be leveraged to improve modeling? Many of the previous successful research attempts to forecast trends by limiting their focus to key features or specific terms, such as "earthquakes."

To verify or disprove any hypothesis, you must first build a data set for testing. I collected tweets from two geographically distinct areas: Boston and the Washington DC, I-495 corridor. The boxes used for collection were approximately the same size. From Boston, 113,112 tweets were collected while 204,987 tweets were from I-495. Collecting from multiple points was required to check if there was correlation between multiple locations. The collection started on the 10th of August 2012 at 20:19:23 and ended on the 15th at 08:15:34. All non-visible characters, newline characters, and tabs were replaced with spaces. The text was converted from Unicode to ASCII. Any URLs and user mentions were stripped from the text. All the tweets were dropped to lowercase. Also, all punctuation was also converted to spaces.

For experiments, unigram language models were built for intervals over the window. The models were built by concatenating the tweets within the interval and by excluding words smaller than three letters, words in a stop list, and words in a global singletons list. The singletons list was built from a single-pass over the entire window. The 458 stop words used were chosen as a mixture of previously built lists and terms found in this data set with dramatically high term frequencies that were human verified as providing no contextual information. In the data set, 71,525 singletons were identified. The remaining dictionary size is 49,417

terms.

The interval length used to build models was varied to increase or decrease the number of tweets put into the model. Also, a sliding window variation was applied whereby there was overlap in the tweet input between neighboring models. Each model for the interval was independent of the previous and later models, which prevented a term from spiking too high and not trailing off fast enough. Anything that was interesting within that interval should still stand out. For the most part the following intervals (in seconds) were used: 225, 450, 600, and 900. For certain experiments, a hidden-markov variation was used.

Once the data set was in place, a variety of techniques were applied to the data in an effort to determine effectiveness of some basic approaches.

3.2 Term Growth and Distinct Terms

Given that we have global knowledge of the data input over the window, we can map the growth in new terms per vector. If the quantity of new terms encountered each interval never drops, then any term based probabilistic modeling attempt will not work well as each vector will occupy smoothed terms.

Figure 3.1 and Figure 3.2 show the number of new terms encountered per each interval and the percentage of each new interval that is new. The percentage of new terms should provide more insight into the change, as a very small vector could only have a few new terms, but that could represent a large percentage of terms.

The models were processed as hidden-markov models for this experiment, so that the new terms encountered are globally new.

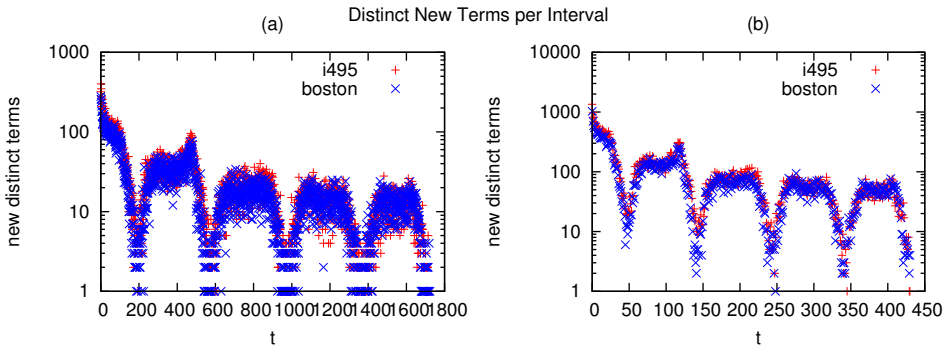


Figure 3.1: Distinct New Terms per Interval: a) 225s b) 900s

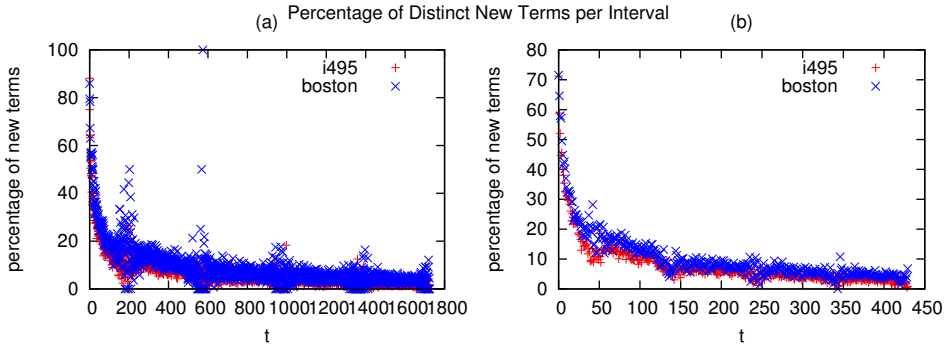


Figure 3.2: Percentage of Distinct New Terms per Interval: a) 225s b) 900s

The percentage of new terms doesn't significantly drop, but rather levels out around 10%. Therefore, any vector-based unigram language model would require regular smoothing. These missing terms typically discount the probability of the next interval existing given the previous. Most of my experiments did not treat the model generated per interval as a hidden markov model, but for the new term examination they effectively were.

