CS 690V – VISUAL ANALYTICS
HOMEWORK 7

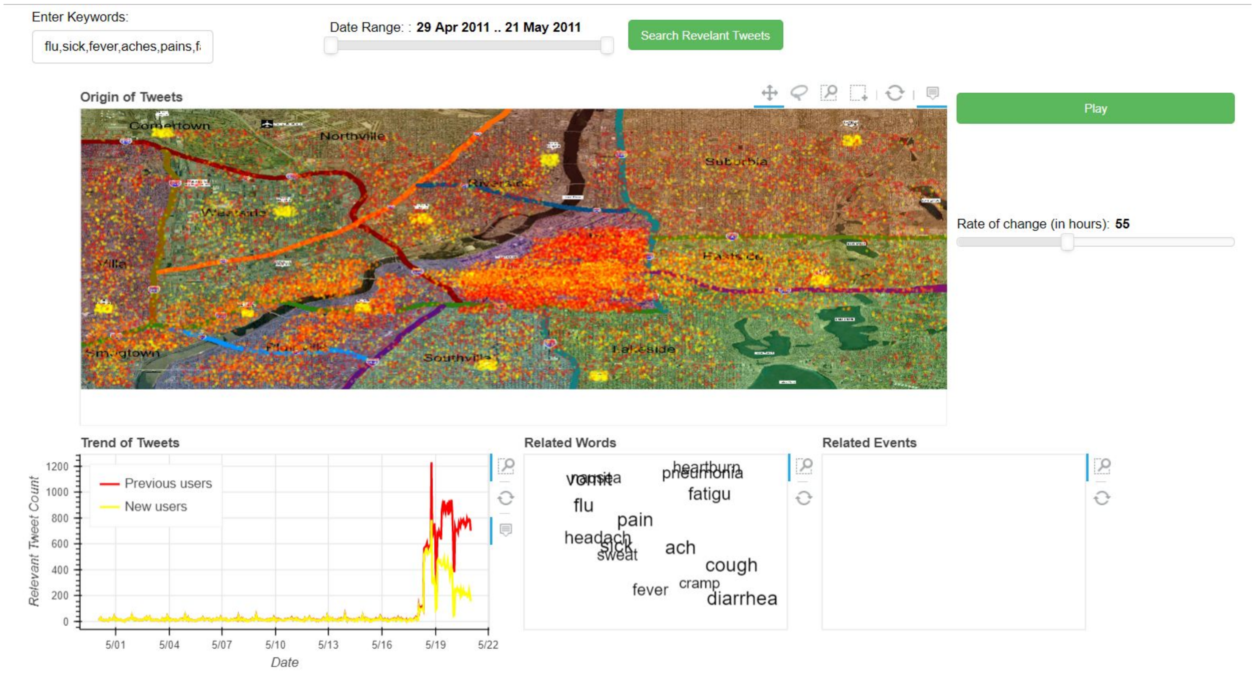
Submitted by:

- Suhas Keshavamurthy
- Kriti Shrivastava

VAST Challenge 2011: MC 1- Characterization of an Epidemic Spread

http://hcil2.cs.umd.edu/newvarepository/VAST%20Challenge%202011/taskdescription-of-all2011challenges-printfromoriginalwebisteofchallenge.pdf

We created a dashboard that would enable us to solve this mini-challenge. User can search for tweets related to some keywords, within a particular time range and location using this dashboard. She can also find events related to the keywords by selecting a region on the map.



Enter Keywords: flu,sick,fever,aches,pains,f

Date Range: : 29 Apr 2011 .. 21 May 2011

Rate of change (in hours): 55

Figure 1. Overview of the system

MC 1.1 Origin and Epidemic Spread: Identify approximately where the outbreak started on the map (ground zero location). If possible, outline the affected area. Explain how you arrived at your conclusion.

