Twitter Disaster Analysis

Maria Mahbub, Linsey S. Passarella, Emily Joyce Herron and Gerald Leon Jones

Main questions and Goals

- Studies show rural low income areas more adversely affected during disasters
- Can this affect be observed by analyzing Twitter
- Compare/Contrast analysis techniques
    - 2 topic modeling techniques, 2 sentiment analysis
- Are some techniques better for this type of analysis

Main tasks:

- Collect clean and parse data
- Decide on classification methods
- Decide on analysis methods
- Analyze compare and contrast results

Data Collection and Cleaning

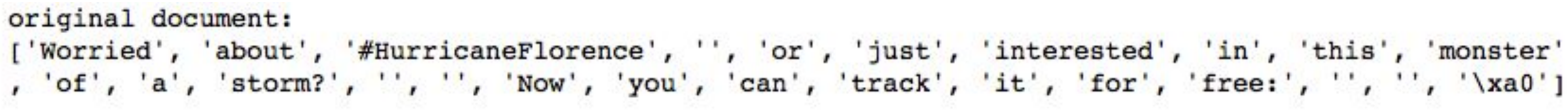
Data Collection

- Tweets containing **'\#hurricaneflorence'** obtained from twitter
- Three ways were available
  - AIDR (Artificial Intelligence for Digital Response)
    - can collect data with just one click
    - time consuming
    - have limitations in identifying keywords & download limitations
  - Python codes to get tweets directly using the Twitter API’s
    - restrictions because of the API's rate limits
    - can collect very few tweets (around 7000)
  - Python codes* built on Scrapy
    - can get tweets from Twitter Search without using Twitter API's.
    - can collect fairly large amount of tweets (around 200,000)
    - **used these codes to get tweets for our project**

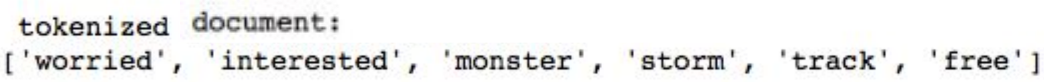
*codes source: github repository (https://github.com/DisasterMasters).

Data Cleaning - Python

- The gensim library's utils.simple_preprocess function and parsing.preprocessing.STOPWORDS list used to lowercase words and remove all links, hashtags, punctuation, stopwords. Words of length less than 3 also removed. The result was a cleaned and tokenized version of each tweet.



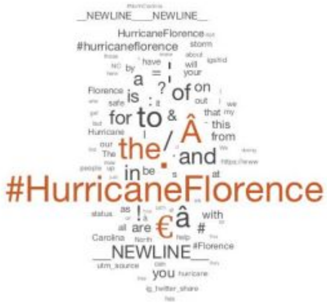
```
original document:
['Worried', 'about', '#HurricaneFlorence', '', 'or', 'just', 'interested', 'in', 'this', 'monster'
, 'of', 'a', 'storm?', '', '', 'Now', 'you', 'can', 'track', 'it', 'for', 'free:', '', '', '\xa0']
```



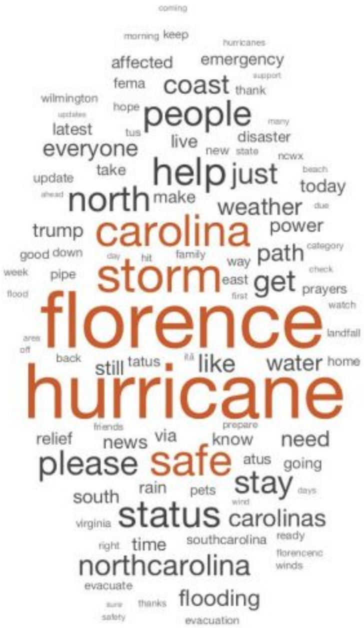
```
tokenized document:
['worried', 'interested', 'monster', 'storm', 'track', 'free']
```

Data Cleaning: Before and After

Raw Data

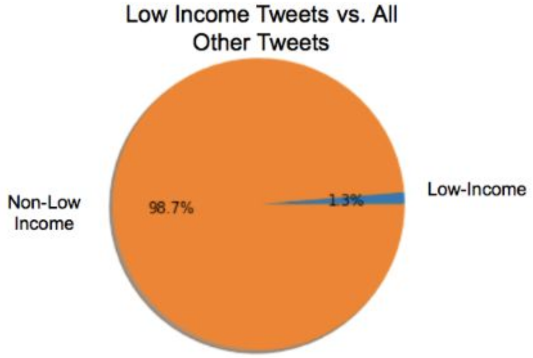


Clean Data

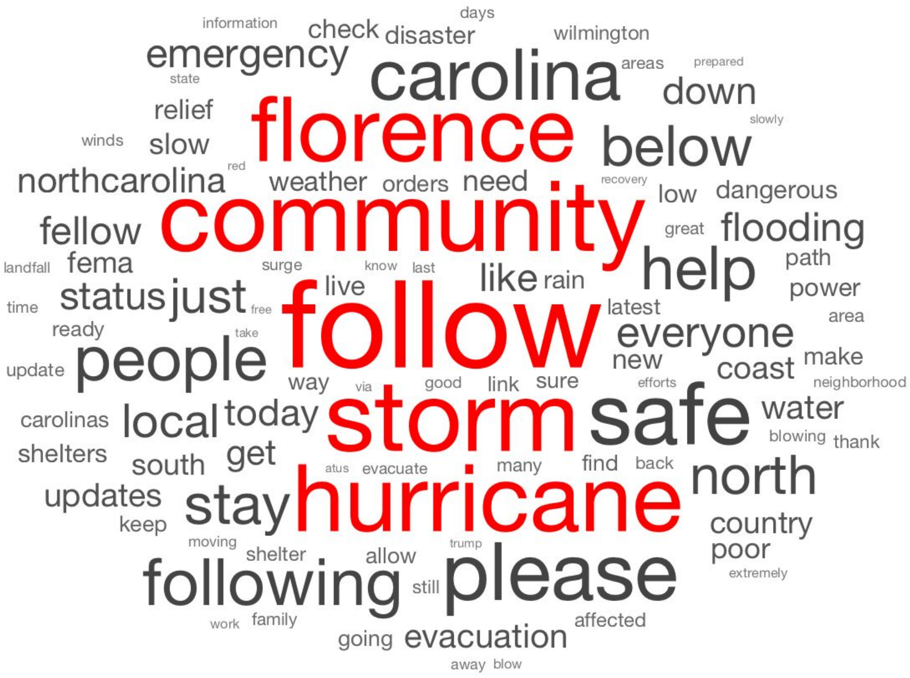


Low-income related tweets

- "In Poor, Rural Communities, Fleeing Hurricane Florence Was Tough".
- If the tweet contained one of the following words, it would be separated into the low-income list: rural, poor, community, communities, country, countryside, neighborhood, impoverished, poverty, broke, underprivileged, low, income, low-income.