Sliding Window Variation

To increase the similarity between neighboring models, half step models were built. They took the form of sliding windows over the time window, versus strict slicing.

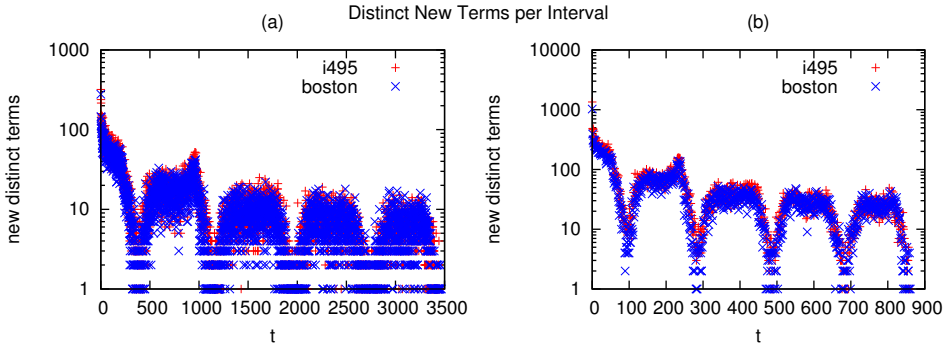


Figure 3.3: Distinct New Terms per Interval,
using a half-step sliding window: a) 225s b) 900s

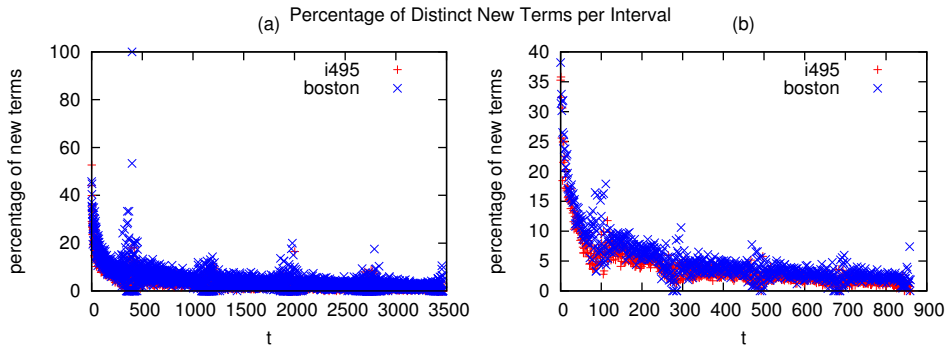


Figure 3.4: Percentage of Distinct New Terms per Interval,
using a half-step sliding window: a) 225s b) 900s

The model size and points of interest tend to follow the plot of the distinct terms per model over time. These both trace a period based on the time of day. The number of tweets collected per hour clearly demonstrates the period.

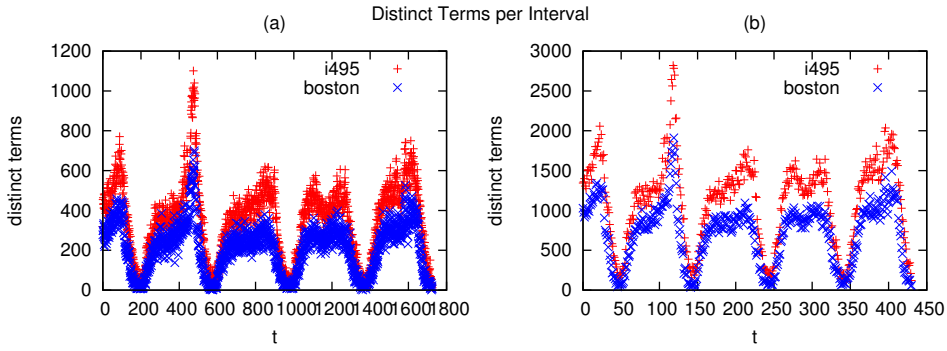


Figure 3.5: Dictionary Size per Interval: a) 225s b) 900s

Figure 3.6 plots the number of distinct terms per hour for the two time series.