Solution Approach:

1. *Preprocessing*: The following tasks were performed to preprocess the data-
   a. Remove the URLs and emojis from the tweet text.
   b. Remove stop words from the tweet text.

    c. Remove punctuation from the tweet text.
    d. Convert all words to lowercase words.
    e. Tokenize and extract words from the tweet text.
    f. Perform stemming on each word of the tweet.

2. *Extract relevant tweets*: After getting the tweets in required format, we filtered out the tweets relevant to our problem. For this we used the symptoms mentioned in the challenge description as our primary keywords to filter data on. For example, flu, sick, fever, pain, fatigue, cough, vomiting, diarrhea were some of the chosen keywords. We then performed the following steps-
    a. Stem the keywords: Since the words in our tweet dataset are stemmed, we stemmed the keywords as well. The main intention behind this was to include cases like- "I have been coughing all day" or "I have a terrible cough.". Stemming would help us extract both these tweets, otherwise we would have missed the tweet with the word "coughing" instead of the exact keyword "cough".
    b. Add similar words to the keyword search list: For all the words in our list of keywords/symptoms we found a list of similar words using the Word2Vec model. Word2vec is a group of related models that are used to produce word embeddings. These models are shallow, two-layer neural networks that are trained to reconstruct linguistic contexts of words. Word2vec takes as its input a large corpus of text and produces a vector space, typically of several hundred dimensions, with each unique word in the corpus being assigned a corresponding vector in the space. Word vectors are positioned in the vector space such that words that share common contexts in the corpus are located in close proximity to one another in the space. We also experimented using the WordNet synset model but observed that Word2Vec performs better and thus decided to chose Word2Vec to solve this challenge.
       Performing this step helped us retain the tweets which might not contain the exact word as our keyword but contain words with a similar meaning. For example, "I fell ill today." and "I am feeling very sick.", both these tweets are relevant. We have the word "sick" in our search list but not the word "ill" but both are similar in meaning. Therefore, we need to add words with similar meanings to our list of keywords.
    c. Filter tweets: As a last step, we extract the tweets which contain one or more of our keywords. Out of more than a million tweets, we extracted around 55000 tweets for the default search terms that we used from the problem description.

3. *Plot relevant tweets on the map*: Once we have the list of relevant tweets, we plotted them across the city map using their latitude and longitude values. We could see the pattern with high number of tweets being concentrated over 5 regions: Downtown,

Uptown and Eastside and, Smogtown and Plainville. With this we were able to outline the affected areas on the map.

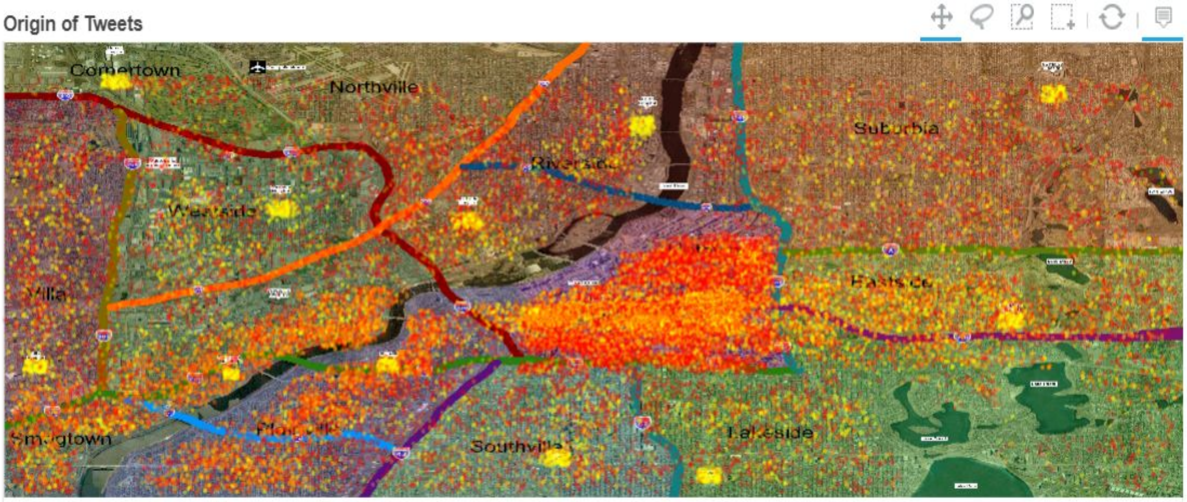


Figure 2. Plotting relevant tweets on the map. Red shows tweets by people who have tweeted about the topic before and the yellow tweets are by new people.