Low-income related tweets



Clustering

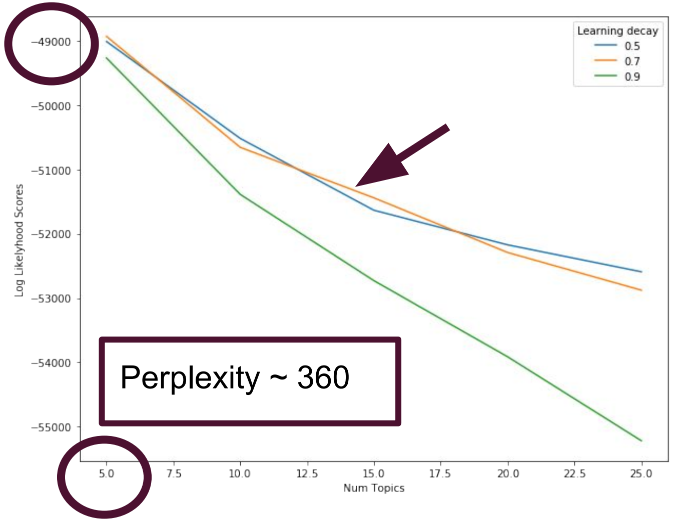
LDA Topic Modeling & K-Means

Latent Dirichlet Allocation (LDA)

Working Procedure:

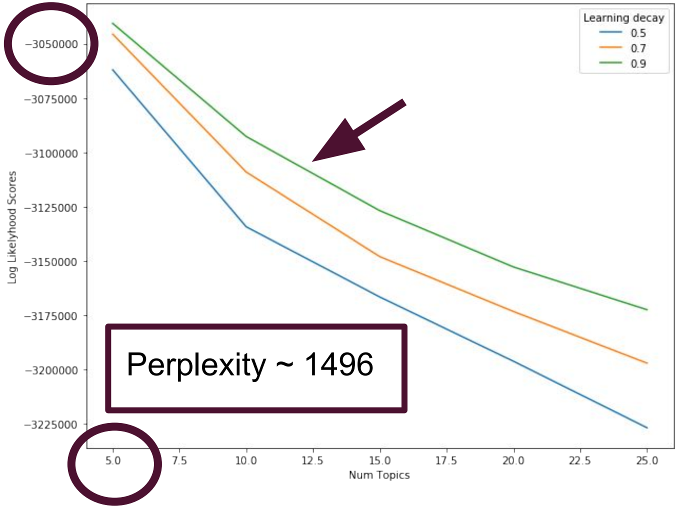
- **Used tools: core package: scikit-learn (sklearn)**
  - pyLDAvis and matplotlib for visualization
  - numpy and pandas for manipulating and viewing data in tabular format.
- **Main Input of LDA:**
  - Convert a collection of text documents to a matrix of token counts.
    - Word properties:
      - Frequency at least 10.
      - has to contain numbers and alphabets of at least length 3.
    - Working Procedure:
      - Convert words to lowercase.
      - Use CountVectorizer class to perform tokenization
      - Apply fit_transform create the matrix
- **Grid Search: Fix Number of Topics and Learning Decay:**
  - Run multiple LDA models on a range of number of topics and learning decays to achieve the best parameters

Number of Topics and Learning Decay



Perplexity ~ 360

Fig. : Choosing optimal LDA model for rural tweets

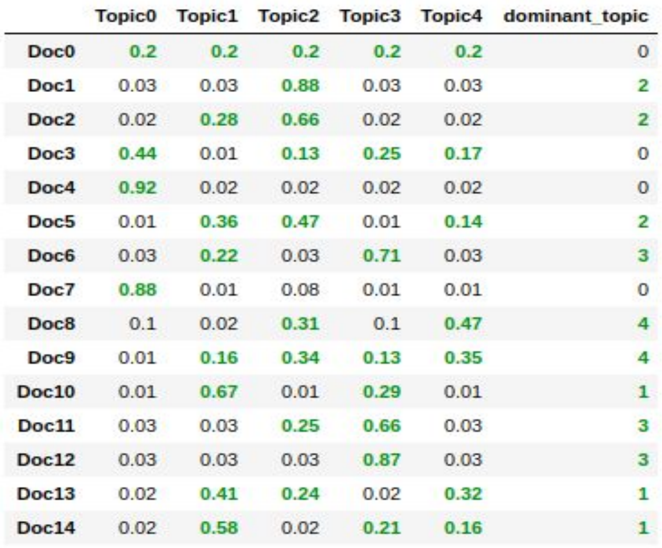


Perplexity ~ 1496

Fig. : Choosing optimal LDA model for nonrural tweets

A model with higher log-likelihood & lower perplexity indicates well performing models

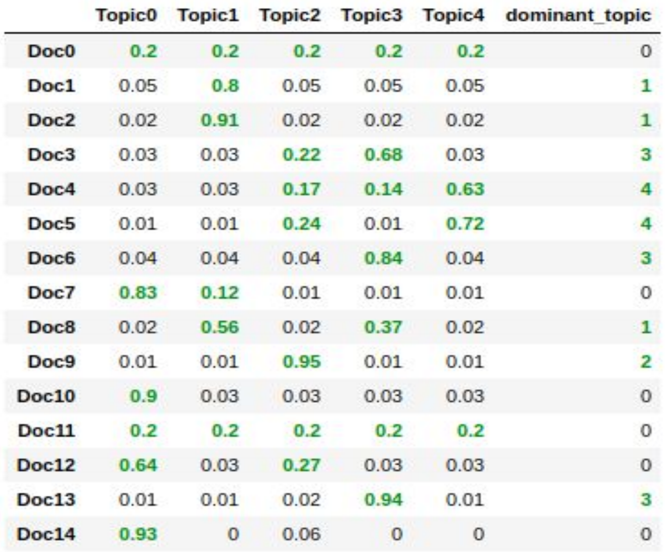
Apply LDA: Classify Tweets



	Topic0	Topic1	Topic2	Topic3	Topic4	dominant_topic
Doc0	0.2	0.2	0.2	0.2	0.2	0
Doc1	0.03	0.03	0.88	0.03	0.03	2
Doc2	0.02	0.28	0.66	0.02	0.02	2
Doc3	0.44	0.01	0.13	0.25	0.17	0
Doc4	0.92	0.02	0.02	0.02	0.02	0
Doc5	0.01	0.36	0.47	0.01	0.14	2
Doc6	0.03	0.22	0.03	0.71	0.03	3
Doc7	0.88	0.01	0.08	0.01	0.01	0
Doc8	0.1	0.02	0.31	0.1	0.47	4
Doc9	0.01	0.16	0.34	0.13	0.35	4
Doc10	0.01	0.67	0.01	0.29	0.01	1
Doc11	0.03	0.03	0.25	0.66	0.03	3
Doc12	0.03	0.03	0.03	0.87	0.03	3
Doc13	0.02	0.41	0.24	0.02	0.32	1
Doc14	0.02	0.58	0.02	0.21	0.16	1

Fig. : Dominant topics for top 15 tweets

LDA assigns each document to a **mixture** of topics.



