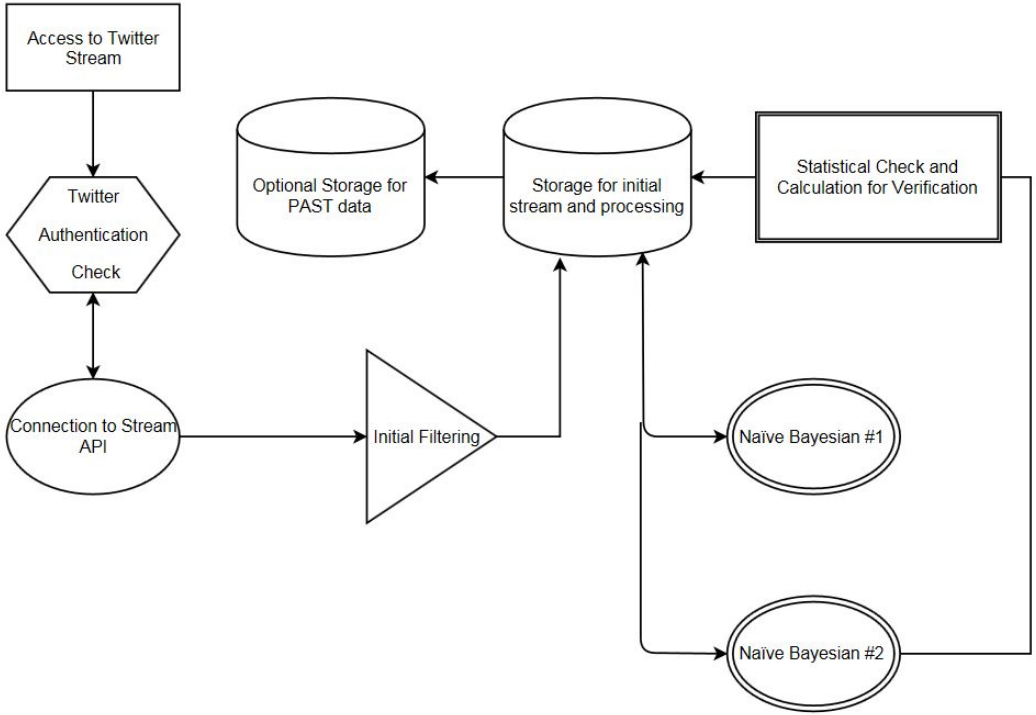
Detecting informative Tweets from Twitter Feed During Natural Calamities

Akash Malla
Samarth Mehta

Overview

- Implementation
- Data Analysis
- Conclusion
- Recommendation/Future Work

Implementation Flowchart



Implementation

Step 1 : Created Twitter Application for Authenticated use of Twitter Feed

- Owner : samarth128, Owner ID : 28956455

Step 2 : Authentication and Connection to API

- Consumer + Authentication Key & tokens passed and checked at runtime

Step 3 : Initial Filtering

- Retweets : Checking for String “RT” present in the recieved tweet, after making sure that original tweet exist
- Difflib Library : Computing Delta to check if “similar” tweet already exists
- Similarity threshold : 0.8

Implementation

Step 5 : Parsing Twitter JSON

Sample :


```
"created_at": "Fri Jan 23 23:57:36 +0000 2015",
"id": 558775589612437504,
"id_str": "558775589612437504",
"text": "Thanks, polar vortex: Attendance dips at major Chicago museums in 2014 http:\/\/t.co\/jQlLEurk9s http:\/\/t.co\/3bss2nGemx",
"source": "\u003ca href=\"http:\/\/twitter.com\" rel=\"nofollow\"\u003eTwitter Web Client\u003c\/a\u003e",
"truncated": false,
"in_reply_to_status_id": null,
"in_reply_to_status_id_str": null,
"in_reply_to_user_id": null,
"in_reply_to_user_id_str": null,
"in_reply_to_screen_name": null,
"user": {
  "id": 7313362,
  "id_str": "7313362",
  "name": "Chicago Tribune",
  "screen_name": "chicagotribune",
  "location": "Chicago, IL",
  "url": "http:\/\/www.chicagotribune.com\/",
  "description": "Chicago Tribune news, features and so much more live from our newsroom. A part of your life since 1847.",
  "protected": false,
  "verified": true,
  "followers_count": 321548,
  "friends_count": 523,
  "listed_count": 8074,
  "favourites_count": 34,
  "statuses_count": 47367,
  "created_at": "Sat Jul 07 14:10:07 +0000 2007",
  "utc_offset": -21600,
  "time_zone": "Central Time (US & Canada)",
  "geo_enabled": false,
  "lang": "en",
```

