Analyzing Big Data with Twitter: Final Report

Pigskin: Visualizing Football Tweets

Team Members & Roles

Andrew Chao (@acchao / andrew.c.chao@ischool)

- Project Manager, primary researcher, backend (Solr)

Gilbert Hernandez (@thegilby / gahernandez@ischool)

- Coder – visualization (d3.js, Ratchet), website (Flask), frontend (Bootstrap)

Jacob Portnoff (@jacobportnoff / jacob.portnoff@ischool)

- Writer – data gathering/processing (Pig), backend

Mentor

Gilad Mishne (@gilad / gilad@twitter.com)

Project Website

http://groups.ischool.berkeley.edu/pigskin/

Github Repository

https://github.com/thegilby/pigskin

Final Presentation

https://github.com/thegilby/pigskin/blob/master/presentation/Pigskin_Final.pdf?raw=true

Goals

The hope of this project is to build an exploratory visualization application from data gathered from Twitter. We will gather data during football games from Twitter to determine the number of tweets that relate to professional football games. We hope to determine the most popular teams every week as well as a number of different metrics by which to compare games, teams, and the sport as a whole.

Strategy

In order to achieve our goal we developed a strategy of heavy tweet collection followed by a significant analysis. While we originally explored the angle of pursuing data collection and analysis simultaneously using Twitter's search filter, we discovered that our term/filter list was longer than what Twitter would allow (over 500 terms versus a maximum of 400 search terms). Further we wanted to explore not only the interest in particular football games, but the general interest in football versus total twitter 'chatter.' As a result, we established a data collection that centered on gaining as much data as possible during the times of interest. Specifically, we did data collection from 10:00am to 9:00pm (Pacific Time) on Sunday. This time frame allowed us to see what was happening on Twitter during football games on Sunday.

Our analytics strategy, heavily informed by our literature review, was designed to minimize our work load and the number of false positives and false negatives. We created a weighting scheme utilizing several whitelists of known football terms, team names, team usernames, player nicknames, and commonly used hashtags. By using this list, we were able to detect whether or not a tweet was about football with over 80% accuracy. Tweet rates and number of tweets were particularly interesting to us as well, and we were able to use Pig on our dataset to determine these metrics. We used this information to determine popularity of teams and games.

In addition to the items of specific interest, our strategy accounts for the variety of timeframes during which our data collection takes place. The level of a team's Twitter popularity across multiple game days is particularly interesting, especially when considering that other events are influencing interest on Sundays.

Accomplishments

Our hope for this project was to build an exploratory visualization application from data gathered from Twitter. We gathered data from Twitter during football games throughout the day on Sunday for seven weeks. We were able to successfully differentiate between football and non-football related tweets and perform analysis in a variety of ways.

We were able to determine and visualize the most popular teams every week as well as the more popular matchups. Through this project we were also able to pinpoint important events in football games and generate lists of trending terms that coincided with big plays, controversies, and game results.

We also were able to generate a map of some of the football tweets. Although only some of the tweets had geolocation information, we were able to leverage that information into a visualization that helped display the geographic distribution of football team supporters. While this is not a total representation of all of the football tweets, it was more than enough to make a useful visual aid. We attempted to exploit the 'location' data attached to twitter users, but the large number of users who did not fill this information out, coupled with the equally large number that filled out the requests with difficult to parse 'locations' like "Tony Montanga$ House" made this additional data difficult to categorize and control.

Data gathering

- 7 Sundays worth of tweets gathered (roughly 13.3 million tweets, ~2 million per day)
- Data structure of each tweet collected: date time, screen name, tweet, hashtags, and geolocation, football weight, football association
- Data uploaded into Solr database and indexed for flexible information retrieval

Algorithm development

- Tweets weighted for expected football relevance
    - Tweets, hashtags, and user names evaluated
        - Trigrams generated for analysis

        - Tweets checked for football relevant terms from an extensive list generated from multiple sources
            - Team names in a variety of forms were checked
            - Player names in a variety of forms were checked
            - Hashtags from verified team accounts were looked for
            - Nicknames and football specific terms were also on the lists
- Total corpus of 13.3+ million tweets evaluated
- Geolocation on relevant tweets evaluated positively (lat, long data)
- Collection of location data from users posting football related tweets was collected but proved insufficiently useful for visualization
    - Locations were not always available
    - Locations were often meaningless or not a place that could be reverse geo-coded

Software architecting/Coding

- Java (twitter4j) implemented data gathering
- Python (tweepy) implemented data gathering
- Python implemented data analysis
- Pig implementation for trending terms for weekly matchups
- Set up cronjob to automatically gather data on Sundays
- Solr to index and feed data to our visualizations
- d3.js and Rickshaw.js implementation of visualizations

Interface design

- Flask Python used to serve website
- Bootstrap used for quick deployment of front-end design