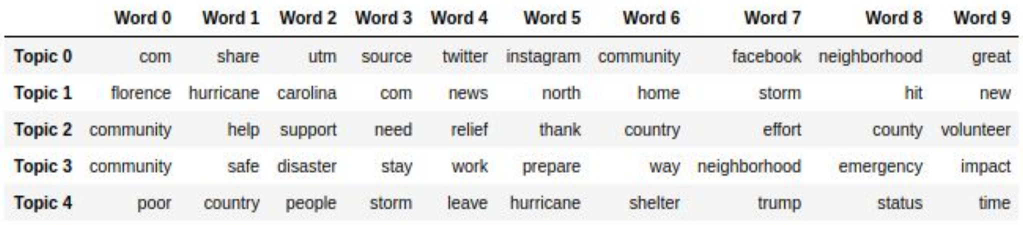
	Topic0	Topic1	Topic2	Topic3	Topic4	dominant_topic
Doc0	0.2	0.2	0.2	0.2	0.2	0
Doc1	0.05	0.8	0.05	0.05	0.05	1
Doc2	0.02	0.91	0.02	0.02	0.02	1
Doc3	0.03	0.03	0.22	0.68	0.03	3
Doc4	0.03	0.03	0.17	0.14	0.63	4
Doc5	0.01	0.01	0.24	0.01	0.72	4
Doc6	0.04	0.04	0.04	0.84	0.04	3
Doc7	0.83	0.12	0.01	0.01	0.01	0
Doc8	0.02	0.56	0.02	0.37	0.02	1
Doc9	0.01	0.01	0.95	0.01	0.01	2
Doc10	0.9	0.03	0.03	0.03	0.03	0
Doc11	0.2	0.2	0.2	0.2	0.2	0
Doc12	0.64	0.03	0.27	0.03	0.03	0
Doc13	0.01	0.01	0.02	0.94	0.01	3
Doc14	0.93	0	0.06	0	0	0

Fig. : Dominant topics for top 15 tweets

Approach:

- **Consider each tweet as a separate document**
- **See which topic has the highest contribution**
- **Assign it to the document**

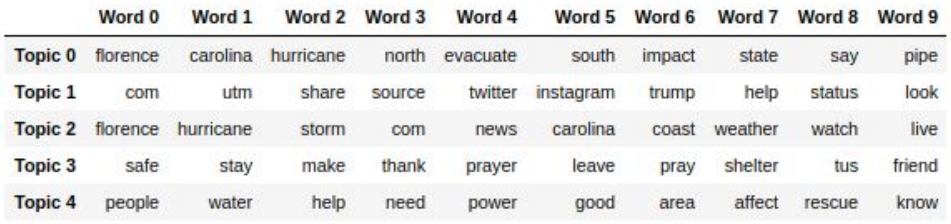
Topic Distribution of the Tweets



	Word 0	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7	Word 8	Word 9
Topic 0	com	share	utm	source	twitter	instagram	community	facebook	neighborhood	great
Topic 1	florence	hurricane	carolina	com	news	north	home	storm	hit	new
Topic 2	community	help	support	need	relief	thank	country	effort	county	volunteer
Topic 3	community	safe	disaster	stay	work	prepare	way	neighborhood	emergency	impact
Topic 4	poor	country	people	storm	leave	hurricane	shelter	trump	status	time

Fig. Top ten keywords present in each topic in the rural tweets

- Social media (15.4%)
- Hurricane & Media (20.1%)
- Help & Relief (23.4%)
- Safety Measure & Concern (21%)
- **Poverty (20%)**

