**Real or Not?**

Classification of Disaster and Emergencies on Twitter

**Abstract**

Natural disasters and other emergency scenarios require fast response and a rapid evaluation of a situation. One possible attempt at leveraging large amounts of data comes in the form of social media, specifically tweets from the users of the social media network *Twitter*. Researchers hope that creation of an accurate real-time classifier for tweets related to natural disasters and other emergencies (crime, suicide, power outages) could help to inform emergency response. Three model classifiers were compared in their F1-Score accuracy to classify tweets as either related to natural disasters or not. The *Multinomial Naive Bayes* classifier was found to be the most accurate with a F1-Score of 0.7539. ‘Model simplicity’ in the form of not removing stop words, and only utilizing unigrams was shown to be associated with higher accuracy scores, though these models were difficult to interpret. Removal of stop words was noted to be especially useful in interpreting the model and interpreting which words strongly associated with disasters or not. The model struggled in correctly classifying tweets which contained juxtapositions of emergency related words with otherwise innocuous content.

**Background**

Text mining in conjunction with topic modeling and other analysis has been done to analyze public perception of virus outbreaks (Lazard et al., 2015), sentiment analysis (Kang et al., 2018), natural disasters (Huang et al., 2010), and other events. In leveraging text mining to the field of natural disasters researchers hope that text mining can aid in the real-time response to these emergencies or gain insight more quickly (Goswami et al., 2016). Of all natural disasters earthquakes, tropical storms, and flooding events account for the majority of global death toll (Guo, 2010). A large-scale text classifier could aid in the emergency response to these and other emergency scenarios.

Multinomial Naive Bayes, Bernoulli Naive Bayes, and Support Vector Machines (specifically support vector classifiers) are common classifiers used to associate text with a given score or classification. Each technique relies on passing the training text with a score or category it is associated.

Multinomial Naive Bayes uses a parameter learning method called Frequency Estimate (FE), which estimates word probabilities by computing frequencies those words are associated with a given category (Chen et al., 2009). In simple terms, the Bernoulli Naive Bayes works similarly to the Multinomial Naive Bayes, except that Bernoulli Naive Bayes only recognizes the presence or absence of a feature and Multinomial Naive Bayes recognizes the count of a given feature. Support vector machines work to create ‘decision boundaries’ between the given categories.

**Data**

The dataset is a collection of tweets and comes from a kaggle challenge called ‘real or not.’ The data consists of a numerical id, a keyword associated with a given tweet, the location of the tweet, the text, and a column that states whether the text is a person tweeting about an actual natural disaster or if they are tweeting about something different. The dataset (already split into training and testing) can be found at the link below. For this analysis only the tweet and the target column (disaster or not) were used for analysis.

<https://www.kaggle.com/c/nlp-getting-started/data?select=test.csv>

The training dataset is mostly balanced as there are 3,271 tweets about natural disasters and 4,342 tweets not about natural disasters.

**Classifier Tuning**

For each classifier, hyperparameters, for the classifier and vectorizer, were tuned manually, one at a time, and the F1-Score was compared to choose the most accurate model. In all cases examined removing stop words from the vectorizer reduced accuracy of the given classifier. Further, reducing the max\_df parameter from its default value of 1, which eliminates words past a certain document frequency, also was found to only decrease the accuracy of the model. Finally, using a count vectorizer with bigrams or trigrams was never found to increase the accuracy of the model. After running the initial base classifier, each parameter tuning was also run on tweets which had been stemmed using the porter stemmer. A small, but insignificant <0.01 accuracy change was noted in each model after using the stemmer.

**Description of Problem**

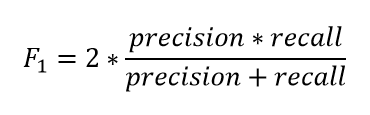
Creating a model which can understand the intricacies of human language is extremely useful. The classification of whether text is truly about a natural disaster or is a hyperbolic statement underlies many fundamental classification problems and improvement in these areas will allow greater application in varying fields of text mining and text classification.

More specifically, an algorithm which has the ability to properly classify tweets as relating to a natural disaster, or any other given event of interest, automatically gives researchers a large corpus of text data which can be studied to determine the occurrence and impact of the occurrence.

**Problem Modeling**

The goal of this project was to create an accurate classifier which can clearly distinguish between tweets which are related to natural disasters and those that are not. Multiple classifiers were analyzed and compared to create the most accurate classifier. *Multinomial Naive Bayes, Bernoulli Naive Bayes,* and *Support Vector Machine* (with a linear kernel) classifiers were compared to create the most accurate model for tweet classification.

Each model was tuned by using a random search of hyperparameters. Model accuracy was graded by using the F1-score (equation 1). The F1-score was chosen as it best represents a balance between precision and recall. Precision is a measure of the true positives vs. false positives. Recall is a measure of the true positives vs. the false negatives.



