**MAPPING HAPPINESS AND FRUSTRATION INDEX ON TOP OF THE WORLD MAP THROUGH TWEETS**

**Karthikeyan Murugesan\*, Raja V Palanisamy***A* **& Chandra P Khatri***A*

\* *School of Biology, Georgia Institute of Technology*

*A School of Computational Science & Engineering, Georgia Institute of Technology*

**INTRODUCTION**

Microblogging sites are a continuous source of diverse user generated data; spanning from politics, weather, the entertainment industry, and user sentiments, up to random unclassifiable babble. Twitter is one such site that has grown paranormal in this field with providing a rich repertoire of information to study community behavior and user sentiments. Once tapped, this data could be analyzed semantically to delimit emotional polarities from each other. Furthermore, a succinct visual representation of the data in real time would highly appeal to the user to follow event detection patterns in the form of sentiments.

Hence in the current study, we have made a framework through which we obtain live feeds from twitter in real-time. Then through this framework we mine the data obtained and perform natural language processing, text classification and sentiment analysis on it. After cleaning and obtaining the data (sentiments and their locations) through this framework , we eventually map ‘happiness’ and ‘frustration’ on top of the World Map.

The main motivation for our work was the visualization of retweets of the Oscar

selfie - #Oscar 2014 that was generated by Twitter using CartoDB tools . (<http://mashable.com/2014/03/04/see-all-the-oscar-night-twitter-activity/>)

The visualization runs for a less than a minute to beautifully capture the 4 hour retweet fest .The manifestation of this visualization appeal is that it shows clearly how and where the retweets happened throughout the globe.

**LITERATURE REVIEW**

People as part of a large community are acting as sensors and generating lots of meaningful information. Recently, researchers have developed efficient techniques to infer sensible information from social-media during any crisis such as earthquakes and mass emergency [2,3,4]. Twitter data has been used for predicting stocks [1]. It has been shown that people switch to social-media, particularly Twitter and Facebook during many incidents .Shulz et al.[5] have analyzed these incidents in real-time on a small scale using Microblogs. Riberio et al., [6] have recently implemented the system to observe the events and conditions using Twitter in Portuguese.

Bontcheva and Rout[7] have surveyed all the existing technologies and research done so far in regards to analyzing the social media streams, particularly Twitter.

Eventually processing and visualizing the data is important too . Marcus et al.[8] have depicted their methods and tools which can be used to visualize the data obtained through tweets.

Although, in the past two years, researchers have started using Twitter as a source to obtain the data for information retrieval, many of the above mentioned studies are not for real-time data. Most of the existing applications and tools are for static data (historic data). Furthermore, their approach is not wholesome because most of them confine only to some local geographical region and therefore include parameters which fit in best for those regions.

Sentiment analysis is a hard and tricky concept that is widely being pursued by researchers across the world. Sentiment viz[9] is an application on similar lines which estimates and visualizes sentiments from twitter feeds , but it lacks a concrete natural language processing filter for slangs and sarcasm. SentiGraph, CyberEmotions project[10], Sysomos, Radian6[11] are other tangentially relevant commercial social media monitoring applications which focus mainly on web analytics of product brands and not on sentiments per se. This field has not progressed much in non-english languages either. Recently, researchers have tried making sense of social-media streams with the help of NLP and Semantics [12][13] Studies specifically dealing with Twitter streams are of our interest[14][15]

In our case, we created our own database of words, which is very specific to our objective i.e. the dictionary has a specific set of words which relate to happiness/frustration with happiness index for each word. Then we performed NLP and customized the computations for better results.

**PROPOSED METHOD**

Intuition:

As explained in the literature review before, the state-of-the-art methods are not dynamic and real-time. Most of the visualizations are static wherein one can’t clearly identify the variation in happiness level of various countries over time.