Timeline (Done by NFL Weeks)

Week 7 (Sunday October 22, 2012)

- collect data on all games
- ID boundaries of the data
- Assumptions: time, user, tweet, location, hashtags
- Gather research material
- Begin working on exploratory visualization application
- Build expectation model
- Build corpus of tags across teams, general football stuff
- ID how to identify location without geolocation tag on (if possible)
- Sentiment analysis tool testing

Week 8 (Sunday October 28, 2012)

- collect data on all games
- Visualization model for football interest versus other trends

Week 9 (Sunday November 4, 2012)

- collect data on all games
- Visualization model for mapped information

Week 10 (Sunday November 11, 2012)

- collect data on all games
- Visualization mockups for all expected outcomes + sample code
- Sentiment analysis testing completed

30%-40% done point – Nov 13 – 1st report due

Week 11 (Sunday November 18, 2012)

- collect data on all games
- Testing hypotheses
- Visualization application work

Week 12 (Sunday November 25, 2012)

- collect data on all games
- Continued analysis and programming

Week 13 (Sunday December 2, 2012)

- collect data on all games
- Visualization application completed
- Finishing touches

100% done – Dec 10

Literature Review

(See Appendix A for full citations)

1. Understanding the Demographics of Twitter Users (2011)

Northeastern University: Alan Mislove, Yong-Yeol Ahn
Technical University of Denmark: Sune Lehmann
Harvard Medical School: Jukka-Pekka Onnela, J Niels Rosenquist
link: http://www.aaai.org/ocs/index.php/ICWSM/ICWSM11/paper/viewFile/2816/3234

This paper tries to analyze the demographic of twitter users. Researchers have begun using twitter as a means of measuring and predicting real world phenomena; such predictions would be greatly enhanced with better demographic data and provide more insight to possible biases.

Takeaways

- 75.3% of publicly visible users listed a location
- Gender can be predicted by the user of the first name. With a US-centric corpus, there was a match for names of 64.2% of users via a list of 5836 names.
- Last name is not a good indicator of ethnicity.

2. Using Twitter to Detect and Tag Important Events in Live Sports (2011)

Dublin City University: James Lanagan and Alan F. Smeaton
link: http://www.aaai.org/ocs/index.php/ICWSM/ICWSM11/paper/view/2821

Summary

This paper compares the effectiveness of Twitter vs audio-visual content analysis on detecting important events during live sports. The primarily method for analysis for twitter is through the use of volume of tweets or the delta through time.

Takeaways

- Twitter is a reactionary response
- It's effective at capturing goals scored, but not events like bookings (violations). It brings up an interesting point on what is worth tweeting in sports.
- Goals scored closer to the end of the game will generate more conversation. The base line of conversation rises which could make threshold detection more difficult.

3. Human as Real-Time Sensors of Social and Physical Events: A Case Study of Twitter and Sports Games (2011)

Rice University, Houston, TX: Siqi Zhao and Lin Zhong
Motorola Mobility, Libertyville, IL: Jehan Wickramasuriya and Venu Vasudevan
Link: http://arxiv.org/abs/1106.4300

Summary

Using Twitter to discover in real-time, events occurring in a NFL football game within 40 seconds and with 90% accuracy.

Takeaways

- Good at predicting events such as touchdowns but poor at fumbles (64%) and other less important events.
- Streaming API is more useful for real time data.
- Team names appear in 60% of game-related tweets.
- top 10 most frequent words are either game terminology or team names.
- Processing was done by first removing urls, @username, emoticons, punctuation, and stop words.
- Post rate was used to determine the important event with an adaptive window, but streaming api has a post rate limit of 50 tweets per second; failed during Super Bowl.
- check out www.sportsense.us
- only really works with predetermined keywords.

4. Lexical Normalisation of Short TExt Messages: Makn Sens a #twitter (2011)

The University of Melbourne: Bo Han, Timothy Baldwin

Link: http://dl.acm.org/citation.cfm?id=2002520

Summary
Due to the 140 character limit, this paper proposes a method for identifying and normalising ill-formed words. Some of the important issues that need to be addressed include the de-abbreviation of words such as “b4” to their canonical versions, “before”. The proposed method is a cascaded one that builds upon sms text normalization, using only single token words; It also excludes hashes, mentions, and urls.

Takeaways
- Uses word similarity, dictionary look ups,
- Not all ill-formed words provide useful context.

5. Analyzing Twitter for Social TV: Sentiment Extraction for Sports (2011)

Rice University: Siqi Zhao and Lin Zhong
Motorola Mobility, Libertyville, IL: Jehan Wickramasuriya and Venu Vasudevan
Link: http://ceur-ws.org/Vol-720/Zhao.pdf

Summary
Building upon their past research, they use their real-time twitter sports event detection website to assist in analyzing real-time sentiment of NFL games. Their goal is assist with advertisers and possible product placement that makes more sense and coincides with the sentiment of tv viewers.