**Results**

The naive bayes methods (Multinomial and Bernoulli) were found to be the most accurate. The multinomial Naive Bayes was found to perform slightly better than the Bernoulli Naive Bayes method. Although not removing stop words and adjusting min\_df to more stringent thresholds lowered the model accuracy, each of these tuning parameters also made the model easier to interpret for a human user.

|  |  |  |  |
| --- | --- | --- | --- |
| **Classifier** | **F1-Score (not stemmed)** | **F1-Score (Porter Stemmed)** | **Parameters** |
| BernoulliNB | 0.7519 | 0.756 | Min\_df = 3  Stop\_words = None |
| MultinomialNB | 0.7539 | 0.742 | Min\_df = 1  Stop\_words = None |
| Support Vector Machines | 0.7227 | 0.725 | Min\_df = 14  Stop\_words = None |

**Challenges**

As we saw in week 3 when attempting to apply *Multinomial Naive Bayes* to a dataset pairing movie reviews to a numerical score, many intricacies of human language are difficult to capture in a model. For example, it can be difficult for naive bayes to detect sarcasm and hyperbole. Thus, tweets using hyperbolic language, or language associated with natural disasters may be difficult to classify as non- natural disasters. As an example, a tweet which stated “On plus side LOOK AT THE SKY LAST NIGHT IT WAS ABLAZE,” may be difficult to recognize as referring to the sky being ‘ablaze’ in beauty (and not literally ablaze).

Although it was noted that the accuracy of the model decreased after removing stop words, not removing stop words also makes the model difficult to analyze. For example, if one wishes to understand which words are most strongly associated with natural disasters or which words or most commonly associated with non-disaster tweets they can print the logarithmic probability of the classifier features. In a model in which stop words have not been removed these strongly associated words are full of meaningless terms such as ‘by’, ‘for’, ‘in’, ‘with’. Removing stop words, decreases overall accuracy, but makes the analysis of the most important non-trivial words more straightforward. After removing stop words the 10 words with highest probability of being associated with disasters and those not associated with disasters are listed below.

**Not-Stemmed**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Disaster** | police | people | killed | disaster | suicide | california | news | https | amp | u |
| **Non** | got | video | don | body | u | new | amp | just | like | https |

One final challenge was the use of stemming. Stemming resulted in only small changes to the accuracy of the model, but stemming can theoretically make the model more difficult to interpret. This can occur if overstemming occurs or words are stemmed to where they are meaningless to the user. The table below shows the 10 most associated words with each category. In this case the stemmed words for each category are more difficult to analyze and more are shared between the disaster and non-disaster category.

**Stemmed**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Disaster** | ha | flood | new | amp | kill | u | thi | bomb | wa | http |
| **Non** | bag | ha | u | new | amp | wa | just | this | like | http |

**Conclusions**

Creation of an accurate classifier which can discern when individuals are writing about a natural disaster would be useful in applying to real-time disaster response. Three different classifiers, *Multinomial Naive Bayes, Bernoulli Naive Bayes, and Support Vector Machines (Linear Kernel)* were used to evaluate the best model in regards to F1-Score. In tuning the vectorizer and model parameters it was found that the most accurate models did not remove stop words, only considered unigrams, and did not benefit from stemming. In model evaluation there is a clear trade-off between model accuracy and model interpretation as models with stop words removed, and stemmed also can create difficulties in interpretation. The most accurate model created utilized the *Multinomial Naive Bayes* algorithm. Although, this model performed with an F1-score of ~0.75, there were still cases in which the model performed poorly. Most notably juxtapositions with words related to natural disasters in other contexts were often misclassified.

**Bibliography**

Huang, C., Chan, E. and Hyder, A., 2010. Web 2.0 and Internet Social Networking: A New tool for Disaster Management? - Lessons from Taiwan. *BMC Medical Informatics and Decision Making*, 10(1).

Lazard, A., Scheinfeld, E., Bernhardt, J., Wilcox, G. and Suran, M., 2015. Detecting themes of public concern: A text mining analysis of the Centers for Disease Control and Prevention's Ebola live Twitter chat. *American Journal of Infection Control*, 43(10), pp.1109-1111.

Kang, M., Ahn, J. and Lee, K., 2018. Opinion mining using ensemble text hidden Markov models for text classification. *Expert Systems with Applications*, 94, pp.218-227.

Goswami, S., Chakraborty, S., Ghosh, S., Chakrabarti, A. and Chakraborty, B., 2018. A review on application of data mining techniques to combat natural disasters. *Ain Shams Engineering Journal*, 9(3), pp.365-378.

Chen, J., Huang, H., Tian, S. and Qu, Y., 2009. Feature selection for text classification with Naïve Bayes. *Expert Systems with Applications*, 36(3), pp.5432-5435.