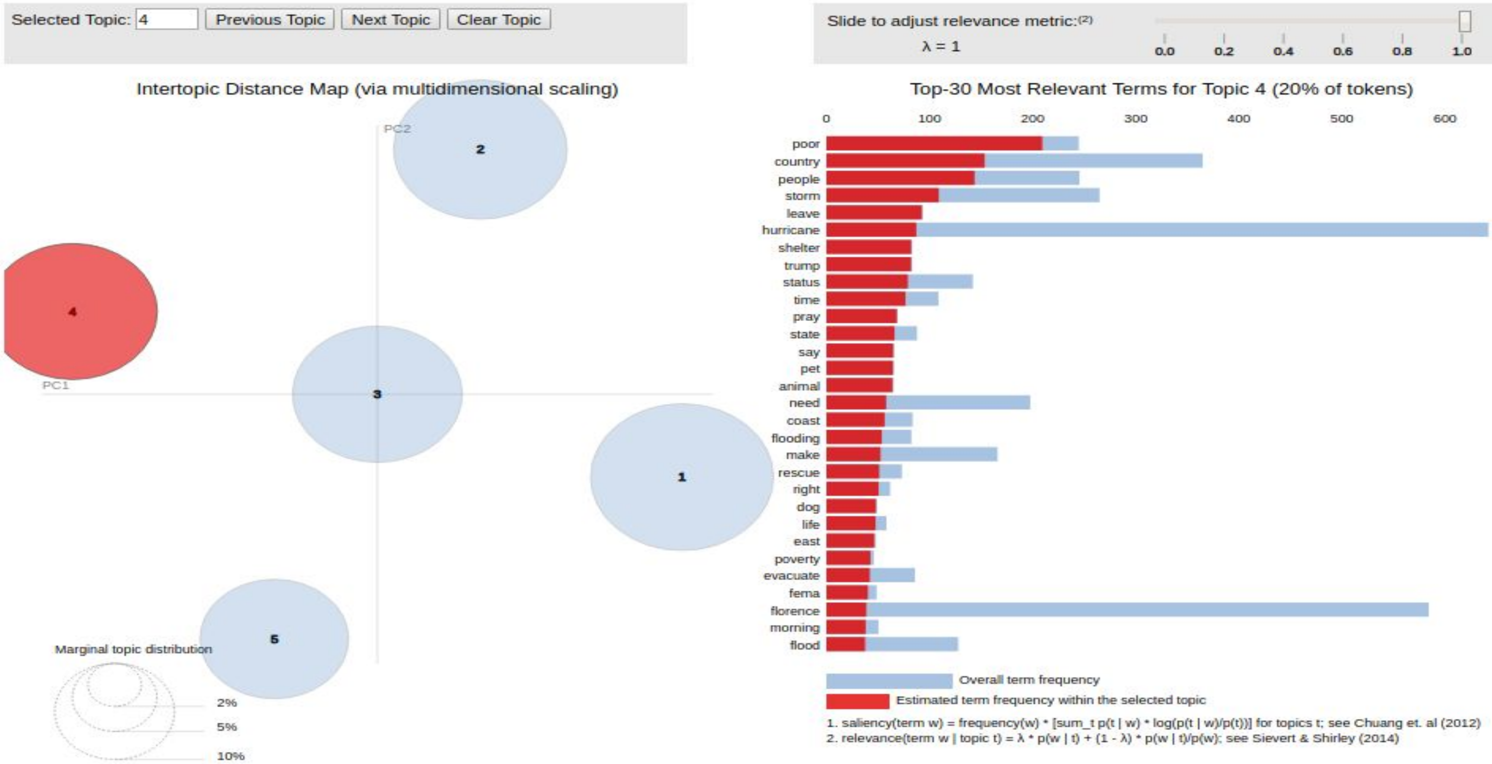


	Word 0	Word 1	Word 2	Word 3	Word 4	Word 5	Word 6	Word 7	Word 8	Word 9
Topic 0	florence	carolina	hurricane	north	evacuate	south	impact	state	say	pipe
Topic 1	com	utm	share	source	twitter	instagram	trump	help	status	look
Topic 2	florence	hurricane	storm	com	news	carolina	coast	weather	watch	live
Topic 3	safe	stay	make	thank	prayer	leave	pray	shelter	tus	friend
Topic 4	people	water	help	need	power	good	area	affect	rescue	know

Fig. Top ten keywords present in each topic in the nonrural tweets

- **Disaster & Impact (18%)**
- Social media (19%)
- Hurricane & Media (25.9%)
- Safety Measure & Concern (17.1%)
- Help & Relief (20.1%)

Visualization of LDA models



Selected Topic: 4

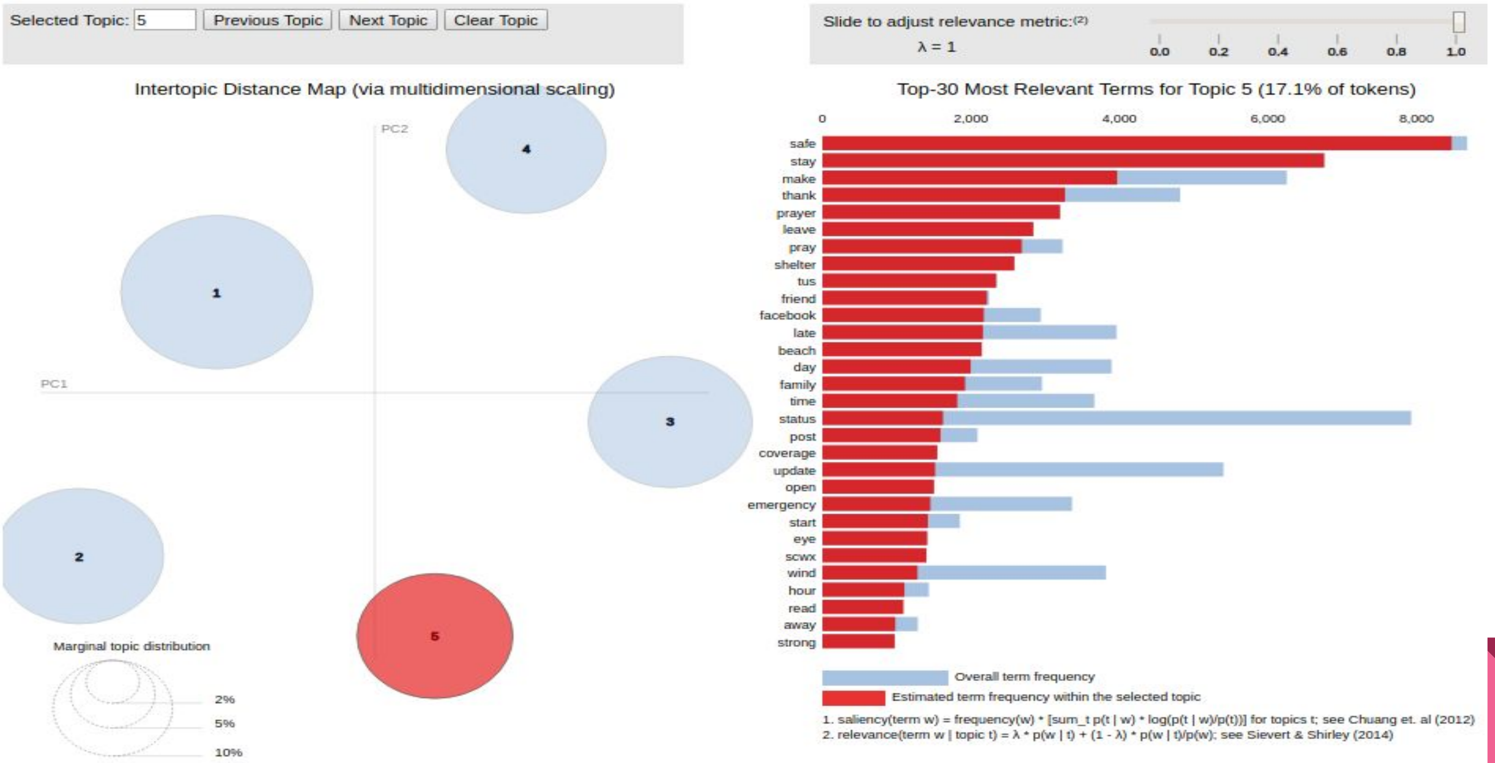
Slide to adjust relevance metric:(2)

λ = 1

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Fig. : LDA Topic Model for Rural Tweets