Takeaways
- Emoticons are a strong signal and useful source for determining sentiment in the tweet.
- Positive Sentiment is the dominant sentiment, comprising of upwards 90% of tweets.
- Sportsense recognized 92% of touchdowns, 75% of interceptions, 74% of fumbles, 67 % of field goals in 33 games.

6. The Wisdom of Bookies? Sentiment Analysis vs. the NFL Point Spread (2010)

Stony Brook University: Yancheng Hong and Steven Skiena
Link: http://www.aaai.org/ocs/index.php/ICWSM/ICWSM10/paper/view/1527

Summary
Using sentiment analysis, they were able to identify the winner roughly 60% of the time; the prediction of the winner was better in the second half of the season compared to the first.

Takeaways
- local media is less reliable than national media due to local bias.
- Social media is just as informative as professional newspaper media.

7. Modeling Public Mood and Emotion: Twitter Sentiment and Socio-Economic Phenomena (2011)

Indiana University: Johan Bollen and Huina Mao
Harvard University: Alberto Pepe
Link: http://www.aaai.org/ocs/index.php/ICWSM/ICWSM11/paper/view/2826

Summary
By adapting a traditional psychometric instrument, they were able to map out a six dimensional mood vector for each day to twitter users. The six mood states consisted of tension, depression, anger, vigor, fatigue, and confusion.

The following steps were used to prepare data for POMS scoring:
1. Separation of individual terms on white-space boundaries
2. Removal of all non-alphanumeric characters from terms
3. Conversion to lowercase of all remaining characters
4. Removal of 214 standard stop words, including highly common verb-forms
5. Porter stemming of all remaining terms in tweet.

Takeaways
- General mood matched well to large events even if delayed.

8. Event Summarization Using Tweets (2011)

Yahoo! Research, Sunnyvale, CA: Deepayan Chakrabarti and Kunal Punera
Link: http://www.aaai.org/ocs/index.php/ICWSM/ICWSM11/paper/view/2885

Summary

Using Hidden Markov Models, they explored the ability to summarize event-based tweets. They assess highly structured and recurring events such as football games. They apply a summarization method on each tweet, and use time intervals to separate events. Their algorithm also has difficulty with proper names.

Takeaways

- Events are bursty.
- subevents can also occur within close temporal proximity.
- This paper focuses on generalizing learning a language model to identify the event, but within the case of our project, we have a set of well defined key terms.
- Event detection on scoring plays are easier to find.

9. TwitInfo: Aggregating and Visualizing Microblogs for Event Exploration (2011)

by Adam Marcus, Michael S. Bernstein, Osama Badar, David R. Karger, Samuel Madden, Robert C. Miller
MIT CSAIL, Cambridge MA
Link: http://dl.acm.org/citation.cfm?id=1978975

Summary

They utilize a novel streaming algorithm to identify peaks of high tweet activity and label them; This also allows for the drill down to subevents. . Twitinfo utilizes signal processing literature. The algorithm requires that users first define the event by providing a keyword query.

Takeaways

- Also utilizes the tweet post rate for identifying the peak.
- Served mostly as an exploratory tool.
- Users that evaluated the visualization did not trust the sentiment analysis.

Software

Data Acquisition

https://github.com/thegilby/pigskin/tree/master/scripts/data_acquisition

These programs were developed and run to gather the necessary twitter information needed to make analysis and visualization possible**:**

TweetGetter.java and screen_name_to_location_just_fbv3.java were developed to ensure that gather tweets in a standardized format that would make analysis and visualization thorough and easy.

Running TweetGetter.java and screen_name_to_location_just_fbv3.java require the twitter4j package

These files will generate text files that can be named, as needed, by editing the 'fstream' variable in TweetGetter and the 'keystream' variable in screen_name_to_location_just_fbv3.

Data Analysis
https://github.com/thegilby/pigskin/tree/master/scripts/data_analysis

These programs took the unfiltered information gathered in the previous phase (data acquisition) and were used to provide greater value and refinement of the data**:**

The data analysis files, index_tweets.py and test_gt.py, require access to the weighter.py class and the 'Total_term_list' text file.

Test_gt.py will generate an output that shows the basic effectiveness of the term list and weighting process in relationship to the 'Ground_Truth' text file. Index_tweets.py takes the term list giving and evaluates all tweets that have been collected and stored prior to the program's running.

All pig scripts require the 'tutorial.jar' used to run the Pig Tutorial as well as access to the indexed data. The pig scripts are designed to be able to be run separately and independent of one another. The large majority of the pig scripts generated

are used to perform chi-square tests to determine the trending terms of the day, in relation to football specific tweets, non-football specific tweets, and the total corpus of tweets gathered.

