TwitterReporter: Breaking News Detection and Visualization through the Geo-Tagged Twitter Network

Brett Meyer, Kevin Bryan, Yamara Santos, Beomjin Kim
Computer Science Department
Indiana University - Purdue University
Fort Wayne, IN, 46805, U.S.A.
{meyebe01, bryakm01, santym01}@students.ipfw.edu, kimb@ipfw.edu

Abstract

The Twitter social network provides a constant stream of concise data, useful within both geospatial and temporal domains. Previous studies have attempted to mine and find recent news topics within the data set. Although the efforts were a step in the right direction, we believe more is needed to accomplish a useful and interesting task: using live Twitter data to automatically identify breaking news events in near real-time. In this paper, we present methods to collect data, identify breaking news topics, and display results in a geo-temporal visualization. Our goal is to provide a solution that discovers breaking news faster than traditional reporting mediums.

1 INTRODUCTION

Twitter is a social network that has quickly gained popularity with today's connected society. The network consists of brief messages, also known as "tweets", containing a maximum of 140 characters. Due to the restricted size, the text may not necessarily contain well-formed ideas or developed context. However, posts are typically complete enough to be coherent. Fortunately, the quick nature allows for large quantities of concise and diverse concepts. Typical post topics include personal updates, current news, editorials, marketing, and discussions. Twitter's massive dataset is created by over 190 current million users worldwide, many of which submit posts frequently per day [4].

Therefore, Twitter can be defined as a "micro-blogging" platform, a special type of Social Networking Service that places emphasis on simplicity and openness [2]. More importantly, Twitter has recently become a major form of communication, not only to stay connected with family and friends, but also for breaking news and global discussions. Information is frequently relayed faster and more effectively than traditional news and service. Further, due to mobile environments, users witnessing the event as it occurs are able to "tweet" news in real-time [10]. Recent examples include many recent earthquakes (described in "Results"), the 2008 Mumbai attacks [6], and California forest fires.

This vast diversity of tweet topics causes Twitter's dataset to be fairly difficult to process. However, Sankaranarayanan et al. (2009) attempted to automatically cluster posts into topic and geospatial-based groups [1]. The process utilized an innovative combination of various Twitter data-streams, clustering, rule-sets, and data mining techniques. However, the solution's geospatial component mainly relied on natural language techniques (the geotag API, described below, had not yet been introduced). Further, heavy emphasis was placed on "quality users" (often major news sources), unfortunately causing results to lag behind traditional sources.

In OMG project, Longueville and colleagues also studied how Twitter can be used as a source of spatio-temporal information [2]. By focusing on a recent, real-life case of a forest fire, they aimed to demonstrate the possible role of supporting emergency planning, risk assessment and damage assessment. Specifically, the analysis drew on publicly available Twitter messages published during a forest fire that took place in France, with a particular focus on the identification of the content's temporal, spatial and social dynamics [2]. Although the study only involved one use case, it is argued that the richness of the information provided in a real event by users from different backgrounds provide robust outcomes to a range of scenarios and related Location Based Social Networks.

In November 2009, Twitter introduced a new capability and updated API allowing individual tweets to be geo-tagged -- associated with a specific location's latitude and longitude values [8]. Due to this update, there has been an explosion of new web and mobile applications. Unfortunately, many of these applications lack usefulness and mainstream relevance, focusing mainly on entertainment, random conversations, and personal locations. The new geo-tag attribute has an important and useful potential: to follow real-time events within geographical locations. Twitter can be used as an informing tool, delivering breaking events through analysis of the vast amount of real-time, geo-tagged data provided by users "in the field."

The goal of TwitterReporter is to demonstrate Twitter's ability to automatically identify breaking events, through processing users' tweets, and visualize them on a geospatial 2-dimensional map representation. Emphasis on near real-time results, paired with a focus on all distributed users, will further the groundwork completed by other previous studies. The vision for the end result is a web application identifying near real-time breaking news events. The solution should be robust enough to be usable

in varied environments (geology, sociology, general news reporting, and more).