Visualization of LDA models



🟦 Overall term frequency

🟥 Estimated term frequency within the selected topic

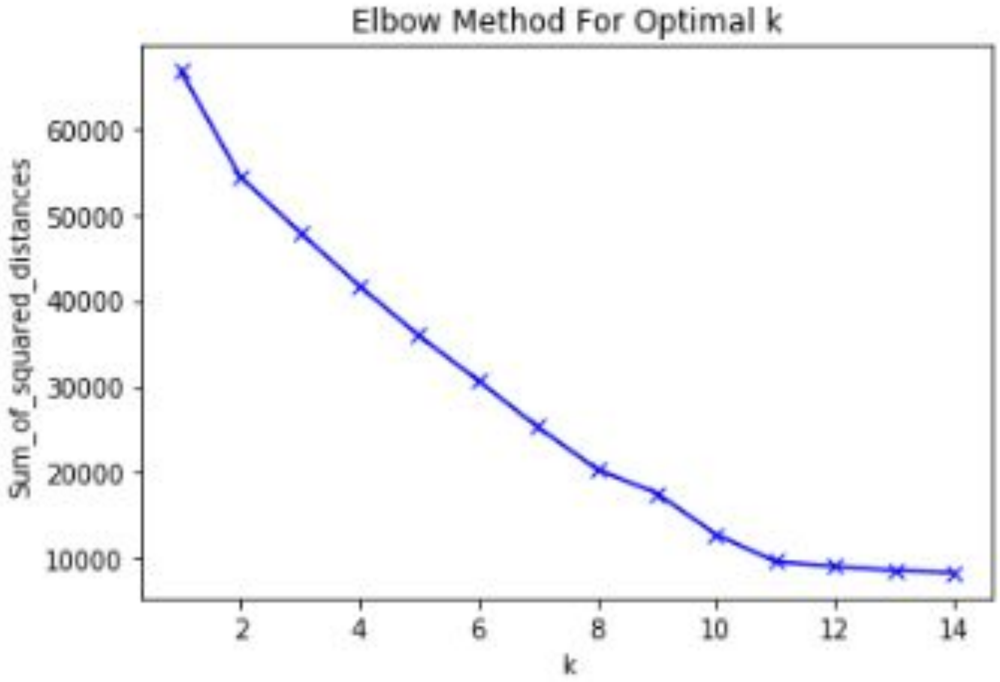
1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

Fig. : LDA Topic Model for Non-Rural Tweets

K-Means Clustering

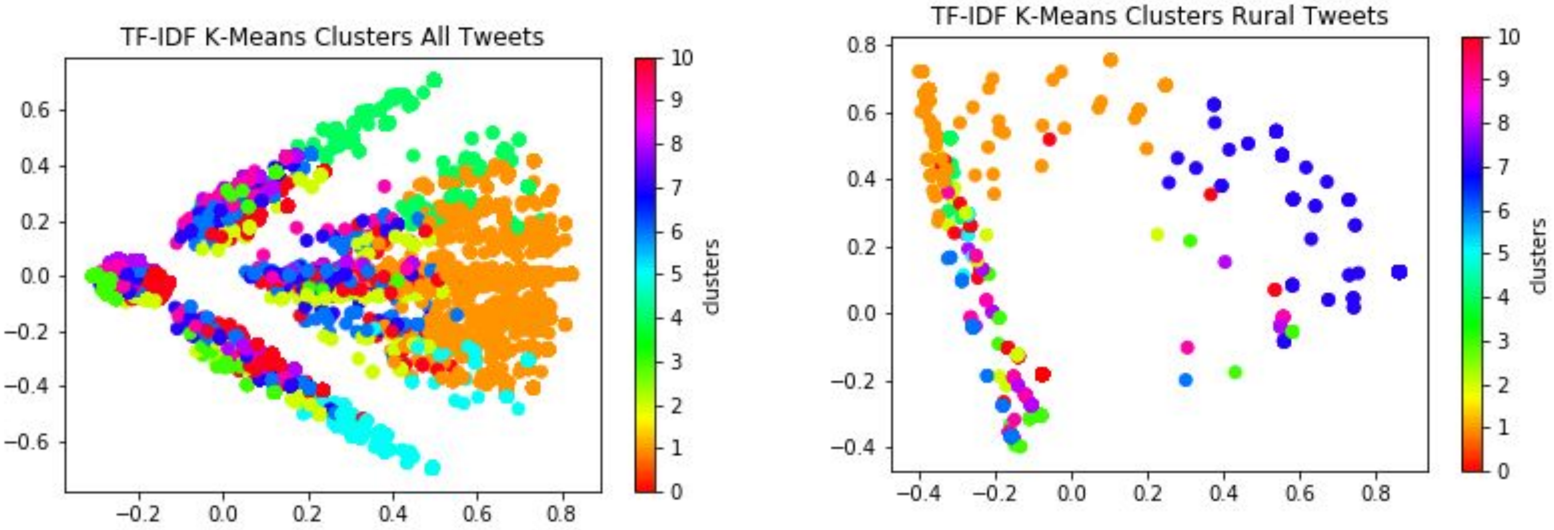
K-Means Clustering Procedure

- K-Means clustering was used to cluster similar tweets.
- The text of each tweet was vectorized using Scikit-Learn's **TfidfVectorizer** function and the corpus was stored in a matrix of TF-IDF values.
- Vectorized tweets were clustered using different values for k. An optimal k-value of 11 was chosen by calculating the sum of squared distances for each clustering, graphing these values and applying the elbow method.



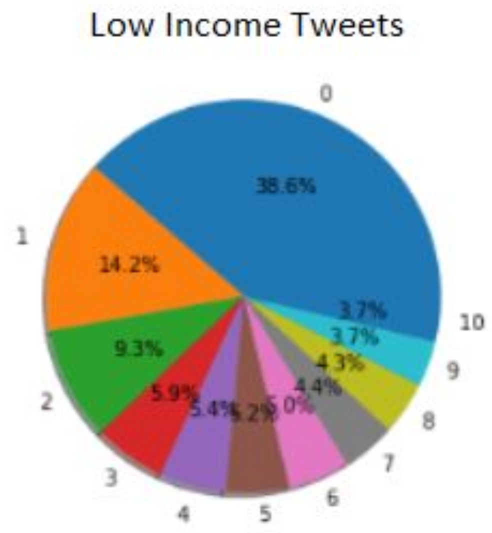
An optimal k of 11 was chosen using the elbow method.

K-Means Scatterplots

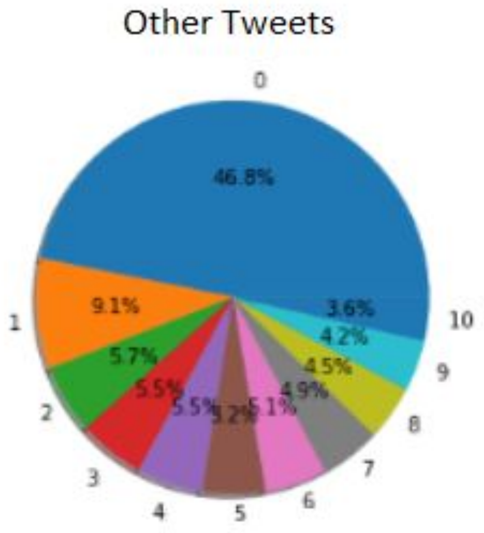


Scatterplots of first two principal components of TF-IDF vectors of all tweets and low-income/rural subset.