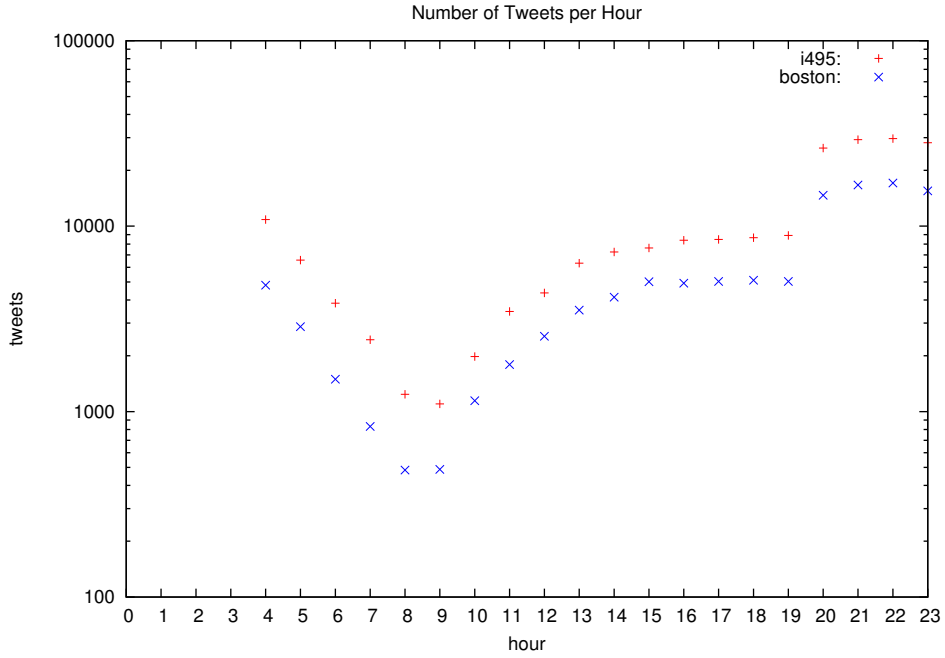


Figure 3.6: Tweets per Hour

The trend of the tweets being published rising and falling in unison is expected for this data because both locations used are in the same time zone.

3.2.1 Tf-Idf

The interval models were treated as documents and fed into a tf-idf computation, this resulted in many uninteresting terms floating to the top.

3.3 Term Matrix

Given the massive dictionary as a row per term, and each column as the model for that interval, you can feed the data into a variety of problem solvers to look for patterns.

Table 3.1: Percentage of Non-zero Cells

	225s	**450s**	**600s**	**900s**
Boston	0.43%	0.80%	1.03%	1.45%
I-495	0.71%	1.29%	1.64%	2.26%

```
EDU>> nnz(X) / (x * y)
```

The matrix was run through PCA with the goal of obtaining lower dimensionality. It was also run it through RPCA (TFOCS[1]) to search for patterns. Both methods provided very little input. The percentage

[1] http://tfocs.stanford.edu

of non-zero cells in the 900s matrix was 3.5% and 1.8% for the 450s interval global matrix. Therefore, even with the significantly smaller dictionary the matrices built are still very sparse.

3.3.1 PCA

Principal component analyst was attempted to reduce the high-dimensionality. The results were inconclusive because the values never converged.

3.3.2 RPCA

The TFOCS formulation of Robust PCA was used. The goal was to identify patterns from the massive background noise. Early attempts caused MatLab to crash. The matrix itself was therefore cut down so that fewer terms were involved and the data was still too sparse and no patterns were really identified.

3.4 Hierarchical Model

Given models for each interval between two locations, a hierarhical model can be built. For each term in an inteval, if it appears in both locations, its term frequency is moved up into a global model.

The goal was to find a better approach for identifying trends by only focusing on the global term space (6155 terms). With these global models, similar approaches were taken as with the location specific ones.

I surmised that the entropy would not change significantly over time because there is sufficient noise.