4. *Get the trend in outbreak:* We also found the number of relevant tweets seen each hour and plotted it across time. With the help of this line graph, we observed that there is a spike in the number of cases on 18th and the number of tweets are significantly high over the next few days. Thus we were able to find the time period when the outbreak started to spread.

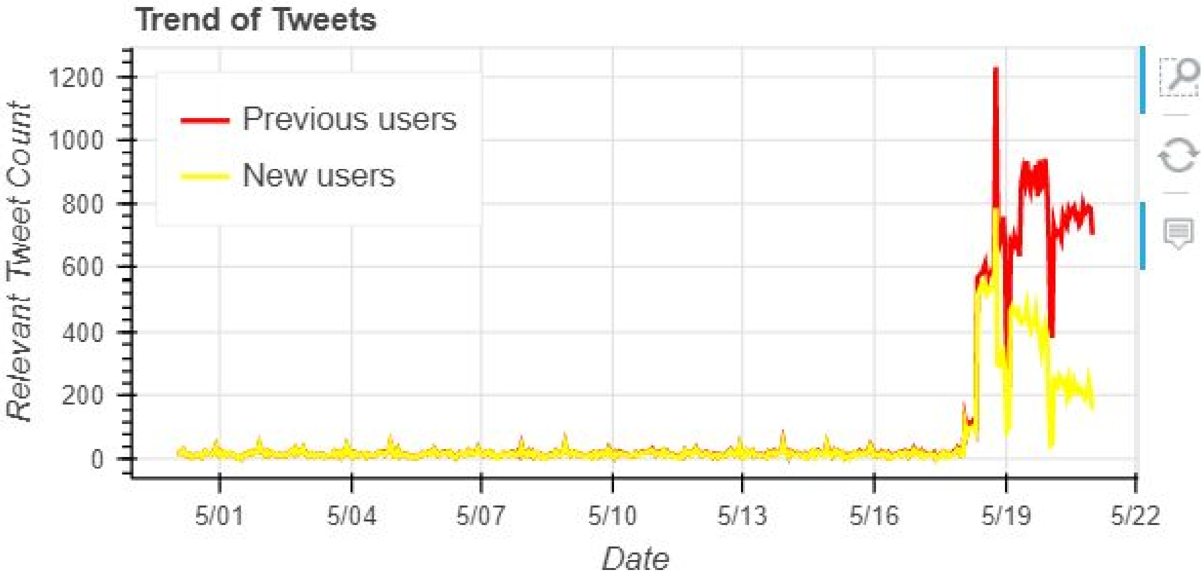
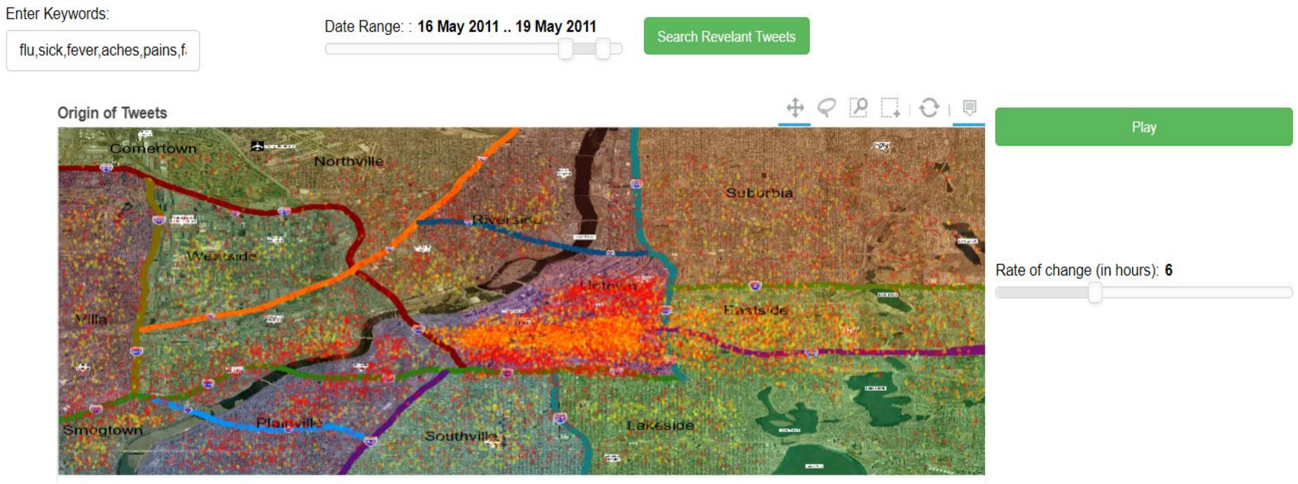


