Detection and Analysis of Disaster-Related Tweets

Daniel Solomon
solomond@mail.tau.ac.il

Gal Ron
galr1@mail.tau.ac.il

Omri Ben-Horin
EMAIL@mail.tau.ac.il

Abstract

TODO: bla.

1 Introduction

The popular microblogging service Twitter is a fruitful source of user-created content. With hundreds of millions of new tweets every day, Twitter has become a probe to human behavior and opinions from around the globe. The Twitter 'corpus' reflects political and social trends, popular culture, global and local happenings, and more. In addition, tweets are easy to access and aggregate in real-time. Therefore, we experience an increased interest in natural language processing research of Twitter data.

As one of the world's most widely used social networks, Twitter is an effective channel of communication and plays an important role during a crisis or emergency. The live stream of tweets can be used to identify reports and calls for help in emergency situations, such as accidents, violent crimes, natural disasters and terror attacks (which we all refer to as 'disasters' in this paper).

In this work we utilize techniques from the natural language processing pipeline (tokenization, part-of-speech tagging and named-entity recognition) to work on Twitter data, as opposed to traditional corpora, in order to detect and analyze disaster-related tweets.

The Dataset We present our experiments on a dataset of 10,877 tweets[1], labeled to *'disaster-related'* and *'not disaster-related'* with confidence in the range $[0, 1]$. For example, the following tweet is *'disaster-related'* with confidence 1,

> Thunderstorms with little rain expected in Central California. High fire danger. #weather #cawx http://t.co/A5GNzbuSqq

while the following tweet is *'not disaster-related'* with confidence 0.59,

> It's been raining since you left me // Now I'm drowning in the flood // You see I've always been a fighter // But without you I give up

Even for one who is not familiar with the Bon Jovi lyrics in the latter tweet, it is clear that the tweet does not refer to a real natural disaster. However, this observation is hard to make examining only the vocabulary used; the latter tweet contains a variety of 'disastrous' words (e.g. raining, drowning, flood, fighter). This example hints that in order to reach meaningful results we may have to examine additional linguistic features of colloquial writing, as well as Twitter-specific features such as hashtags (**#**), user at-mentions (**@**), internet links and emoticons.

Our Contribution In this paper we present our work tackling three missions involving disaster-related tweets.

The first mission is *identification* of disaster-related tweets among a variety of tweets. We implemented several classifiers, the best of which achieved TODO: % accuracy on the dataset. We note that this method could have easily been adjusted to identify tweets related to themes other than disasters (e.g. politics-related, sports-related, etc.), given the appropriate dataset.

The second mission is binary classification of disaster-related tweets to one of two categories, *subjective tweets* (i.e. tweets that express an emotion) vs. *objective tweets* (such as news reports on disasters). To achieve this we manually tagged 2,410 disaster-related tweets. The motivation behind this task is that objective tweets like informative news reports are likely to be published after the event had already become clear to emergency services, while subjective tweets may contain invaluable first-person testimonies of ongoing events.

Finally, we extracted named entities to enrich our knowledge on the disaster (mostly location)... TODO: Omri - short description of method and achievements.

To demonstrate our framework we aggregated recent tweets from various locations in the US, extracted disaster-related tweets using our classifier, and then used named-entity recognition to discover entities related to ongoing disasters. For example, "Hurricane Harvey" appeared as a top named-entity among recent tweets sent from Houston, TX, which we identified as *disaster-related*.

The code of our project is available at `https://github.com/glrn/nlp-disaster-analysis`.

1.1 Twitter vs. Traditional Corpora

Tweets are limited to 140 characters and are widely used by non-professional writers. Therefore, Tweet datasets have some unique features that differ from traditional corpora (such as WSJ corpus). These features should be addressed when implementing natural language processing techniques.

First, the grammar of tweets is quite different from edited news text. It is common that tweets are written as colloquial sentences in first person where the subject ('I') is omitted, as in: 'see the flames from my window OMG'.

Tweets are also characterized by extensive use of abbreviations and slang specific to social-media (e.g. ily for 'I love you', iono for 'I don't know). Such abbreviations may squash several parts-of-speech into one token, which poses a challenge to POS tagging.

In addition, due to the colloquial nature of user-created content, it is common that proper words are replaced by phonetically or morphologically similar ones (e.g. 'wtchng' instead of 'watching', 'gr8' instead of 'great'). Users may also use capitalization irregularities, deliberate spelling errors, punctuation irregularities and interjections as a means to express their sentiment, as in the following tweet:

> Haha South Tampa is getting flooded hah- WAIT A SECOND I LIVE IN SOUTH TAMPA WHAT AM I GONNA DO WHAT AM I GONNA DO FVCK #flooding

Lastly, tweets may contain a variety of tokens seen mainly in Twitter and other social media, such as: URLs; emoticons; Twitter hashtags, of the form #tagname, which the authoer may supply to label a tweet; Twitter at-mentions of the form @user which link to other Twitter users; and Twitter discourse functions such as RT ("re-tweet"), indicating that a tweet was originally posted by some other Twitter user, or ellipsis dots (...) at the end (or beginning) of a tweet, indicating that the tweet will be continued in a subsequent tweet by the same user. We note that hashtags and at-mentions can also serve as words or phrases within a tweet, as in:

> Heard about the #earthquake, stay safe everyone.