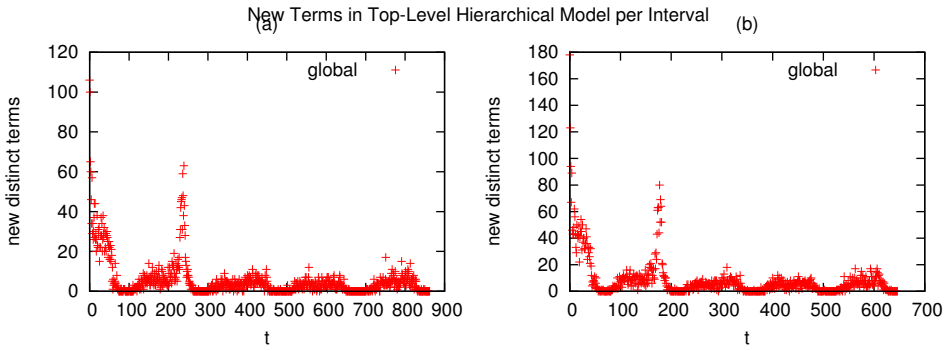


Figure 3.7: New Terms at Top-Level per Interval: a) 450s b) 600s

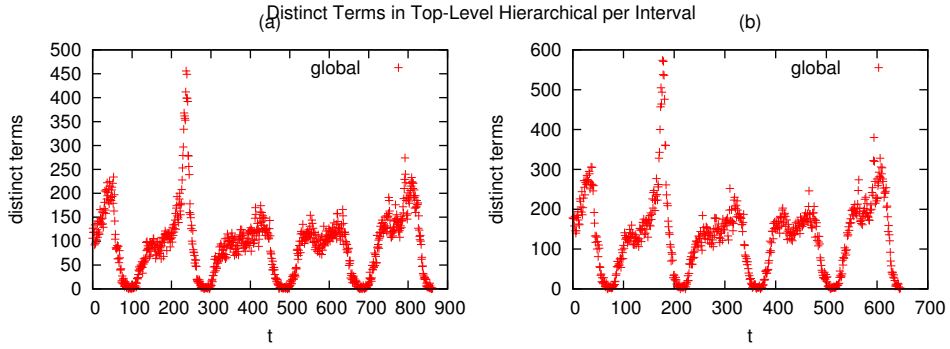


Figure 3.8: Dictionary Size at Top-Level per Interval: a) 450s b) 600s

These top terms are the terms in the global model that haven't the highest tf-idf values. Each model was treated as a document for the document count. Nothing of interest jumped out from these computations.

3.5 Entropy

Entropy is easy to compute and may be effective at determining the rise and fall of trends. There are a variety of entropy values you can consider. This work focused on the entropy of the models for each interval, as well as the overall window entropy in the form of the permutation entropy.

```
def compute_entropy(weights):
    """weight is a dictionary of term weights."""

    entropy = 0.0 +
        sum([(weights[term] * log10(1.0/weights[term])) \
              for term in weights])

    if entropy == 0.0:
        return 0.0

    # normalize entropy value.
    return entropy / log10(len(weights))
```

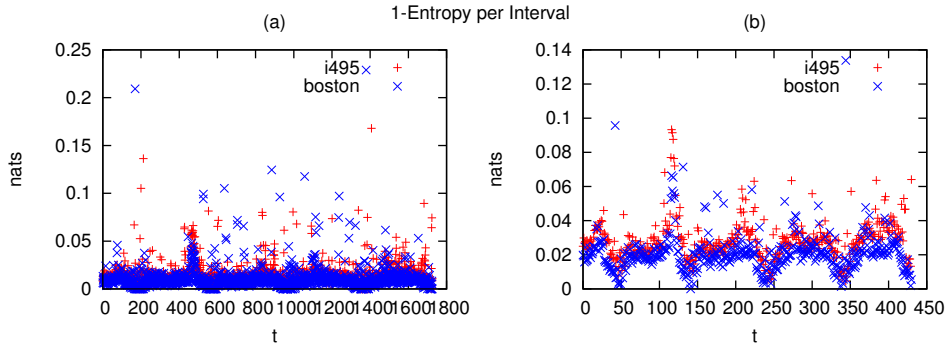


Figure 3.9: 1-Entropy per Interval a) 225s b) 900s

Most of the vectors have the highest possible value as their entropy value, indicating that each term within the vector is just as likely as any other. Therefore, the vectors of interest are those with lower entropy values.

This threshold that jumps out from the graphs is 0.045. The percentage of intervals that meets this point is 2.1% for the 225s interval, 3.0% for 450s, and 5.6% for 900s. If you further restrict that both models for the interval must exceed the threshold, the percentage of intervals for 900s drops from 5.6% to 1.39%.

The top term weight term lists for the matching intervals are detailed. These terms are the high term weight terms for the case where the entropy is lower than normal.