Implementation

Step 5 (Continued) : Storing Retrieval of Data (.csv)

- Fields of Interest : Text, Tweet ID, Created at, Geo Enabled, Hashtag, URL

Step 6 : Naive Bayesian run#1 and run#2

- Used sklearn module's term frequency and inverse document frequency API to compute importance of every word in all tweets.
- First run predicted tweets to be classified as informative or noninformative.
- Second run predicted tweets to be donation/help related or caution/advice related

Step 7 : Statistical Calculation

Step 8 : Output Screen

Data Analysis : Accuracy Output with single Dataset as training

- Alberta floods, 2013:

Related vs **Non Related** : 73%
Train Data : 983
Test Data : 658

Donations and Volunteering vs **Caution and Advice** : 77%
Train Data : 217
Test Data : 107

```
Run:  test   test   streaming

/Library/Frameworks/Python.framework/Versions/3.6/bin/python3.6 /Users/akashmalla/PycharmProjects/TwitterEventDetection/streaming.py
Split 983 tweets into training set with 658 tweets and testing set with 325 tweets
['Related - but not informative', 'Related and informative']
Bayes accuracy:  0.735384615385
Classification Report:
                             precision    recall  f1-score   support

Related - but not informative       1.00      0.03      0.07        89
      Related and informative       0.73      1.00      0.85       236

                  avg / total       0.81      0.74      0.63       325

Confusion Matrix:
[[  3  86]
 [  0 236]]

Below we predict caution and help related tweets:
Train data set:  217 Test data set:  107
Bayes accuracy:  0.775700934579
Classification Report:
                            precision    recall  f1-score   support

Donations and volunteering       1.00      0.23      0.37        31
        Caution and advice       0.76      1.00      0.86        76

               avg / total       0.83      0.78      0.72       107

Confusion Matrix:
[[ 7 24]
 [ 0 76]]

Process finished with exit code 0
```

Data Analysis : “Learning” from past Calamities

Taking multiple datasets as training

- Alberta, 2013 + Colorado, 2013 + Philippines, 2012 + Sardinia, 2013 + Queensland, 2013

Related vs **Non Related** : 79%

Train Data : 5580

Test Data : 3738

Donations and Volunteering vs **Caution and Advice** : 87%

Train Data : 1353

Test Data : 667

```
Run:  test  test  streaming
/Library/Frameworks/Python.framework/Versions/3.6/bin/python3.6 /Users/akashmalla/PycharmProjects/TwitterEventDetection/streaming.py
Split 5580 tweets into training set with 3738 tweets and testing set with 1842 tweets
['Related - but not informative', 'Related and informative']
Bayes accuracy:  0.790445168295
Classification Report:
                             precision    recall  f1-score   support

Related - but not informative       0.92      0.18      0.31       464
      Related and informative       0.78      0.99      0.88      1378

                  avg / total       0.82      0.79      0.73      1842

Confusion Matrix:
[[  85  379]
 [   7 1371]]

Below we predict caution and help related tweets:
Train data set:  1353 Test data set:  667
Bayes accuracy:  0.869565217391
Classification Report:
                           precision    recall  f1-score   support

        Caution and advice       0.94      0.76      0.84       305
Donations and volunteering       0.83      0.96      0.89       362

               avg / total       0.88      0.87      0.87       667

Confusion Matrix:
[[233  72]
 [ 15 347]]

Process finished with exit code 0
```