Dynamic visualization can give us a feel about the effect of time (morning/noon/night) on the happiness level and also has a great potential to see the spread of viral posts and also explain their effect on the emotions of tweeters

Tools Used:

The technologies which we are using for the current study are:

* *Python:* For obtaining, mining and processing data throughout the project.
* *Tweepy Python Library:* Used for obtaining structured data from Twitter, based on various parameters (Location, keywords).
* *SQLite:* For storing key-words and their sentiments scores. Acts as a dictionary, where words are the key and sentiment scores are the values.
* *Natural Language Toolkit (NLTK):* Used for tokenization and entity extraction from the data obtained.
* *CartoDB:* Open source tool for visualization of geospatial data. Various dependencies of cartodb are listed later

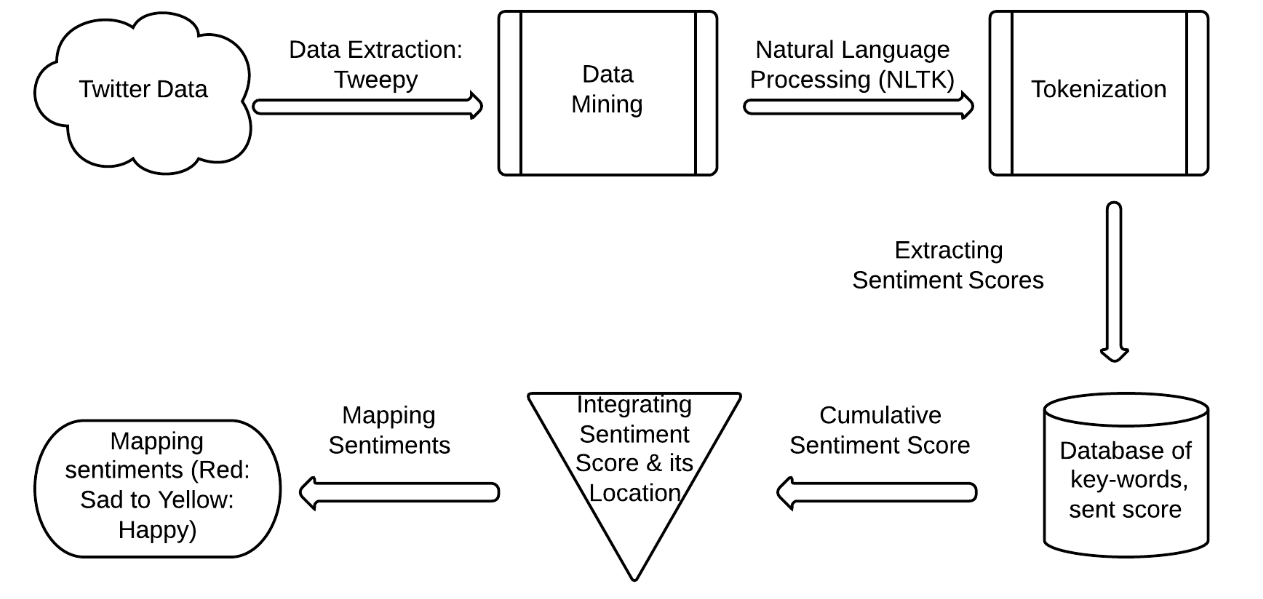


Figure 1: Methodology

**DATA EXTRACTION AND MINING**

We used the Tweepy Python library to extract the data from Twitter in a structured format. This results in latency issues and limits on client requests[16]. Not all tweets are precisely geo annotated [17][18][19] and the geo annotation of the such tweets depends on self-reported location fields.

Tweepy uses Twitter Rest and Twitter Search APIs for obtaining the data. To access the data, a user needs to generate access and token keys. These keys are then integrated with the Tweepy Python library. Tweepy returns the data in JSON format. A lot of related information (user\_name, user\_id, time, location, user\_profile details) are also retrieved along with Tweets. For our current study, only the geo-location and the tweet fields are relevant.