Interval	1st	2nd	3rd	4th	5th
20121012010423-i495	biden	vpdebate	debate	ryan	joe
20121012010423-boston	biden	ryan	debate	vpdebate	joe
20121012011923-i495	biden	vpdebate	ryan	joe	debate
20121012011923-boston	biden	vpdebate	ryan	debate	joe
20121012013423-i495	biden	ryan	vpdebate	debate	joe
20121012013423-boston	biden	vpdebate	ryan	debate	joe
20121012014923-i495	biden	ryan	vpdebate	debate	joe
20121012014923-boston	biden	ryan	debate	vpdebate	joe
20121012020423-i495	biden	ryan	vpdebate	debate	joe
20121012020423-boston	ryan	vpdebate	biden	debate	abortion
20121012021923-i495	ryan	biden	vpdebate	debate	paul
20121012021923-boston	biden	ryan	vpdebate	debate	joe

The table clearly details that nearly all the top terms are identical for these intervals. If there were fewer tweets for this interval, the entropy value should remain unaffected because we normalized it by the maximum entropy value for the specific model.

3.5.1 Permutation Entropy

If any term count vectors had different values for the terms, but the same ranking these terms, they could be considered similar. The entropy of these variations should change as vectors become more and less similar over the time series window.

Permutation entropy is the entropy of the variations in the sorted indexes of the values in a list. This computation was applied to the data set for a variety of interval spans and there were no two vectors found with the same ordering of the term counts for any pairing.

It would be interesting to perform this experiment by searching for localized matches in the ordering, versus requiring a match over the entire vector.

3.5.2 Set Resemblance

Given the failure to group vectors into bins for the overall time series, a more lenient method was attempted. By grouping the vectors by set resemblance, we place vectors whose set resemblance exceeds some threshold into the same bin. For set resemblance, the vector is treated as a binary document representation and the value is computed by:

$$\text{resem} = |A| \cap |B| \,/\, |A| \cup |B|$$

This approach to clustering with the 450s models took several hours and all pairings were below 40% resemblance. Less helpfully, approximately 79% of resemblance values were at most 10%. Using 225s interval lengths, approximately 98% of the the values were at most 10%. The shorter inteval length means fewer tweets are used as input to the model.

3.5.3 Global Entropy/Hierarchical

Given the pair of time series from two locations, they can be converted into a hierarchical model. The terms that appear in both locations for each interval are moved into a top-level global version of the model, the remaining terms remain in the lower levels. The entropy of this top model should change as terms enter and exit.

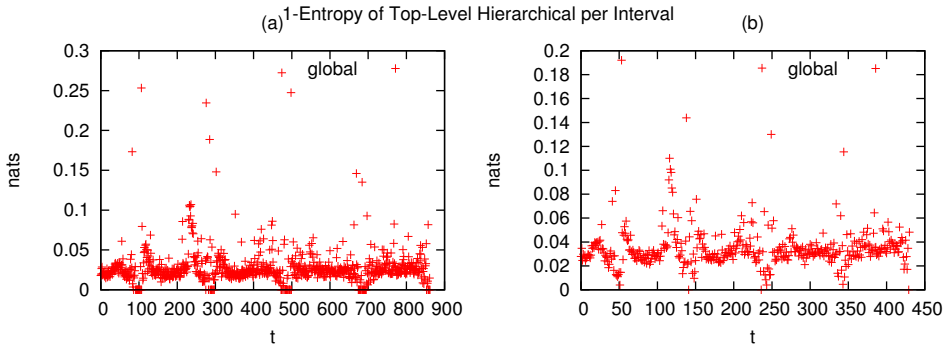


Figure 3.10: Top-Level Hierarcical Model Entropy: a) 450s b) 900s

Some points in the graph jump out from the others. The threshold line appears to be 0.045. Given, this point the percentage of models at or above that point for 225s is 9.3%, 450s is 10.2%, and for 900s is 15.1%. This threshold is the same as what was used for the basic model, but produced far more matching intervals. Therefore, it's possible there is a better cut-off line.

Sample from the 900s, from the models picked out, the following terms have the highest term weight values. They have manually been grouped by color.

Interval	1st	2nd	3rd	4th	5th
20121011030423	yankees	raul	ibanez	hate	sleep
20121011103423	morning	cold	school	bed	happy
20121012004923	debate	vpdebate	ryan	tomorrow	tonight
20121012014923	biden	ryan	vpdebate	debate	joe
20121012030423	biden	debate	tonight	tomorrow	home
20121012040423	game	orioles	tomorrow	happy	life
20121012064923	detected	event	jobs	ill	sleep
20121012080423	sleep	airport	tired	win	months
20121012100423	event	detected	morning	sleep	school
20121012103423	morning	thank	friday	cold	happy
20121012113423	morning	school	friday	boston	cold
20121012181923	smile	boston	beautiful	miss	photo
20121013001923	bar	nationals	tonight	park	game
20121013004923	nats	natitude	nationals	game	yes
20121013013423	nats	natitude	game	tonight	home
20121013041923	nats	nationals	game	season	natitude
20121013081923	stay	sleep	feel	damn	guys
20121013093423	cold	run	tired	max	bed

Clearly, some of the terms with high term weights that appear in this table should be trimmed off as stopwords. Beyond that, many intervals have terms in common, which may show that the global model's entropy dropping below a threshold is indicative of a trend forming.