2 METHOD AND IMPLEMENTATION

In this section, the methods used to collect Twitter data, cleanse and process the data, generate topics, and visualize results are described. This process is depicted in Figure 1.

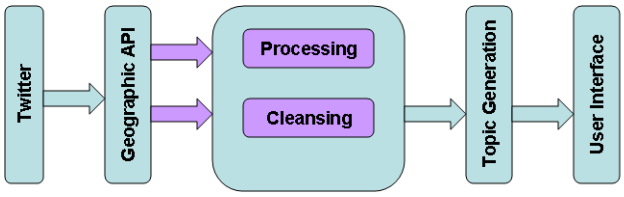

Figure 1. System Architecture

2.1 Data

The data set consists of live tweets collected from the Twitter API. Raw data is obtained through a restricted geographic API access level, used with Twitter's permission. A Perl script opens a streaming connection and receives Tweets in real-time. The geographic API method accepts up to 200 bounding boxes (rectangles identifying geological areas) as parameters, each with a maximum of 10 decimal degrees per edge. The method is used to divide the entire 48 continental US states into a grid. This effectively provides us with all geo-tagged tweets, within the US, in real-time. The API responses, originally in JavaScript Object Notation (JSON, a lightweight data-interchange format) are parsed by a Perl module and stored in a MySQL database. Each "tweet" consists of attributes relevant and irrelevant to our application. TwitterReporter focuses solely on date/time, "tweet" text and geo-tag (latitude/longitude). All others are ignored.

Commonly, the collected tweets include many types of noise that are not related to the main concept of the post. Through another Perl script, we applied the following raw data cleaning techniques to purify and decrease the complexity of the compiled data. First, remove all posts from authors having user profiles identified as non-English. Second, remove all tweets that contain non-US ASCII characters. The script UTF-8 encodes the tweet text and checks for Unicode values. Although this effectively removes numerous non-English posts, this will also remove some English tweets that use icons, mathematic symbols, etc. In general, those types of posts are extremely limited and should have a minimal effect on the results. Next, polish the data: remove URLs, remove reply and re-post syntax (@username), remove XHTML encoded characters, remove non-alphanumeric characters, and remove extra white-space characters including new lines, tabs, and non-needed spaces. Finally, remove blank (and nearly blank) entries resulting from the polishing phase.

2.2 Topic Generation

First, identify topics that are either deemed important or relevant in relation to breaking news. For a proof of concept, disaster topics were utilized. The topics were grouped into three categories: natural events, man-made events, or other uncategorized events. For example, a natural event includes "tornado", "earthquake", and "hurricane". A man-made event includes "riots", "protest", and "arson". Uncategorized events include other relevant topics (such as "terrorism"). Then, for each topic, applicable synonyms are manually identified. For instance, the topic "tornado" utilizes the synonyms "twister" and "funnel".

Twitter is notable in its design in relation to both time and space [2]. Tweets are organized in timelines (i.e., series of tweets sorted and displayed in reverse chronological order) with a level of accuracy of one second [2]. Also, each geo-tagged tweet contains the exact location of the post. Therefore, the data is able to be organized as a spatial-temporal domain in which posts are processed in one hour batches. Then, each one hour batch is further broken down into .5 by .5 decimal degree geospatial areas.

Each batch is processed through a Porter2 stemming algorithm which removes various word suffixes, without using a stem dictionary [5]. This algorithm decreases complexity by converting every word to its root; for example, the word "tornadoes" has a "tornado" stem. The resulting subset of data is run through document frequency (DF) weighting [3]. The DF calculates the number of occurrences of a given term within the entire batch of posts. If a topic's DF weight is high enough, it is stored as a geospatial, real-time breaking news topic occurrence to be displayed in the visualization. The DF weight is a combined sum of all occurrences of the topic term and its synonyms within the batch of data. The threshold, α, is determined through empirical studies.