*Keyword Database:*

* + 2500 words with both positive and negative sentiments
  + Assigning weights to each word based on its sentiment

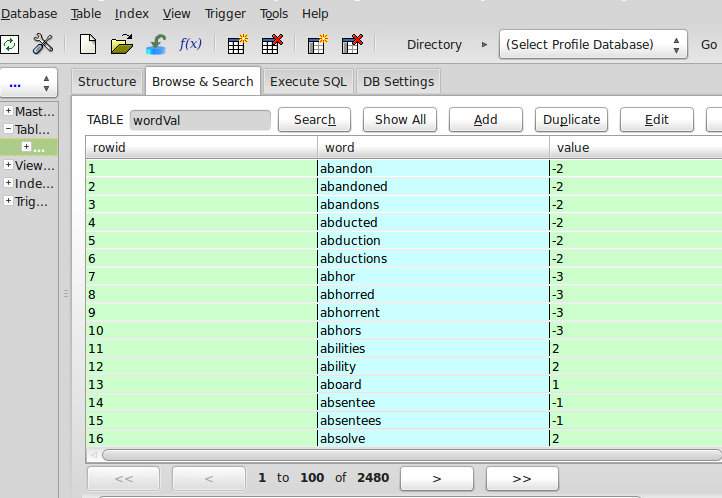


Figure 2: Database of words and their sentiment scores

*Computation of Index:*

* + Natural Language Processing: Tokenizing each word, key word extraction from the Tweets obtained using *NLTK*
  + Sentiment Analysis

- mine keywords in tweets,

- obtain weights of each word from

the database

-generate cumulative sum of sentiments for each tweet

* + For each tweet of interest, obtaining overall sentiment and geo-location as the final output

*Tweepy Implementation:*

We have designed the framework in such a way that we process the data as soon as we receive it, hence we don’t need to save it. Every time a Tweet is obtained, we tokenize it and chop the words. Stop words (e.g. “the”, ”a”, ”is”, ”that” etc.) are removed. Remaining words are then searched in the already existing database of key-words, and if the word exists then its sentiment score is obtained. This is done for all the words in obtained Tweet. Eventually, a cumulative sum of score (Sentiment) is appended with the geo-location.

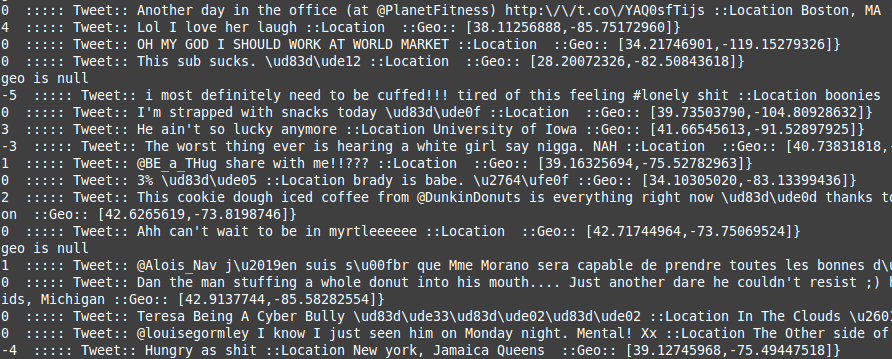


Figure 3: Mined Data (Sentiment Score, Tweet, Geo Location)

Figure 2 represents the mined data. The sentiment score (Happiness Index) is indicated on the left and the geo-location is indicated on the right.

**Use Case:**

Tweet: “Happy by Pharrel Williams! Too good.”