Chapter 4

Conclusion

4.1 Data Set

The largest impediment to modeling was the sparseness of the data. Finding an effective and programmatic method to keep the dictionary small is a required step in developing basic algorithms that can be extended for the large data.

As previously mentioned, there is likely value in tying metadata into a modeling attempt. However, this work did not explore that avenue because an effective method to reduce the sparse data set wasn't identified.

Twitter provides an API for retrieving what they have identified at a given point as a trending topic. Collecting both the list of trends for each interval alongside the data may provide some assistance in reducing the dictionary. One could use the trending terms over the time window as the dictionary itself, and track the changes in the occurrences (rise and fall) of those trending topics.

By collecting data from many points as well as the trending topics, one could attempt to determine how Twitter identifies its trending topics. Questions include: are any areas ignored, or given more weight into their algorithm? Do they watch key cities or locations as tipoff point for trends? And is their algorithm language independent? By understanding their algorithm one can better model the data, as many users watch what is trending and post to simply join the trend. Also, there is an element of spam involved that leverage recently created accounts that post only to trending topics in the hopes of getting noticed by users, who then click on a link in the post itself.

4.2 Hierarchical

The hypothesis pertaining to terms occurring in multiple locations was not fully evaluated with this data set. However, future experiments can further explore the hierarchical modeling of social media.

4.3 Entropy

Computing the entropy for a specific interval is computationally simple and provided interesting results as the entropy value dropped beneath a threshold. However, it has not been verified that this change in entropy is sufficient to forecast a trend. Currently, it has only been verified to identify one.

The permutation entropy approach should be further explored to include localized matching orders, versus requiring the entire vector to match.

4.4 Periodicity

Hot topics, or trends in the data should follow periods. For instance, elections occur with some regularity. There are some events that occur a-periodically, such as earthquakes. These aperiodic events may be more difficult to forecast. Any attempt to model the trends may benefit from identifying their periodicity.

Although this work did not exhaustively exercise the state of the art in modeling of social media, it did explore underdeveloped areas, such as entropy and multi-location modeling for Twitter data.

Bibliography

[1] S. Asur and B. A. Huberman. Predicting the future with social media. Technical report, HP Laboratories, 2010.

[2] S. Asur, B. A. Huberman, G. Szabo, and C. Wang. Trends in social media: Persistence and decay. In *in ICWSM 10: Proceedings of international AAAI Conference on Weblogs and Social*, 2011.

[3] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts. Everyone's an influencer: quantifying influence on twitter. In *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM '11, pages 65–74, New York, NY, USA, 2011. ACM.

[4] S. Banerjee, K. Ramanathan, and A. Gupta. Clustering short texts using wikipedia. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 787–788, New York, NY, USA, 2007. ACM.

[5] A. L. Berger, V. J. D. Pietra, and S. A. D. Pietra. A maximum entropy approach to natural language processing. *Comput. Linguist.*, 22:39–71, March 1996.

[6] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 1 edition 2006 edition, 2006.

[7] D. M. Blei. Introduction to probabilistic topic models. Publication Pending.

[8] J. Bollen, H. Mao, and A. Pepe. Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media*, 2011.

[9] D. Boyd, S. Golder, and G. Lotan. Tweet, tweet, retweet: Conversational aspects of retweeting on twitter. *Hawaii International Conference on System Sciences*, 0:1–10, 2010.

[10] T. Brants, A. C. Popat, P. Xu, F. J. Och, J. Dean, and G. Inc. Large language models in machine translation. In *In EMNLP*, pages 858–867, 2007.

[11] M. Cataldi, L. Di Caro, and C. Schifanella. Emerging topic detection on twitter based on temporal and social terms evaluation. In *MDMKDD '10: Proceedings of the Tenth International Workshop on Multimedia Data Mining*, pages 1–10, New York, NY, USA, 2010. ACM.

[12] M. Cha, H. Haddadi, F. Benevenuto, and K. P. Gummadi. Measuring user influence in twitter: The million follower fallacy. In *in ICWSM 10: Proceedings of international AAAI Conference on Weblogs and Social*, 2010.

[13] S. F. Chen and J. Goodman. An empirical study of smoothing techniques for language modeling. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, ACL '96, pages 310–318, Stroudsburg, PA, USA, 1996. Association for Computational Linguistics.