Figure 3. Trend of tweets over time

5. *Filter on outbreak start date:* Since now we have the affected area and date when the outbreak started, we try to narrow down the cause. Using the time slider, we select the

date from 2011-05-16 - 2011-05-19. Using the play button, we observed the rising spread of the disease in the affected regions. We see two distinct regions of disease spread. One is along the river, Plainville and Smogtown and the other from Downtown to Uptown and Eastside. The intersection of these regions could be the possible ground zero location.



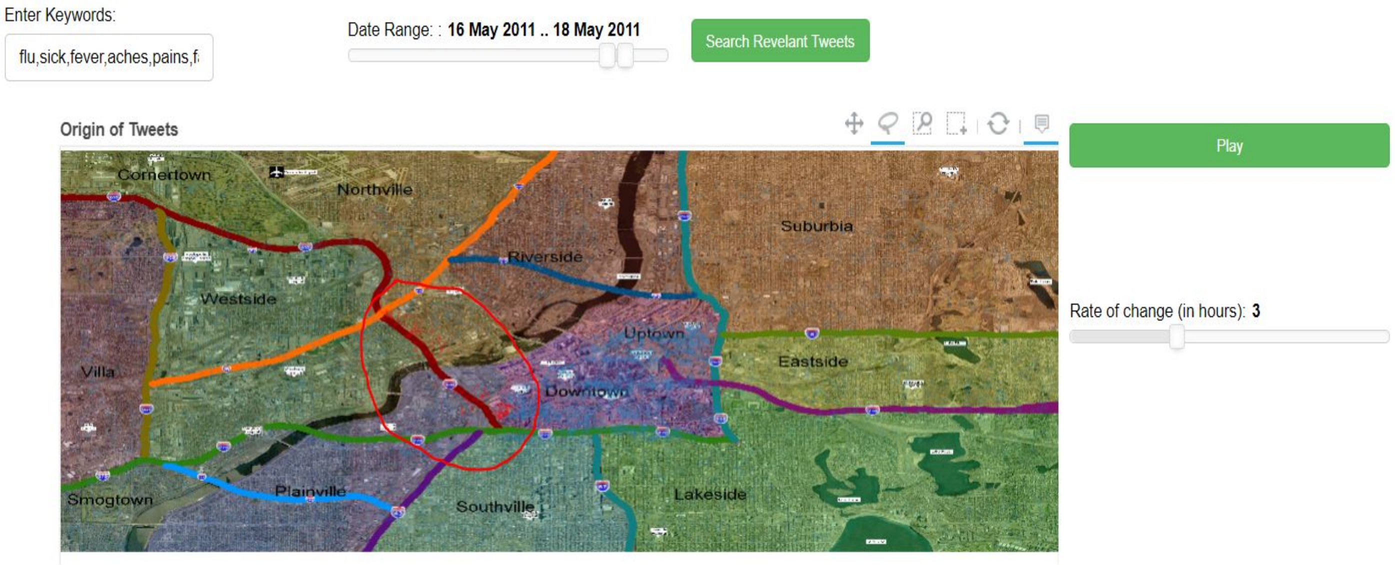
Enter Keywords: flu,sick,fever,aches,pains,f

