Diagnosing the Disaster Tweets

An NLP Project by Jon Rosenberg
conducted in participation in Springboard

Preprocessing

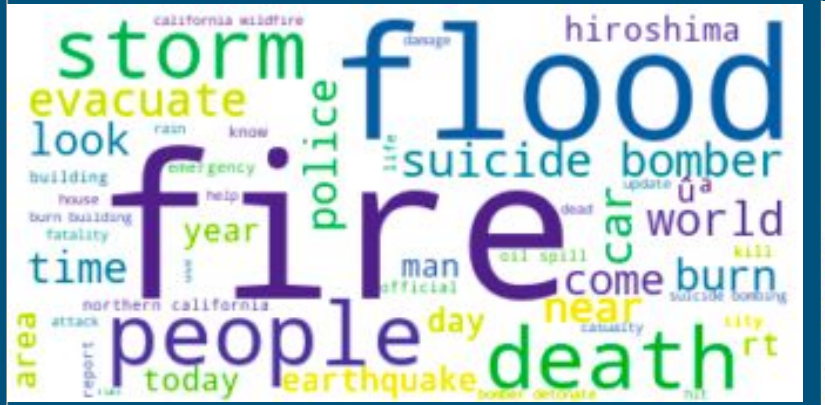
- Working in NLP is largely preprocessing and then model application

- In preprocessing we:
  - Reduce tokens to lemmas
  - Remove stop words
  - Remove excess spaces and punctuation (but are careful to keep emoticons for our use case)
  - Clean out some garbled characters and out of vocabulary words

- We create an TF-IDF representation of our corpus for traditional machine learning methods.

Exploration

- NLP problems typically don’t offer many avenues for summary of the space.
- When classifying though as in our case, word clouds can be descriptive:

Disaster	Safe
	

Machine Learning

- We have a classification problem in a high dimension space (the vocabulary of our tweets), so linear models are intractable. There are powerful non linear methods we can use, and all train to higher than random accuracy:

Model	Random Forest	XGBoost	Naive Bayes
Accuracy	0.79	0.78	0.67

Neural Networks

- In NLP neural networks are often used to learn the relations between words
- These can be used as starting layers for other classification networks
- We can also feed our tweets as sequences to Neural Networks using LSTM layers to factor the context of words into our final classification
- Our end accuracy with such a network: 0.79

Next Steps

- The random forest model should be chosen going forward as it had the highest accuracy, is relatively simple to explain, and trains quickly
- We should aim to obtain more data, and try other preprocessing filters on the words in our input vocabulary
- We should try splitting our models on particular type of disaster. And consider factors such as location and trending volume of tweets