Sentiment = Cumulative Sent score = 3 (Happy) + 0 (by: stop word) + 0 (Pharrel/Williams: Proper Noun) + 0 (Too: stop word) + 2 (Good)

Senti Score = 5

**DATA VISUALIZATION**

We use an open-source tool called CartoDB[22] which is specifically used for mapping geo-spatial data. It has the following components,

* A User Interface
* A database for storing geospatial data
* SQL API for SQL queries over HTTP
* A Map tiler

CartoDB is written in Ruby and has plenty of dependencies, which are described in the next paragraph. CartoDB can be run on a local server with an user-interface to upload, edit and visualize the geo-data. It can also be run on their website [22] which has additional features for uploading the data. Its advantageous to use CartoDB instead of any other visualization software because it offers dynamic visualization of the data. Moreover, if we set the program to automatically update the data table, then the visualization can be made real-time as well.

*Framework for running the application on local server*

The CartoDB application interface required the installation of a variety of software packages,libraries and dependencies. The application is available open source with the source files on github[23] (<https://github.com/CartoDB/cartodb>)   
  
The following dependencies had to be installed for CartoDB to run locally -

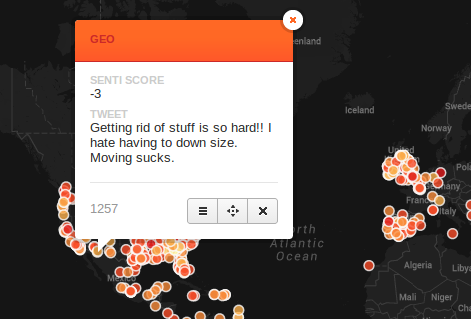
Table 1: Dependencies for CartoDB

|  |  |
| --- | --- |
| CartoDB-SQL-API | This component powers the SQL queries over HTTP. |
| GDAL 1.10 | Raster support |
| GEOS 3.3.4 | Geometry function support |
| Mapnik 2.1.1 | API for creating and styling map tiles |
| NodeJS 0.8 | Tiler API and SQL API |
| PostGIS 2.0.x | Geospatial extension for PostgreSQL |
| Postgres 9.1 | Relational database |
| Redis 2.2 | Required for Windshaft & SQL API |
| Ruby 1.9 | Language that CartoDB uses |
| Varnish 2.1 | Web application accelerator |
| Windshaft | Powers CartoDB Maps API |

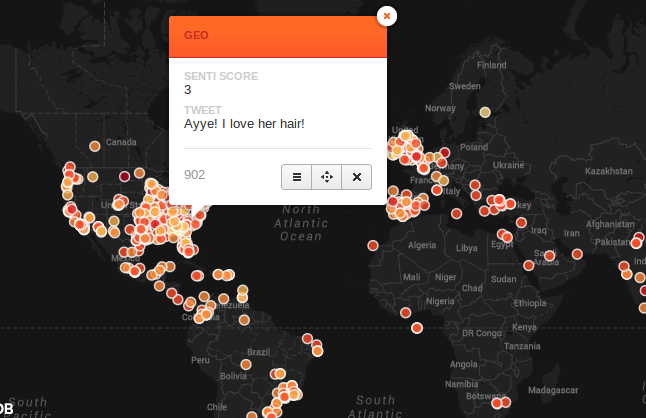
With interface on a local server or the website, one can create, upload, edit and visualize the geospatial data

**Visualization**

In order to have a perspective of the visualization on CartoDB, we developed a static version of the visualization with the georeferenced data and represented the tweets with their Sentiment index on the world map.



4(a): A sad tweet



(b): A happy tweet

Figure 4: A static CartoDB visualization

The chloropleth visualization wizard of cartoDB was used to generate these static visualizations. It uses a customizable colour ramp gradient to color the points based on the field’s corresponding senti\_score.   
 The dynamic visualization was created using another viz feature of CartoDB called Torque. It renders time-series data by converting the input data into a ‘layercube’ format and animates it over a google maps layer . Our dynamic visualization can be found at <http://cdb.io/1irs2jq> . Figure 5 is an illustration of the actual dynamic visualization , of our application

Various parameters of the visualization can be configured using CartoDB’s custom CSS feature called CartoCSS.One can adjust the radius of the points , color of the blobs,point border thickness and its color , point shape,blend mode,duration and frequency of the flicker. The parameters can be adjusted using CartoCSS or through the visualization UI that CartoDB provides

**EXPERIMENTS**

**1.)Happiness and frustration statistics**

Our experiment consist of a visualization of few minutes of twitter data taken at around 1 pm EST on the 27th of April obtained from approximately 1000 Tweets. Our plan of the experiment is to infer the following from the visualization:

1. Happiness-frustration ratio by continent: Obtained by calculating positive and negative sentiments and taking their ratio.