2.3 Visualization

The final step in the solution is visualization. Due to Twitter being a web application, as well as usage consisting entirely of mobile and web tools, the usefulness of this work relies on implementation within a web environment. To provide this convenience to users, the topics are visualized using Google Maps, a popular and very customizable map visualization API. The web application was implemented to allow users to search by topic selection, date range, and result limitations, as well as provide intuitive results. The following section describes the user interface, its capabilities, and the mechanics behind it.

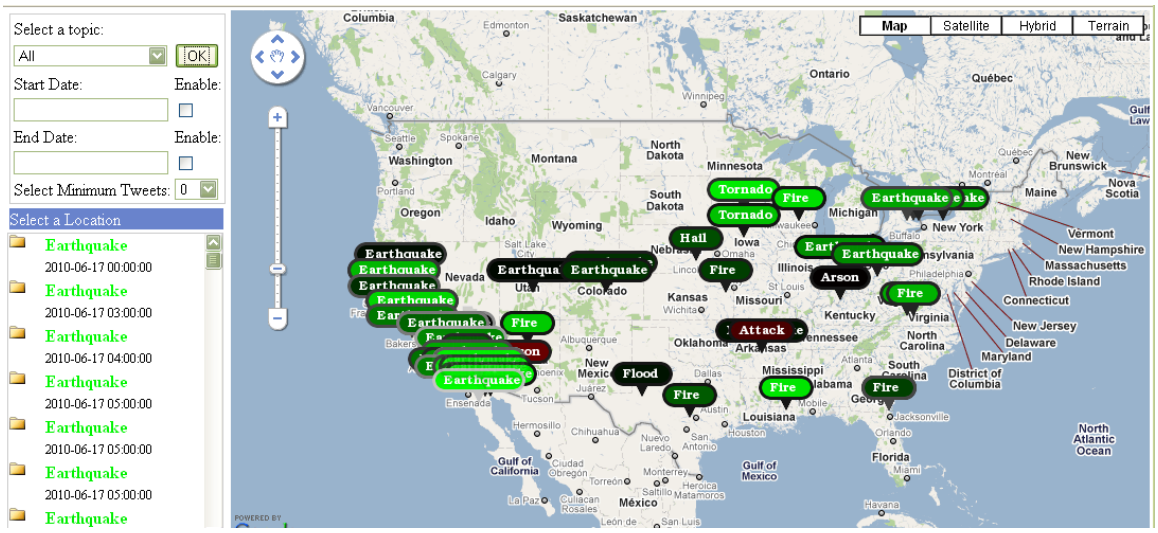

Figure 2. Screenshot of the website with the left pane showing user selections and the results of the search with the tweets corresponding geographic marker shown on the right pane

2.4 User Interaction

The visualization is organized into a few key areas. These include search inputs, map navigation and topic results. As shown in Figure 2, the application is organized with the map, containing the topic icons, on the right side of the page. The user inputs are on the upper left-hand side and include topic, date/time range, and a minimum tweets threshold (if a topic's composition contains less tweets than the threshold, it is not displayed). The lower left-hand side contains topic results, along with their tweet compositions.

After the user submits inputs, the topics are returned and displayed in chronological order on the result tree (lower left-hand side) and plotted on the map. We used a color-coding system to represent pre-defined categories. The red stands for man-made disasters, green for natural disasters, and blue for all others. The intensity of the color specifies the relative quantity of a topic's tweets in comparison to the other topics listed.

The children of each topic, in the result tree, consist of the exact tweets that composed the topic occurrence. For most uses, the quantity of displayed children is limited to 20 for each topic. However, this is waved for the circumstance in which all topics are requested, at which point no limit is utilized. Upon clicking a topic in the tree, child tweets are listed below and displayed on the map as eye-drop icons (as shown in Figure 3).