Date Range: : 16 May 2011 .. 19 May 2011

Rate of change (in hours): 6

Figure 4. Filtering on date to find affected regions and approximate start date

6. *Select approximate area of origin:* We decided to closely observe the tweets originated from the location from step 5 around 2011-05-17. For this, we selected the approximate area of origin on the map using lasso select tool(marked using red pen in the below screenshot) and filtered the tweets using the date.



Enter Keywords: flu,sick,fever,aches,pains,f

Date Range: : 16 May 2011 .. 18 May 2011

Rate of change (in hours): 3

Figure 5. Selecting approximate area of origin on the map

7. *Get possible causes of origin:* Once we had the subset of tweets with specific location and date, we decided to further analyze them to identify possible causes of the outbreak. We tokenized and performed part of speech tagging on the tweet text. To find suspicious

events and causes, we then extracted the most frequent nouns that occur in these tweets. On selecting the location using lasso tool on the map, the most interesting words that pop up are as shown below.

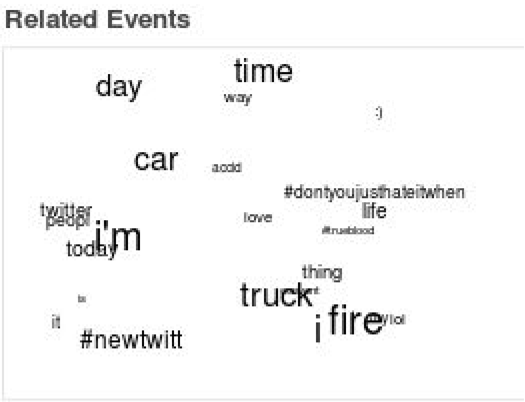


Figure 6. Finding possible events near the approximate area of origin

8. *Narrow down on the most probable cause:* Ignoring trivial words like day, it etc, there are 3 major events in the area in the given time period. We investigate each of these events further. One of them turns out to be a bomb drill in downtown. The second is a car accident on Highway 270. The third is a truck accident on highway 610. After looking at the tweets for the three incidents above and searching for relevant words and events for each of these three events, we can infer that the truck accident is the likely cause. We filter the tweets further, now using the keywords like "truck", "accident", while still filtering on time and location. We get a list of approximately 500 tweets which are about the truck accident. The location of the accident can be narrowed down to I-610 Bridge over the VAST River.