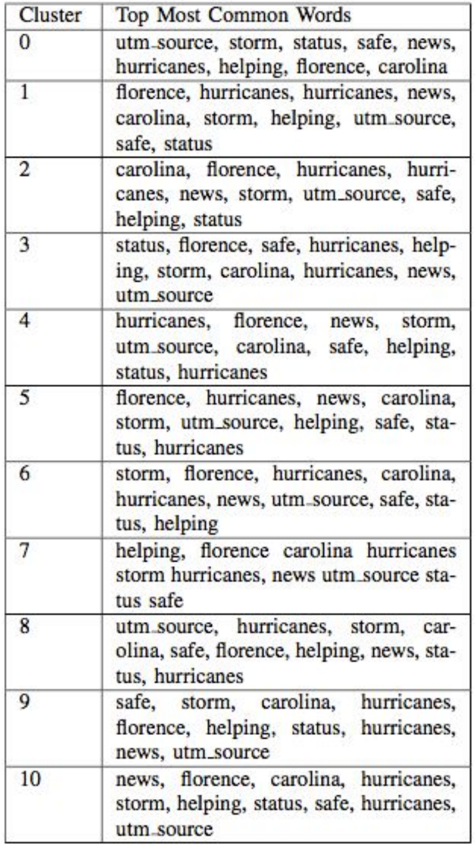
Comparing K-Means Terms by Socioeconomic Status



Low Income Tweets



Other Tweets



Cluster	Top Most Common Words
0	utm_source, storm, status, safe, news, hurricanes, helping, florence, carolina
1	florence, hurricanes, hurricanes, news, carolina, storm, helping, utm_source, safe, status
2	carolina, florence, hurricanes, hurricanes, news, storm, utm_source, safe, helping, status
3	status, florence, safe, hurricanes, helping, storm, carolina, hurricanes, news, utm_source
4	hurricanes, florence, news, storm, utm_source, carolina, safe, helping, status, hurricanes
5	florence, hurricanes, news, carolina, storm, utm_source, helping, safe, status, hurricanes
6	storm, florence, hurricanes, carolina, hurricanes, news, utm_source, safe, status, helping
7	helping, florence carolina hurricanes storm hurricanes, news utm_source status safe
8	utm_source, hurricanes, storm, carolina, safe, florence, helping, news, status, hurricanes
9	safe, storm, carolina, hurricanes, florence, helping, status, hurricanes, news, utm_source
10	news, florence, carolina, hurricanes, storm, helping, status, safe, hurricanes, utm_source

K-Means Clustering: Key Findings

- Each cluster seemed to have relatively similar values for its most common words.
- The key difference between the clusters is each word’s rank in popularity
- The rural/low-income subset of tweets had a smaller proportion of tweets falling into cluster 0 and a larger proportion of tweets representing clusters 1 and 2.

What are the differences between these two?

- **Both K-means and Latent Dirichlet Allocation (LDA) are unsupervised learning algorithms**
    - Needs to decide a priori the parameter K (K-Means: number of clusters, LDA: number of topics)
- **Applied both to assign K topics to a set of tweets (N documents)**
    - **Noticed:** optimum number of topic distribution
        - LDA: 5
        - K-Means: 11
    - **Probable Reason:**
        - K-means partitioned the documents set in K **disjoint** topics
        - LDA assigned each document to a **mixture** of topics. Hence, each topic is characterized by one or more topics.

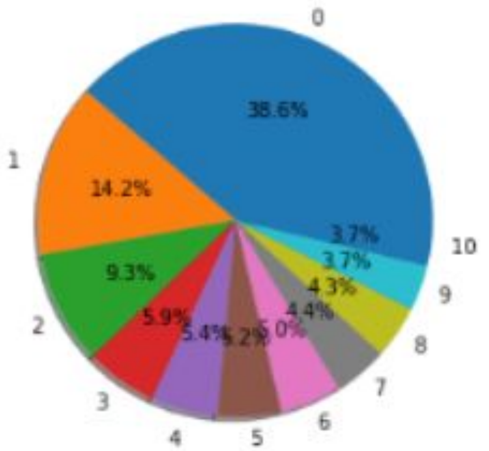
Benefits and Drawbacks of LDA and K-Means

- **LDA:**
  - Pros: Produces list of topics in corpus alongside list of specific keywords representing each topic.
  - Cons: Method not designed to assign topics to individual tweets, takes hours to Grid Search
- **K-Means:**
  - Pros: Assigns each tweet to a specific cluster based on similarity.
  - Cons: Does not directly yield list of topics defining each cluster.