All topic icons are dynamically generated by a PHP script using the graphics library, ImageMagik. The size of the icon is adjusted dynamically to encompass the entire keyword displayed. The background of the icon is colored the same as the topic listed on the left side of the screen. The border of the icon is colored by a grayscale spectrum based on the relative number of tweets compared to the other topics. Topics containing the largest number of tweets are closer to white, whereas topics with low numbers of tweets are closer to black.

Figure 3. Screenshot of the website with the left pane showing the expanded topic

3 EVALUATION AND RESULTS

The following example is presented as an empirical demonstration of TwitterReporter's capabilities. According to news sources, the state of California suffered a multitude of earthquakes during the month of June. Therefore, as a test, the inputs that were entered on the website were topic: earthquake, start date: 6/1/2010 and end date: 6/30/10.

The results returned were tweets dated from 6/17/2010 (the start of TwitterReporter's scripts) to 6/30/2010. As shown in Figure 4, the topics displayed not only contained many tweets in California, but also in Colorado, Arkansas, Indiana, Ohio, and the New York/Canada border.

Analysis of the results demonstrated that the tweets from Colorado and Indiana (Figure 5) consisted of re-

tweets (re-posting another author's tweet, verbatim, within your own account), general discussions about the high number of recent US earthquakes, or noise.

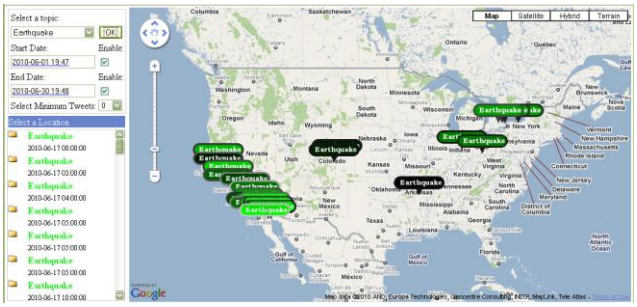

Figure 4. An output showing the visualization using keyword "Earthquake"

Figure 5. Re-Tweets, Discussions, and Noise

Even though California had the most tweets concerning earthquakes, there seems to be a few earthquake activities occurring in New York, Ohio and Arkansas. On June 24, 2010, the Wall Street Journal reported that an earthquake with a magnitude of 5.0 hit the Quebec-Ontario border [13]. The tremors were also felt across the border in New York. According to newsnet5.com, the Canadian earthquake was also felt as far as Ohio [14]. As shown in Figure 6, this activity has been also represented properly by the TwitterReporter.

Figure 6. A zoomed-in view showing tweets associated with a Canada earthquake

Further, on June 28, 2010, a small earthquake hit Arkansas. Although the application also identified the occurrence, there seems to be much less discussion [12]. This may have been a result of geological location and the size of population; it is assumed that the smaller population density offered fewer relevant tweets.

Another noticeable scale earthquake was identified and well presented through the developed visualization. On June 15, an earthquake of a magnitude of 5.7 hit Southern California and the Mexican Border [7]. Throughout the month other earthquakes hit Los Angeles, Northern California, and the San Francisco area [9] [11]. As seen in Figure 7, the application visibly displays tweet activities identifying the earthquake.

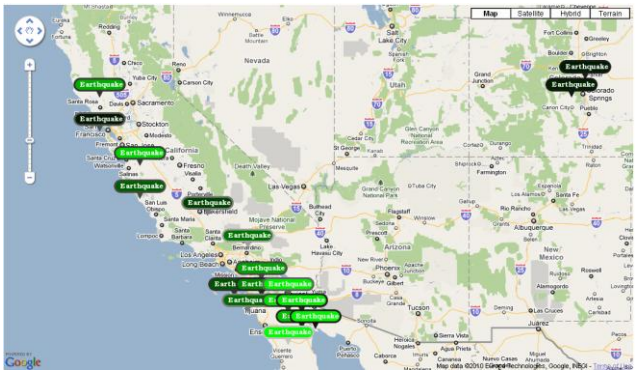

Figure 7. A zoomed-in view showing tweets associated with a California earthquake