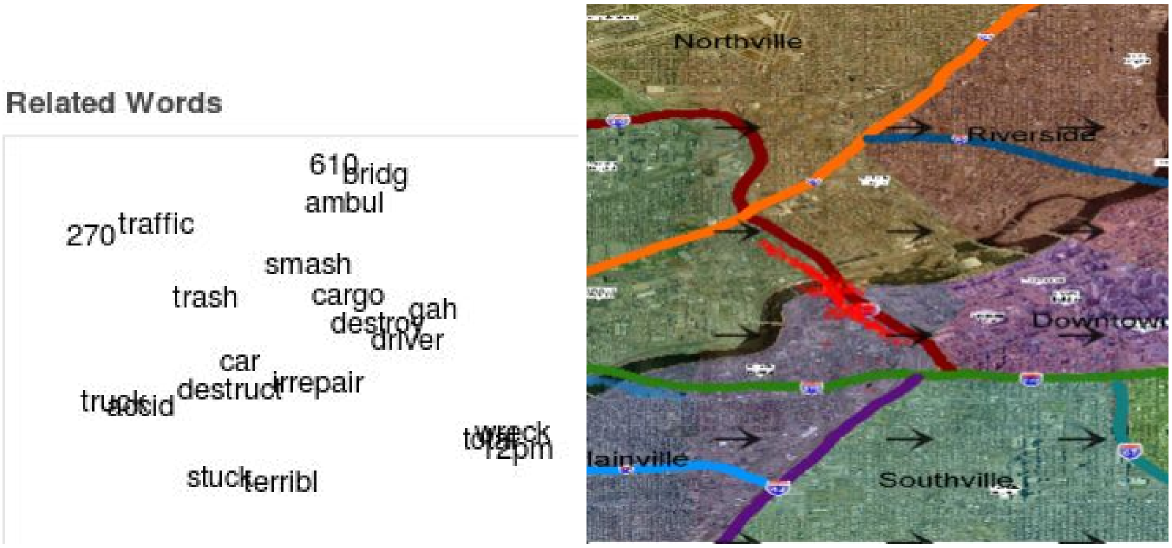


Figure 7. Finding the most probable cause and the words related to it

To conclude, using NLP techniques and interactive visualizations, we were able to locate the ground zero location for the outbreak successfully.

MC 1.2 Epidemic Spread: Present a hypothesis on how the infection is being transmitted. For example, is the method of transmission person to person, airborne, waterborne, or something else? Identify the trends that support your hypothesis. Is the outbreak contained? Is it necessary for emergency management personnel to deploy treatment resources outside the affected area? Explain your reasoning.

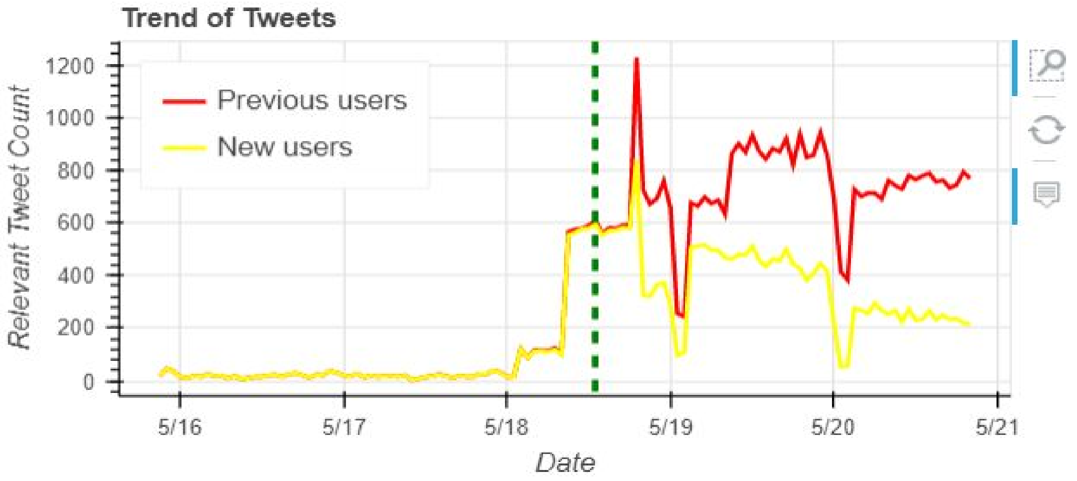


Figure 8. Timeline for epidemic

As discussed previously we see two distinct regions apart from Downtown where the disease spread is observed clearly.

From the visualization we observe initially the spread of disease towards east direction primarily 'Eastside' region. This also corresponds to the wind direction that was reported for that day in the weather data. The arrows on the map show the wind direction at that time in the area. We observe that the direction of wind is similar to the direction of epidemic spread. This gives us a clue to look for place of origin to the west of affected area. Selecting the tweets in the 'Eastside' region also confirms the suspicions of airborne disease as the symptoms primarily reported in the Microblogs is mainly flu-like (headache, fever, cough etc).