Output Comparison by number of Categories

- 2 Categories

	precision	recall	f1-score	support
Caution and advice	0.94	0.76	0.84	305
Donations and volunteering	0.83	0.96	0.89	362
avg / total	0.88	0.87	0.87	667

- 4 Categories

	precision	recall	f1-score	support
Donations and volunteering	0.84	0.66	0.74	250
Infrastructure and utilities	0.70	0.71	0.70	287
Affected individuals	0.67	0.91	0.77	389
Caution and advice	0.74	0.41	0.53	210
avg / total	0.73	0.71	0.70	1136

San Jose Flood Tweets Statistics

Tweets Observed over search API:

Limit : 100
Query Results containing “San Jose Floods” : 68
After 1st Filtering (Duplicates) : 36

Category Results:

- Donation - 15, Caution - 6, Other - 13, Sympathy - 2

Tweet: Don't stop finger-pointing: "The Coyote Creek flood is too significant a failure to worry about hurt feelings." https://t.co/tDXkfVGUHI . Prediction: Sympathy and support

Tweet: Come out and donate this weekend at Olinder Elementary!! Donation Drive! Coyote Creek Flood Relief Fundraising... https://t.co/kRN4pCN7FG . Prediction: Donations and volunteering

Tweet: Public Hearing Held In San Jose To Discuss Response To Coyote Creek Flooding https://t.co/A3RVB26Hgh #sanfrancisco . Prediction: Other Useful Information

Tweet: VIDEO: San Jose declares shelter crisis amid devastating Coyote Creek flood https://t.co/NevAb8Z0Up via @kron4news . Prediction: Caution and advice

Conclusions

- Positives:
  - Naive Bayes is very fast for text classification
  - With a large enough dataset, accuracy can be as high as,
    - Informative - 79%
    - Caution vs Help - 86%
  - Upon wise selection of Categories for classification results precision is as high as
    - Donations - 94%
    - Caution - 83%

Conclusion

- Challenges:
  - 7-10 Days limit:
    - Twitter doesn't let you extract topic-based tweets after this duration
    - Storing data older than 7 days becomes crucial
  - Continuous Data Storage :
    - Bigger Storage and Processing requirements to use it as a Live Application
  - Intended Spam to go past filtering:
    - Eg. Victoria Beckham's winter protection styling choice during New York Snow Storm
    - Use of Mirror Links so duplicate checker fails
  - Twitter Error 420 & 429 : Overuse of Set Limit of Twitter Resources
    - Eg. Trying to fetch thousands of 'past' tweets continuously breaks connection from Twitter's side
  - Tweepy API :
    - Often timeouts and needs to be reset when the stream is longer than a few hours

Recommendation/Future Work

- Parallel Processing:
  - Can reduce overhead of storing to disk and reading it back for filtering
  - 2 processes in parallel:
    - Handler for incoming stream (Initial Filter + Storage)
    - Hierarchical Classification to categorize
- Geo-based additional filtering to extract tweets from hotspots only:
  - Using GeoNames origin of the source can be found and filtered which let's us focus more on local tweets as they might have more recent and eyewitness news.
- Web-based application that outputs the result of Classification
  - Emergency Forces
  - Authenticated Stream of Live Events

Bibliography

1. M. Rajdev and K. Lee, "Fake and Spam Messages: Detecting Misinformation During Natural Disasters on Social Media," *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, Singapore, 2015, pp. 17-20.
2. Himanshu Shekhar, Shankar Setty, Uma Mudenagudi, "Vehicular traffic analysis from social media data", *Advances in Computing Communications and Informatics (ICACCI) 2016 International Conference on*, pp. 1628-1634, 2016.
3. Vishal Kumar, Kunwar Singh Vaisla, Jaydeep Kishore, "Analyzing Email Account Creation: Expectations vs Reality", *Communication Systems and Network Technologies (CSNT) 2014 Fourth International Conference on*, pp. 597-600, 2014.
4. Muhammad Imran, Shady Elbassuoni, Carlos, Fernando Diaz, Patrick Meiser, “Extracting Information Nuggets from Disaster-Related Messages in Social Media”, *ICIS - International Conference on Information Systems, 2013*

Thank You…