[14] M. Cheong and V. Lee. Integrating web-based intelligence retrieval and decision-making from the twitter trends knowledge base. In *SWSM '09: Proceeding of the 2nd ACM workshop on Social web search and mining*, pages 1–8, New York, NY, USA, 2009. ACM.

[15] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[16] S. T. Dumais and M. Berry. Using linear algebra for intelligent information retrieval. *SIAM Review*, 37:573–595, 1995.

[17] R. Ford. Earthquake: Twitter users learned of tremors seconds before feeling them. `http://www.hollywoodreporter.com/news/earthquake-twitter-users-learned-tremors-226481`, August 2011.

[18] D. Frankowski, D. Cosley, S. Sen, L. Terveen, and J. Riedl. You are what you say: privacy risks of public mentions. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, pages 565–572, New York, NY, USA, 2006. ACM.

[19] M. Gladwell. *The Tipping Point: How Little Things Can Make a Big Difference*. Back Bay Books, New York, NY, USA, 2002.

[20] R. Greenfield. New twitter algorithm could out dudes pretending to be lesbians. `http://www.theatlanticwire.com/technology/2011/07/new-twitter-alogrithm-could-out-dudes-pretending-be-lesbians/40451/`, July 2011.

[21] C. Grier, K. Thomas, V. Paxson, and M. Zhang. @spam: the underground on 140 characters or less. In *Proceedings of the 17th ACM conference on Computer and communications security*, CCS '10, pages 27–37, New York, NY, USA, 2010. ACM.

[22] D. Gruhl, R. Guha, R. Kumar, J. Novak, and A. Tomkins. The predictive power of online chatter. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, KDD '05, pages 78–87, New York, NY, USA, 2005. ACM.

[23] L. Hong and B. D. Davison. Empirical study of topic modeling in twitter. In *Proceedings of the First Workshop on Social Media Analytics*, SOMA '10, pages 80–88, New York, NY, USA, 2010. ACM.

[24] B. J. Jansen, M. Zhang, K. Sobel, and A. Chowdury. Twitter power: Tweets as electronic word of mouth. *Journal of the American Society for Information Science and Technolog*, pages 2169–2188, July 2009.

[25] A. Java, X. Song, T. Finin, and B. Tseng. Why we twitter: understanding microblogging usage and communities. In *WebKDD/SNA-KDD '07: Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis*, pages 56–65, New York, NY, USA, 2007. ACM.

[26] C. Lee, H. Kwak, H. Park, and S. Moon. Finding influentials based on the temporal order of information adoption in twitter. In *WWW '10: Proceedings of the 19th international conference on World wide web*, pages 1137–1138, New York, NY, USA, 2010. ACM.

[27] K. Lee, J. Caverlee, and S. Webb. Uncovering social spammers: social honeypots + machine learning. In *Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '10, pages 435–442, New York, NY, USA, 2010. ACM.

[28] K. Lerman and R. Ghosh. Information contagion: An empirical study of the spread of news on digg and twitter social networks. In *in ICWSM 10: Proceedings of international AAAI Conference on Weblogs and Social*, 2010.

[29] J. Leskovec, L. A. Adamic, and B. A. Huberman. The dynamics of viral marketing. *ACM Trans. Web*, 1, May 2007.

[30] C. X. Lin, B. Zhao, Q. Mei, and J. Han. Pet: a statistical model for popular events tracking in social communities. In *KDD '10: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 929–938, New York, NY, USA, 2010. ACM.

[31] J. Lin, R. Snow, and W. Morgan. Smoothing techniques for adaptive online language models: Topic tracking in tweet streams. In *KDD '11: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, USA, 2011. ACM.

[32] X. Liu, S. Zhang, F. Wei, and M. Zhou. Recognizing named entities in tweets. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 359–367, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[33] B. Malkin. Osama bin laden killed: Pakistani man live tweets deadly raid. `http://www.telegraph.co.uk/technology/twitter/8487686/Osama-bin-Laden-killed-Pakistani-man-live-tweets-deadly-raid.html`, May 2011.

[34] C. Manning, P. Raghavan, and H. Schuetze. *Introduction to Information Retrieval*. Cambridge University Press, 1 edition july 7, 2008 edition, 2008.

[35] D. McElroy. Iran protest news travels fast and far on twitter. `http://www.telegraph.co.uk/news/worldnews/middleeast/iran/5549955/Iran-protest-news-travels-fast-and-far-on-Twitter.html`, June 2009.

[36] Q. Mei, C. Liu, H. Su, and C. Zhai. A probabilistic approach to spatiotemporal theme pattern mining on weblogs. In *Proceedings of the 15th international conference on World Wide Web*, WWW '06, pages 533–542, New York, NY, USA, 2006. ACM.