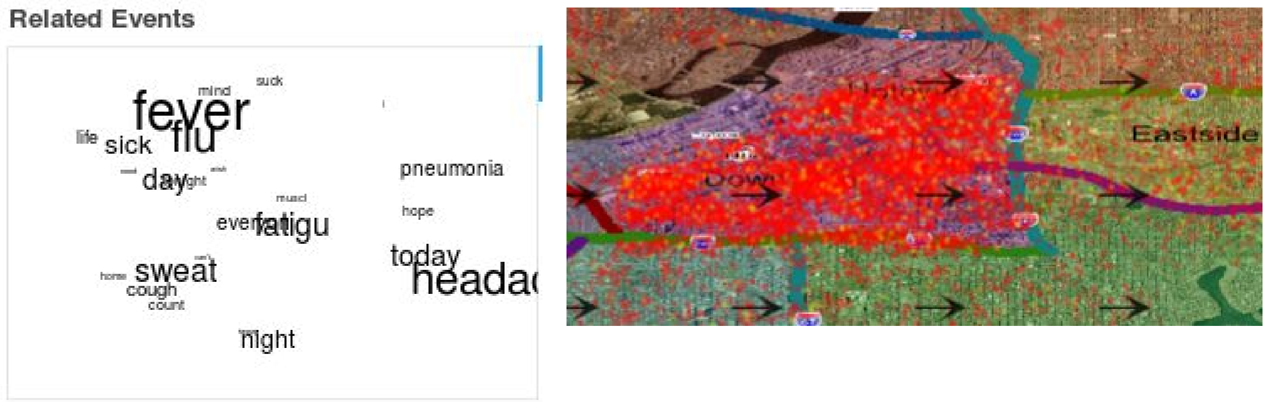


Figure 9. Spread in Region 1

We also observe spread of disease along the VAST river primarily in ‘Plainville’ and ‘Smogtown’ region later on (after 18th). Upon selecting the tweets in this region, we find that the complaints/symptoms are primarily those of waterborne disease (diarrhea, stomach, cramp, toilet, nausea etc).

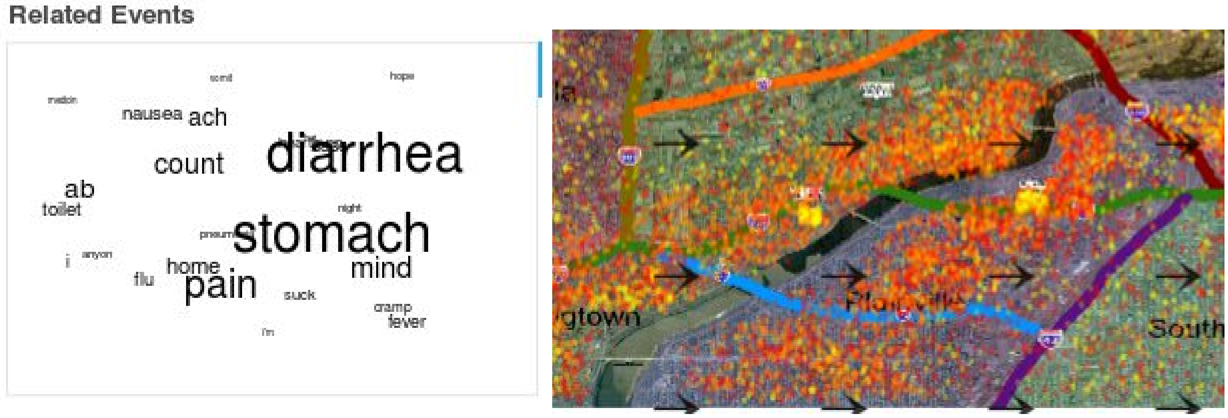


Figure 10. Spread in Region 2

From the trend graph of tweets, we observe that there was a spike in reporting of disease on 18th. There was a slump for a short time before increasing again. These two spikes can be attributed to the airborne and waterborne spread of disease. The trend graph also plots the number of new users and old users reporting in the microblogs. We can observe that the trend of new users reporting of any kind of symptoms is reducing day by day. Also from the map we do not observe any increase or trend in spread of the disease. From this we can conclude that the outbreak is contained and additional steps to control it is not necessary at the present moment. Also we observe a large concentration of microblog posts. Upon analysis, we find that this corresponds to the location of hospitals in Vastopolis. This indicates that the people are getting appropriate medical attention for their illness and additional resources is not required at the present moment.