North America - 22:1

South America - 35:1

Europe- 12:1

Asia- 10:1

Africa- 7:1

Australia: No data available during the time of study

Figure 5 represents a snapshot of the visualization taken at 1pm , April 27th. The yellow dots represent the happy tweets and red dots represent the frustration tweets.

The following table details the results which we obtained during the time period of sampling



**Figure 5: A snapshot of the visualization at 1pm EST , April 27th**

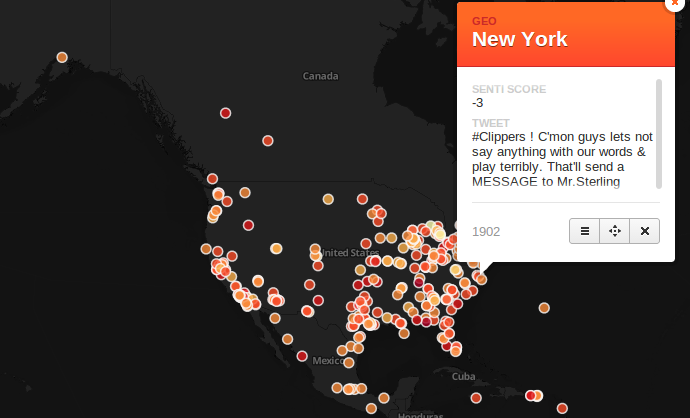
**2.)Sentiments on trending topics**

**#Clippers vs #Warriors**

The following analysis was done to observe real time twitter emotions on the Los Angeles Clippers vs Golden State warriors game (27th April , 2014)

A tweet filter was used to filter out tweets from the United States that were associated with #Clippers and #Warriors and these were then visualized using our application.

We observed that the general emotion was negative amongst the tweeters across USA and many of the tweets contained keywords like ‘racism’ and ‘protest’ .This made sense as a major furore was caused previously due to alleged racist remarks by Mr.Donald Sterling , owner of the Clippers .



**Figure 6: A snapshot of the tweet frenzy #Clippers vs #Warriors**

**POINTS TO IMPROVE UPON**

1) Effective natural language processing filters for sarcasm

2) Currently the visualization is binary i.e Yellow for “Happy“ and Red for “Frustration” The gradient in sentiments can also be mapped to a corresponding color gradient

3) Sentiment analysis for non english languages yet to be incorporated

**DISCUSSIONS and CONCLUSIONS**

Visualizing the happiness/frustration over time gives us a key idea about the emotional nature of social media trends.This process has great capacity to help us analyze brand analytics predict stock variation,advertising campaigns,celebrity news , trending videos/posts , poll opinions and many other interesting time varying trends.

With all the above experiments, we see that in essence our key selling points are an extensive database of keywords for sentiments, location based ‘sentiment mapping’ and a real time dynamic visualization.

**REFERENCES**

1. Raquel Meyer Alexander, James K. Gentry: Business Horizons Volume 57, Issue 2, March–April 2014, Pages 161–167 **Using social media to report financial results**

2. L. Palen, K. Anderson, G. Mark, J. Martin, D. Sicker, M. Palmer, and D. Grunwald (2010). **“A vision for technology- mediated support for public participation & assistance in mass emergencies & disasters.”** In Proceedings of the 2010 ACM-BCS Visions of Computer Science Conference.