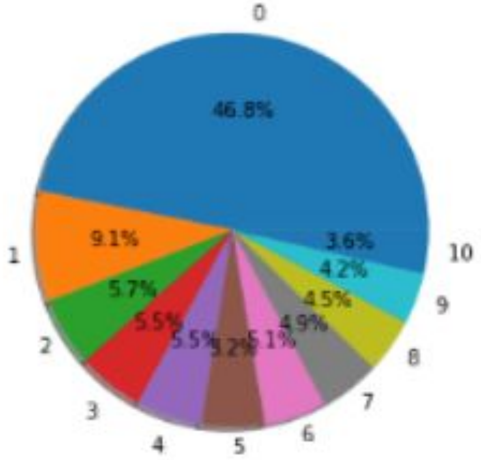
Applying LDA to K-Means Clusters

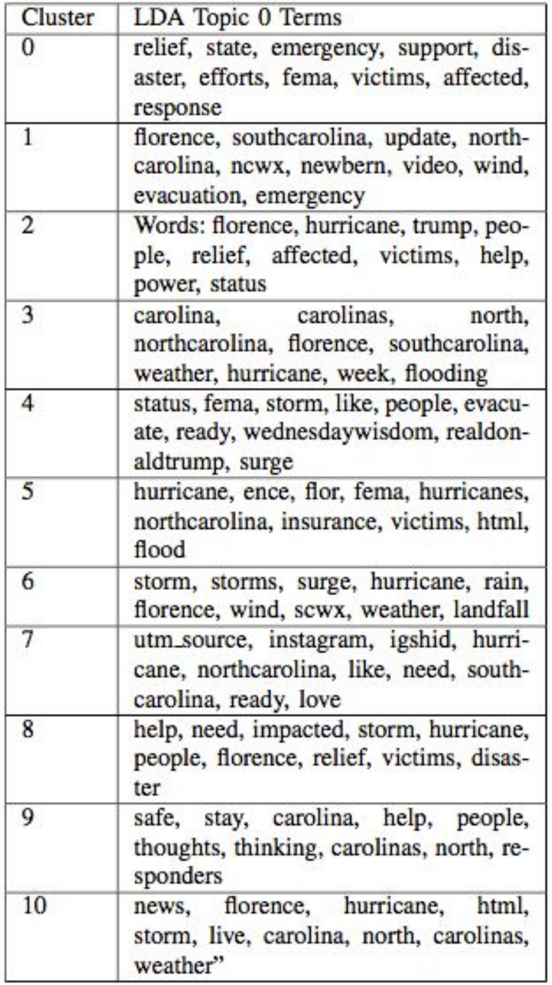
- LDA applied to K-Means Clusters to obtain a list of key topics for each.
  - Expand this application in future work.

Low Income Tweets



Other Tweets





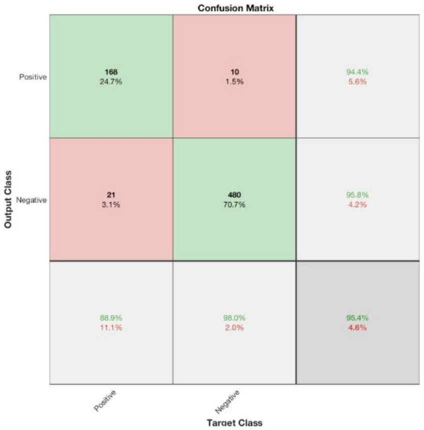
Cluster	LDA Topic 0 Terms
0	relief, state, emergency, support, disaster, efforts, fema, victims, affected, response
1	florence, southcarolina, update, northcarolina, ncwx, newbern, video, wind, evacuation, emergency
2	Words: florence, hurricane, trump, people, relief, affected, victims, help, power, status
3	carolina, carolinas, north, northcarolina, florence, southcarolina, weather, hurricane, week, flooding
4	status, fema, storm, like, people, evacuate, ready, wednesdaywisdom, realdonaldtrump, surge
5	hurricane, ence, flor, fema, hurricanes, northcarolina, insurance, victims, html, flood
6	storm, storms, surge, hurricane, rain, florence, wind, scwx, weather, landfall
7	utm_source, instagram, igshid, hurricane, northcarolina, like, need, southcarolina, ready, love
8	help, need, impacted, storm, hurricane, people, florence, relief, victims, disaster
9	safe, stay, carolina, help, people, thoughts, thinking, carolinas, north, responders
10	news, florence, hurricane, html, storm, live, carolina, north, carolinas, weather”

Sentiment Analysis

Support Vector Machine and Python TextBlob

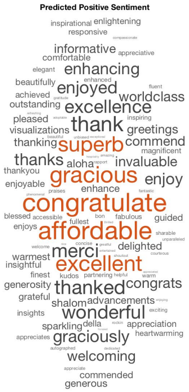
Support Vector Machine (SVM) in MATLAB

- A sentiment word list compiled by the University of Illinois Chicago was used to train the SVM to predict the sentiments of tweets used in this study (6,790 words that were categorically labeled either "positive" or "negative")
- SVM ran on the Florence twitter dataset and output a sentiment score for the individual words in the tweet >0 being positive and <0 being negative

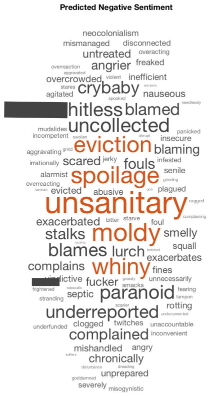


Confusion Matrix

Output Class \ Target Class	Positive	Negative	
Positive	168 24.7%	10 1.5%	94.4% 5.6%
Negative	21 3.1%	480 70.7%	95.8% 4.2%
	88.9% 11.1%	98.0% 2.0%	95.4% 4.6%



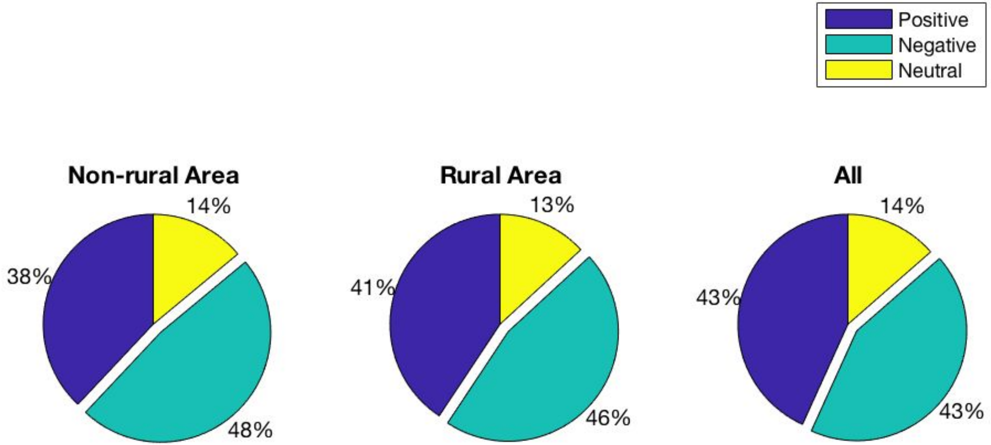
Predicted Positive Sentiment



Predicted Negative Sentiment

SVM in MATLAB

- There were several instances of words that were not recognized by the built-in word embedding, these words were removed
- Tweets given a score between -0.3 and 0.3 were counted as neutral
- Overall, the non-rural area related tweets had the highest percent of negative tweets, but not by a significant amount.