From post experimental studies, we found that TwitterReporter outperformed the news media in identifying most earthquakes. For example, on June 28, 2010, the NowPublic News Coverage website reported that a small earthquake struck the San Francisco area around 7:47am [11]. While NowPublic (and other sources) reported the earthquake after 12:00pm, TwitterReporter presented the news before 10:00am. The earthquake is only one of several examples identified where the application beat traditional media by at least 2-3 hours. This quick response is extremely significant, as it proves the plausibility of tracking breaking events in near real-time.

4 DISCUSSION AND FUTURE WORK

As with any research in new areas, multiple paths were investigated. It was found that Twitter presents an extremely unique environment for data mining and natural language processing. This is due to the very nature of the product (140 characters or less), as well as the users and their intentions.

We explored several language processing techniques to automatically identify a representative term for a group of tweets. The first method used a variation of Term

Frequency-Inverse Document Frequency (TF-IDF) [3]. In TF-IDF weighting, terms are emphasized if they appear multiple times within one post ("term frequency"), but decremented if appearing in multiple posts ("inverse document frequency"). It quickly became apparent that TF-IDF was counter-intuitive for this use. First, 140 characters typically allow an important keyword to be used only once within a post. Authors are not able to fully develop main ideas; there is usually room for only one instance of the main topic keyword. Thus, term frequency hurt the results dramatically. Second, the project's goal revolves around multiple users reporting the same event. If a major event occurs, there is potential for the majority of the area's users writing about the topic. Therefore, the inverse of document frequency was of no assistance either.

The second, automated approach utilized a straight-forward Document Frequency (DF) algorithm, including the use of an extensive stopword list (commonly used English words to be removed from the posts). A dictionary of all words, within a chunk of processed posts, was utilized to identify terms whose frequencies passed a given threshold [3]. If the threshold was met, the term was presented as a breaking topic. This method immediately presented another Twitter data characteristic: noise. The results were overwhelmed with one of two types of terms: location-based keywords (ex: "New York", "The Big Apple", "Manhattan") and Twitter web service names (ex: "tweetmyjobs"). Even with an improved stopword list and an extensive geology database, the noise appeared to be extremely difficult to deal with. As discussed below, this may be an area of future work.

After exploring several feasible approaches, we ended with the proposed method (described in "Method and Implementation") which resolved previously explained issues. Through the use of specified lists of terms and synonyms, noise is effectively eliminated. The issues of slang, acronyms, and misspellings are effectively eliminated. Also, emphasis on DF, rather than TF or IDF, promotes occurrences of news-worthy topics identified by "the masses".

In the future, scope should be increased to world-wide data collection. A continental US limitation was put in place as a means to reduce noise and other variables. Better mechanisms to handle foreign languages and other international data characteristics will need to be introduced.

As seen in the evaluation and result section, events (such as "earthquake") are often wide-spread. The current system repeats all instances within their respective .5 by .5 decimal degree areas. This often results in dramatically overlapping topics among icons. Instead, adjacent, identical topics should be combined into one central topic. The aggregated topics would need to be differentiated from single events within the visualization.

Other areas can be improved upon in the future, including the effectiveness of presented results. Several of the selected topics are extremely seasonal. For example, "hurricane", "tornado", and other weather-related events only occur within portions of a year. Therefore, it would be interesting to dynamically modify a topic's DF threshold by using a seasonal factor. This will be effective in identifying useful information while avoiding noisy contents.

For TwitterReporter to be useful as a released, commercial product, user configurable capabilities are a must. Users should be able to identify their own sets of topics and synonyms, useful to their individual purposes. For instance, an earthquake research group would only be interested in their specific topics and would have no use for anything else. Likewise, a general news source may introduce widespread, all-encompassing topics as a new primary source.