3. Sudha Verma, Sarah Vieweg, William J. Corvey, Leysia Palen, James H. Martin, Martha Palmer, Aaron Schram, Kenneth M. Anderson, (2010) **“Natural Language Processing to the Rescue? Extracting "Situational Awareness" Tweets During Mass Emergency”**, AAAI Publications, Fifth International AAAI Conference on Weblogs and Social Media

4. Soudip Roy Chowdhury, Sihem Amer-Yahia, Carlos Castillo Muhammad Imran Muhammad Rizwan Asghar, **“Tweet4act: Using Incident-Specific Profiles for Classifying Crisis-Related Messages”** (2013), The 10th International Conference on Information Systems for Crisis Response and Management (ISCRAM)

5.Axel Schulz and Petar Ristoski, (2013) **“The Car that Hit The Burning House: Understanding Small Scale Incident Related Information in Microblogs”**, AAAI Technical Report WS-13-0 10.

6. Sílvio S. Ribeiro, Jr., Clodoveu A. Davis, Jr., Diogo Rennó R. Oliveira, Wagner Meira, Tatiana S. Gonçalves Jr. Gisele L. Pappa Jr. (2012) **“Traffic observatory: a system to detect and locate traffic events and conditions using Twitter”**

7. Kalina Bontcheva a; Dominic Rout (2012), **“Making Sense of Social Media Streams through Semantics: a Survey.”** Semantic Web 0 (0) 1 1 IOS Press, Semantic Web journal, (2012)

8. Adam Marcus, Michael S. Bernstein, Osama Badar, David R. Karger, Samuel Madden, Robert C. Miller, (2013) **“Processing and Visualizing the Data in Tweets”**

9. [www.csc.ncsu.edu/faculty/healey/tweet\_viz/](http://www.csc.ncsu.edu/faculty/healey/tweet_viz/)

10. [www.cyberemotions.eu/](http://www.cyberemotions.eu/)

11. [www.eurekalert.org/pub\_releases/2013-09/uosc-ta090313.php](http://www.eurekalert.org/pub_releases/2013-09/uosc-ta090313.php)

12. Bontcheva, K., & Rout, D. (2012). **Making sense of social media streams through semantics: a survey**. *Semantic Web*.

13. Ghiassi, M., Skinner, J., & Zimbra, D. (2013). **Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network**. *Expert Systems with Applications: An International Journal*, *40*(16), 6266-6282.

14. Hienert, D., Wegener, D., & Paulheim, H. (2012). **Automatic classification and relationship extraction for multi-lingual and multi-granular events from wikipedia**. *Detection, Representation, and Exploitation of Events in the Semantic Web (DeRiVE 2012)*, *902*, 1-10.

15. Go, A., Bhayani, R., & Huang, L. (2009). **Twitter sentiment classification using distant supervision**. *CS224N Project Report, Stanford*, 1-12.

16. Marcus, A., Bernstein, M. S., Badar, O., Karger, D. R., Madden, S., & Miller, R. C. (2012). **Processing and visualizing the data in tweets**. *ACM SIGMOD Record*, *40*(4), 21-27.

17. www.[social-media-monitoring-review.toptenreviews.com/](http://social-media-monitoring-review.toptenreviews.com/)

18. Mahmud, J., Nichols, J., & Drews, C. (2012, May). **Where Is This Tweet From? Inferring Home Locations of Twitter Users**. In *ICWSM*.

19. Hecht, B., Hong, L., Suh, B., & Chi, E. H. (2011, May). **Tweets from Justin Bieber's heart: the dynamics of the location field in user profiles**. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 237-246). ACM.

20. Zanzotto, F. M., (2011, July). **Linguistic redundancy in twitter**. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 659-669). Association for Computational Linguistics.

21. Wu, W., (2010, June). **Automatic generation of personalized annotation tags for twitter users**.

22. [www.cartodb.com](http://www.cartodb.com)

23. <https://github.com/CartoDB/cartodb>