The weighter Python class was developed in order to weight tweets in relation to a pre-generated term list. Each term, was also assigned a 'football association' that allowed us to simultaneously weight a tweet and understand that tweet's relationship to the teams involved.

Test_gt.py was built to evaluate the effectiveness of the weighter class in relationship to a pre-determined ground truth. It generates an output that is used to evaluate how many football related tweets we can identify correctly. Index_tweets.py was used to build the files necessary for indexing in our Solr database. Column headers were added to the data as well as each tweets football weight and team association (if any).

Pig scripts were developed to perform chi-square tests and other related analyses to the weighted data.

Geolocated tweets were gathered using pig script.

Chi-square tests were performed on every week's tweets in relation to the rest of the corpus. Each week was separated into football and non-football tweets and these tweet bodies were compared against the rest of the corpus using a chi-square test. The resulting list of terms was shortened to the top 100 and displayed as needed for the website.

Total football related counts by team were also gathered and used to determine team popularity by week and in total.

Flask and Solr

While we used Pig to perform most of our heavy analysis, we used the search engine Solr and python micro-framework, Flask to serve our data and website. Solr is a full-text search engine based off of Lucene. Our advisor has recommended that

we use lucene to quickly search and serve our data without having to fully build out our database. Solr utilizes an XML schema to define the data types and index. By using Solr's query language, we can quickly filter across multiple categories and easily generate dynamic JSON files in the format needed for our visualizations.

We did run into some difficulty with hosting a live dynamic version of the website since Solr requires a sizeable chunk of memory; the costs for hosting such a server were a little bit outside of our budget. Our compromise to get a website up was to save the expected outputs in static json files that we serve as needed.

python Solr.py -ca -f <filename>
Solr.py is the python script that indexes our data into Solr.
-ca clears the index
-f file to be indexed, expects a first header row and an "id" column

pigsearch.py
This file is the api which feeds the necessary json's we use to the webapp. Functions that start with output<> and are not a part of the FootballIndex class are used to generate our static JSON files.

Data Visualization
Using the static JSON files from Flask and Solr, we were able to use the d3.js and Rickshaw.js libraries to display our data in a visual manner. Each JSON file is loaded from the backend through Flask when a particular page is called. Each page of the website is templated using Flask and can be found in /templates/ on our Github page.

/static/js/main.js (on Github) houses all of our application specific Javascript to display the Rickshaw graphs, Google maps, and interactions.

Data

Link: https://www.dropbox.com/sh/pgwj8piatvu0ys5/xLwql2q5P-
The dataset includes 7 separate days worth of tweets, indexed and ready to upload into our Solr database.

Work Percentages

	Research	Backend Coding	Frontend Coding	Presentation & Report
Andrew	75%	20% Solr, Pig, Java	20% Flask, Google Maps	30%
Gilbert	15%	5% Pig	75% Rickshaw, Flask, Bootstrap, Google Maps	40%
Jacob	10%	75% Pig, Java	5% Flask	30%

Appendix A - References

1. Mislove, A., Lehmann, S., Ahn, Y. Y., Onnela, J. P., & Rosenquist, J. N. (2011, July). Understanding the demographics of twitter users. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media (ICWSM'11), Barcelona, Spain*.
2. Lanagan, J., & Smeaton, A. F. (2011, July). Using twitter to detect and tag important events in live sports. In *Proceedings of AAAI*.
3. Zhao, S., Zhong, L., Wickramasuriya, J., & Vasudevan, V. (2011). Human as real-time sensors of social and physical events: A case study of twitter and sports games. *arXiv preprint arXiv:1106.4300*
4. Han, B., & Baldwin, T. (2011, June). Lexical normalisation of short text messages: Makn sens a# twitter. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*(Vol. 1, pp. 368-378).
5. Zhao, L. Z., Wickramasuriya, J., & Vasudevan, V. (2011). Analyzing twitter for social tv: Sentiment extraction for sports. In *Proceedings of the 2nd International Workshop on Future of Television*.
6. Hong, Y., & Skiena, S. (2010). the Wisdom of Bookies? Sentiment analysis vs. the nFL Point Spread. In *Proceedings of the international conference on Weblogs and Social media (icWSm-2010)*.
7. Bollen, J., Pepe, A., & Mao, H. (2011, July). Modeling public mood and emotion: Twitter sentiment and socio-economic phenomena. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media* (pp. 450-453).
8. Chakrabarti, D., & Punera, K. (2011, May). Event summarization using tweets. In *Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media* (pp. 66-73).
9. Marcus, A., Bernstein, M. S., Badar, O., Karger, D. R., Madden, S., & Miller, R. C. (2011, May). Twitinfo: aggregating and visualizing microblogs for event exploration. In *Proceedings of the 2011 annual conference on Human factors in computing systems* (pp. 227-236). ACM.