Users may often be concerned with a limited geographical area. Therefore, the visualization web application should provide an interactive bounding box. On the map, users would click and drag and box around a specific area. Only topics from within the reduced geographic scope would be presented.

Lastly, more effort should be dedicated to revisiting the fully-automated topic generation solution, utilizing DF and stopword lists. Dynamic mechanisms would need to be developed to reduce as much noise as possible. It seems that this method may be useful in an entirely different way from the current solution.

5 REFERENCES

[1] Sankaranarayanan, J., Samet, H., Teitler, B.E., Lieberman, M.D., and Sperling, J., "TwitterStand: News in Tweets," *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 42-51, 2009.

[2] De Longueville, B., Smith, R. S., and Luraschi, G., "OMG, from here, I can see the flames!: a use case of mining Location Based Social Networks to acquire spatio-temporal data on forest fires," *Proceedings of the 2009 International Workshop on Location Based Social Networks*, pp. 73-80, 2009.

[3] Orasan, C., "Comparative Evaluation of Term-Weighting Methods for Automatic Summarization," *Journal of Quantitative Linguistics*, Vol. 16, Issue 1, pages 67-95, 2009.

[4] Schonfeld, E., "Costolo: Twitter Now Has 190 Million Users Tweeting 65 Million Times A Day," *TechCrunch*. N.p., 8 June 2010. Web. 5 Nov. 2010. <http://techcrunch.com/2010/06/08/twitter-190-million-users>.

[5] Porter, M., "An algorithm for suffix stripping," *Program*, Vol. 14, No. 33, pp. 130-137, 1980.

[6] Busari, S., "Tweeting the terror: How social media reacted to Mumbai," *CNN*. N.p., 27 Nov. 2008. Web. 5 Nov. 2010. <http://edition.cnn.com/2008/WORLD/asiapcf/11/27/mumbai.twitter/index.html>.

[7] "Poster of the Southern California Earthquake of 15 June 2010 - Magnitude 5.7," *Earthquake Hazards Program*. USGS, n.d. Web. 5 Nov. 2010. <http://earthquake.usgs.gov/earthquakes/eqarchives/poster/2010/20100615.php>.

[8] "Think Globally, Tweet Locally." *Twitter Blog*. Twitter, 19 Nov. 2009. Web. 5 Nov. 2010. <http://blog.twitter.com/2009/11/think-globally-tweet-locally.html>.

[9] "3.8 earthquake rattles California border," *Los Angeles Times*. N.p., 19 July 2010. Web. 5 Nov. 2010. <http://latimesblogs.latimes.com/lanow/2010/07/38-earthquake-rattles-california-border.html>.

[10] Ulanoff, L., "Twitter is the New CNN," *PC Mag*. N.p., 3 May 2010. Web. 5 Nov. 2010. <http://www.pcmag.com/article2/0,2817,2363351,00.asp>

[11] Judd, A., "San Francisco Earthquake June 28 2010: 3.5 Magnitude," *NowPublic*. N.p., 28 June 2010. Web. 5 Nov. 2010. <http://www.nowpublic.com/environment/san-francisco-earthquake-june-28-2010-3-5-magnitude-2634808.html>.

[12] "Magnitude 3.2 - ARKANSAS 6/28/2010," *Before It's News*. N.p., 29 June 2010. Web. 5 Nov. 2010. <http://beforeitsnews.com/story/88/483/Magnitude_3.2_-_ARKANSAS_6_28_2010.html>.

[13] Herring, C., "Canada Earthquake Shakes Area," *The Wall Street Journal*. N.p., 24 June 2010. Web. 5 Nov. 2010. <http://online.wsj.com/article/NA_WSJ_PUB:SB10001424052748703900004575325243652359592.html>.

[14] "5.0 quake in Canada felt in Northeast Ohio, aftershocks possible," *WEWS*. N.p., 23 June 2010. Web. 5 Nov. 2010. <http://www.newsnet5.com/dpp/news/local_news/possible-earthquake-reported-in-northeast-ohio>.