[37] K. Murphy. *Dynamic Bayesian Networks: Representation, Inference and Learning*. PhD thesis, University of California, Berkeley, 2002.

[38] S.-H. Na, J. Kim, I.-S. Kang, and J.-H. Lee. Exploiting proximity feature in bigram language model for information retrieval. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 821–822, New York, NY, USA, 2008. ACM.

[39] B. O'Connor, R. Balasubramanyan, B. R. Routledge, and N. A. Smith. From tweets to polls: Linking text sentiment to public opinion time series. In *in ICWSM 10: Proceedings of international AAAI Conference on Weblogs and Social*, 2010.

[40] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. In *Proceedings of the 7th International World Wide Web Conference*, pages 161–172, 1998.

[41] D. Pell. Egypt, twitter and the straw man revolution. `http://www.huffingtonpost.com/dave-pell/egypt-twitter-and-the-str_b_815906.html`, January 2011.

[42] S. Petrović, M. Osborne, and V. Lavrenko. Streaming first story detection with application to twitter. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 181–189, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[43] X.-H. Phan, L.-M. Nguyen, and S. Horiguchi. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *Proceeding of the 17th international conference on World Wide Web*, WWW '08, pages 91–100, New York, NY, USA, 2008. ACM.

[44] D. Ramage, S. Dumais, and D. Liebling. Characterizing microblogs with topic models. In *in ICWSM 10: Proceedings of international AAAI Conference on Weblogs and Social*, 2010.

[45] M. Raymond. Twitter donates entire tweet archive to library of congress. `http://www.loc.gov/today/pr/2010/10-081.html`, April 2010.

[46] A. Ritter, C. Cherry, and B. Dolan. Unsupervised modeling of twitter conversations. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 172–180, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.

[47] D. M. Romero, B. Meeder, and J. Kleinberg. Differences in the mechanics of information diffusion across topics: idioms, political hashtags, and complex contagion on twitter. In *Proceedings of the 20th international conference on World wide web*, WWW '11, pages 695–704, New York, NY, USA, 2011. ACM.

[48] M. Rosen-Zvi, C. Chemudugunta, T. Griffiths, P. Smyth, and M. Steyvers. Learning author-topic models from text corpora. *ACM Trans. Inf. Syst.*, 28:4:1–4:38, January 2010.

[49] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth. The author-topic model for authors and documents. In *Proceedings of the 20th conference on Uncertainty in artificial intelligence*, UAI '04, pages 487–494, Arlington, Virginia, United States, 2004. AUAI Press.

[50] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *WWW '10: Proceedings of the 19th international conference on World wide web*, pages 851–860, New York, NY, USA, 2010. ACM.

[51] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, and M. Demirbas. Short text classification in twitter to improve information filtering. In *SIGIR '10: Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 841–842, New York, NY, USA, 2010. ACM.

[52] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe. Predicting elections with twitter: What 140 characters reveal about political sentiment. In *Proceedings of the Fourth International AAAI Conference on Weblogs and Social Media*, 2010.

[53] S. Vieweg, A. L. Hughes, K. Starbird, and L. Palen. Microblogging during two natural hazards events: what twitter may contribute to situational awareness. In *CHI '10: Proceedings of the 28th international conference on Human factors in computing systems*, pages 1079–1088, New York, NY, USA, 2010. ACM.

[54] A. Watters. The library of congress' twitter archive, one year later. `http://www.forbes.com/sites/oreillymedia/2011/06/13/the-library-of-congress-twitter-archive-one-year-later/`, June 2011.

[55] J. Weng, E.-P. Lim, J. Jiang, and Q. He. Twitterrank: finding topic-sensitive influential twitterers. In *Proceedings of the third ACM international conference on Web search and data mining*, WSDM '10, pages 261–270, New York, NY, USA, 2010. ACM.

[56] Y. Yang. An evaluation of statistical approaches to text categorization. *Inf. Retr.*, 1:69–90, May 1999.

[57] L. Yao, D. Mimno, and A. McCallum. Efficient methods for topic model inference on streaming document collections. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 937–946, New York, NY, USA, 2009. ACM.

[58] D. Zhao and M. B. Rosson. How and why people twitter: the role that micro-blogging plays in informal communication at work. In *GROUP '09: Proceedings of the ACM 2009 international conference on Supporting group work*, pages 243–252, New York, NY, USA, 2009. ACM.

[59] J. Zhao and Y. Yun. A proximity language model for information retrieval. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, pages 291–298, New York, NY, USA, 2009. ACM.