Visualizations:

- **Time series:** Provides information regarding the count of tweets grouped by hour which contain words similar or closely related to the input search terms. The x-axis is the time of creation of the tweet and the y-axis is the count of relevant tweets for that particular hour.
- **Word Cloud (Similar words):** The word cloud provides a visual representation of the words most similar or closely related to the search term. The size of the word in word cloud depicts its similarity score using the Word2Vec model in the relevant tweets.
- **Word Cloud (Related Events):** The word cloud provides a visual representation of the nouns by its occurrence frequency in the microblogs selected by the user.
- **Geospatial Scatter Plot:** The plot displays the origin of tweet against the background of the map of the United States of America.

Interactions:

- ○ On hover over the tweet, the tweet text is displayed.
- ○ The plot can be zoomed in with the use of the Box Zoom Tool to enlarge and identify the location of tweet in a particular region.
- ○ Lasso tool and box select tool can be used to select particular microblogs and the corresponding events are generated in the Word Cloud (Related Events).

- **Widgets**:
  - ○ Text Input : Provides space for the user to enter the list of symptoms or desired keywords.
  - ○ Buttons:
    - i. Search Tweets button: Used to filter the tweets using the criteria selected by the user (symptoms and the approach). All the graphs are updated on click of this button. *Please wait for a while after clicking this button to see the new results.*
    - ii. Play/Pause buttons: to play or pause the time-series plotting in the scatter plot.
  - ○ Slider : Used to control the rate of plotting the scatter plot (speed for play button).
  - ○ DateTimeSlider : User can select a date range within which he/she would like to investigate the data more closely.

Other interesting observations:

1. Out of curiosity, we plotted all 1 million tweets on the map and saw that they are evenly distributed excluding the areas of river and mountains. It takes a decent amount of time for the plot to render with million points.
2. Microblogs frequency correspond well with the density of population in each region.

Comparison with Midterm:

For our midterm we worked with real time twitter data and performed tasks like Mini-batch K-means clustering and sentiment analysis. To solve this mini-challenge we used different approaches and hence a direct comparison on the approach for both is not feasible. While a lot of the preprocessing tasks remained the same, we felt that dealing with real time data was a lot more complex. With static data, we had the flexibility of running the preprocessing and analysis algorithms separately and storing the results for future use. This made the rendering of graphs a lot faster as compared to real time data where preprocessing and analysis was done on the fly.

Additionally the data provided for the mini-challenge was much cleaner than the real-time streaming tweets that we were observing in the midterm submission. In the midterm the text investigation was largely dependent on the user search terms and some amount of filtering was done by the Twitter API's. With static, synthetic data, there is time for the user to investigate the textual information. Also most of the real-time streaming data did not contain location information. In this mini-challenge, the location information played an integral part in solving and investigating the problem.

Bokeh Version: 0.12.10
Python Version: 3.6.0

Package Dependencies:

1. Tweet-Preprocessor: This library makes it easy to clean, parse or tokenize the tweets.
   - Install using: $ pip install tweet-preprocessor
2. Geopy:
   - Install using: $ pip install geopy
3. NLTK: NLP library
   - Install using: $ pip install nltk
4. Gensim: To use Word2Vec
   - Install using: $ pip install gensim
5. Cython:
   - Install using: $ pip install cython
6. Other packages: numpy, pandas, sklearn, random

Please NOTE that this is a Bokeh directory application. Please download the entire folder in the Google drive link provided.
To run the code use the command: bokeh serve --show HW7

Google Drive Link (Code + Data):
https://drive.google.com/open?id=1n0OJM16NcJAvfCuX9pab2XHfd0OFvhgV

References:

1) https://en.wikipedia.org/wiki/Word2vec