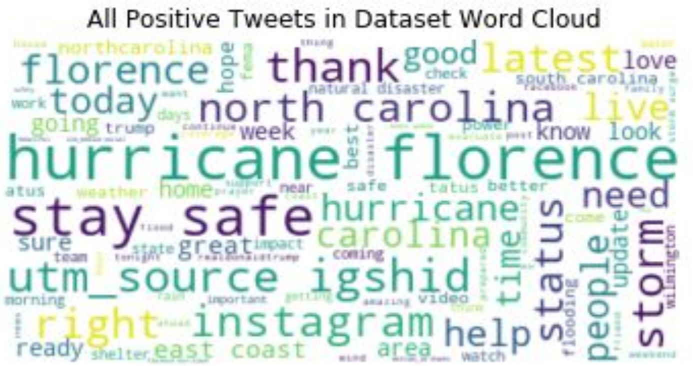


Issues with SVM

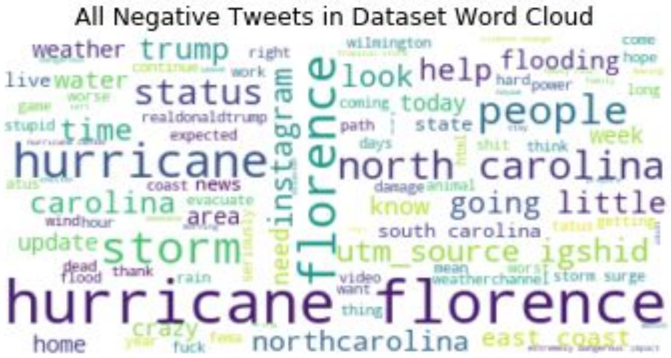
- No categorically labeled “neutral” sentiments
- Average sentiment of all words in a tweet was used for the sentiment of the whole tweet
- Many words in the tweets were not recognized by the built-in word embeddings, so they were not given a sentiment score

TextBlob in Python

- Sentiment analysis was performed on text of each tweet using Python's TextBlob library for NLP.
- All tweets passed to the TextBlob class's sentiment function, which yielded a polarity value between -1.0 and 1.0, denoting a sentiment classification of either negative (sentiment < 0), positive (sentiment > 0), or neutral (sentiment = 0).
- Word clouds were constructed from the words of tweets identified as either positive, negative or neutral.



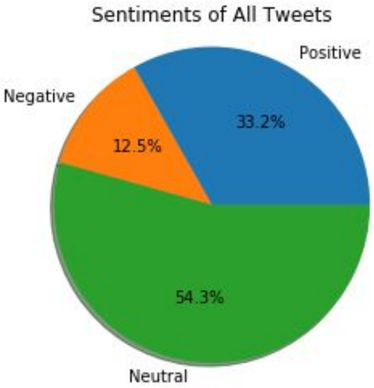
All Positive Tweets in Dataset Word Cloud

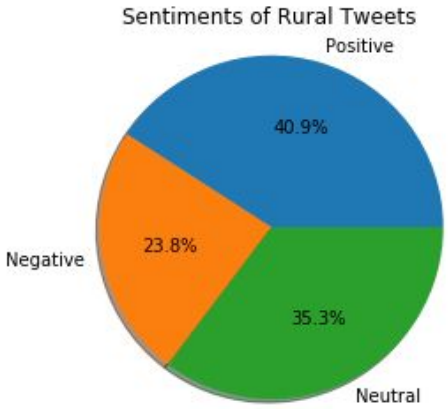


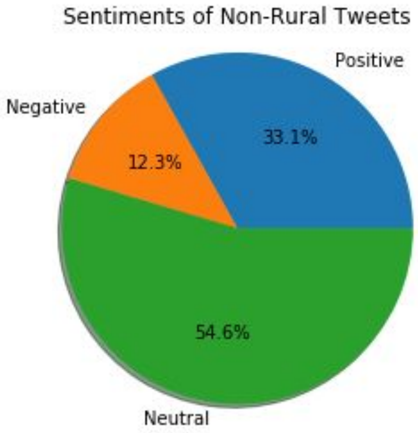
All Negative Tweets in Dataset Word Cloud

Comparing Sentiments by Socioeconomic Status

- 46,225 tweets classified as positive, 17,357 as negative, and 75,625 as neutral.
- The sentiment percentage breakdowns of tweets in the entire dataset, the subset of tweets containing rural/low-income related terms are shown below.
- The percentage of neutral tweets lower in the rural/low-income subset than in the set of all other tweets.
- The proportions of tweets classified as negative and positive were higher in this subset in comparison to that of all other tweets. The largest percentage of tweets in this subset had a classification of positive.


Sentiments of All Tweets


Sentiments of Rural Tweets


Sentiments of Non-Rural Tweets

Conclusions

Pros and Cons: LDA vs. K Means

LDA

Pros :

- Concise model
- Easily understandable

Cons:

- Time consuming

K-Means

Pros:

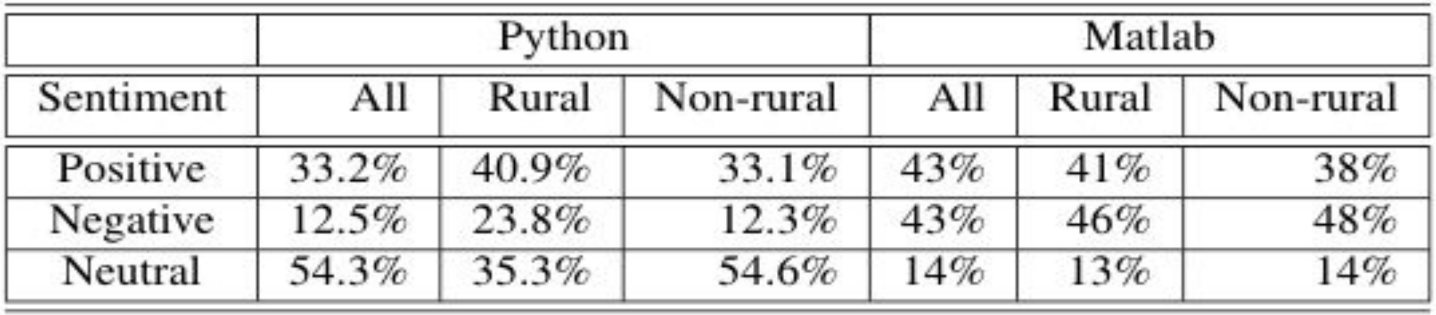
- Better at mass clustering

Cons:

- Harder to interpret

Comparison of SVM and TextBlob

- Rural tweets more positive sentiment than non rural
- Differences in Python TextBlob vs. Matlab SVM



	Python			Matlab		
Sentiment	All	Rural	Non-rural	All	Rural	Non-rural
Positive	33.2%	40.9%	33.1%	43%	41%	38%
Negative	12.5%	23.8%	12.3%	43%	46%	48%
Neutral	54.3%	35.3%	54.6%	14%	13%	14%

Future Work

- Way of scraping tweets with geographic data
- Using geographic data to improve parsing of data
- Further analysis of which methods work best for this type of project

Team Member Roles

Maria Mahbub: Data collection, LDA

Linsey S. Passarella: Low-income tweet database, MATLAB Sentiment Analysis/SVM

Emily J. Herron: K-Means, Python Sentiment Analysis

Gerald L. Jones: Research, Comparison analysis, Goal Manager

Questions?