Regarding URLs, all links posted in tweets are shortened using Twitter's link service, `http://t.co`, and are converted to a seemingly random 23 characters URL that redirects to the original web address.

2 Analysis Wokrflow

keywords TODO

- A
- B
- C

TODO: (Gal) Complete this section

3 Classification of Disaster-Related Tweets

In the first part of our work we developed a classifier that identifies *disaster-related* tweets from *not disaster-related* tweets, trained on a dataset of 10,877 labeled tweets (TODO: note: maybe better to say that the dataset was 2400 tweets, since we looked only at tweets with confident > 0.9). We experimented with a Naive Bayes (NB), random forest (RF) and support vector machine (SVM) classifiers.

Naive Bayes TODO: (Daniel) short explanation

Support Vector Machine (SVM) TODO: (Daniel) short explanation

Random Forest TODO: (Daniel) short explanation

3.1 Feature Extraction

To train the classifiers we extracted several features for each tweet:

- **Unigrams and bigrams** of tokens in tweet; we used a Python version of `Twokenizer`, tokenizer for Twitter data (Gimpel et al. [2], Myle Ott, 2013 [3]).
- **Tweet metadata**; hashtags, at-mentions and URLs; we parsed the tweet, crawled URLs and used the referred webpage title as supplementary information.

- **Part-of-speech (POS) tags** (bigram and unigram); we used a twitter-specific tagset and tagger presented by Gimpel et al. [2].

Tokenization Splitting a tweet to tokens (separated ordered words) is hard due to irregular punctuation patterns and use of punctuation marks for emoticons. For example, the correct tokenization of 'hello (#hashtag)' is to the four tokens [hello , (, #hashtag ,)], but 'hello (:' should be split only to the tuple [hello , (:].

`Twokenizer` is a tokenizer designed for Twitter text in English that addresses these issues. It was originally developed in Python by O'Connor et al., 2010 [1], then improved and ported to Java by Gimpel et al., 2011 [2] and later ported back to Python [3]. We use the last version by Myle Ott.

Metadata Extraction The hashtags, at-mentions, URLs and emoticons in a tweet carry information that may help better understand the subject and context. Therefore, for each tweet we created a vector of the following metadata features, which we found to be the most expressive:

- Does the tweet contain a link?
- How many link are in tweet?
- Is '`https://twitter.com`' one of the links?
- Does the tweet contain a user at-mention?
- Does the tweet contain a hashtag?
- How many hashtags are in tweet?
- Does the tweet contain a happy emoticon?

In addition to these features, we extracted information from hashtags and URLs. We attempted to split hashtgs to separate words looking for a CamelCase pattern (for example, #JeSuisCharlie → 'Je Suis Charlie') or words separated by underline. We note that this method is not exhaustive since Twitter users tend to create hashtags composed of joined words, all lower-case.

We also found that the domain name of the URLs in a tweet may give an indication to whether the tweet is disaster-related or not (for example, a link to `https://9gag.com` is a negative hint). However, Twitter applies URL shortening on links, so for each shortened link in the dataset we attempted to reach the original Internet address. We also collected the HTML page title, which often contains the title of an article (for example, in news sites). We managed to expand 4,823 URLs out of 6,157 in the dataset.

For each tweet in the dataset we created an 'extended' version with extracted hashtag, expanded URL, and where every user at-mention is replaced by the token __USERREF__. For example, the following is a tweet,

TODO: Gal

and its extended version is,

TODO: Gal

Twitter POS Tagging TODO: Gal [2]

3.2 Results

TODO: Daniel?

4 Sentiment Analysis of Tweets

TODO:

5 Named-Entity Recognition in Tweets

TODO:

6 Experimenting with Recent Tweets

TODO: (Gal) Complete this section

keywords Twitter's Search API

7 Conclusions

Future work TODO

References

[1] Brendan O'Connor, M. K., and Ahn, D. Tweetmotif: Exploratory search and topic summarization for twitter. In *Proc. ICWSM* (2010).

[2] Kevin Gimpel, Nathan Schneider, B. O. D. D. D. M. J. E. M. H. D. Y. J. F., and Smith, N. A. Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proc. ACL* (2011).

[3] Ott, M. Python port of `Twokenizer`. `https://github.com/myleott/ark-twokenize-py`, 2013.

Notes

[1]**"Disasters on social media" Dataset by CrowdFlower**: Contributors looked at over 10,000 tweets culled with a variety of searches like "ablaze", "quarantine", and "pandemonium", then noted whether the tweet referred to a disaster. `https://www.crowdflower.com/wp-content/uploads/2016/03/socialmedia-disaster-tweets-DFE.csv`