Real or Not? NLP with Disaster Tweets EDA

Vegar Andreas Bergum, *vab1g19,* ChangKai Fu, *ckf1n19,* Adam Ghoumrassi, *ag3u19,* PingChun Tsai, *pct1g19.*

***Abstract*—This exploratory data analysis of the *Real or Not? NLP with Disaster Tweets* Kaggle competition dataset looks at possible underlying patterns in the textual data contained in tweets. The analysis includes syntactical features such as punctuation, spelling mistakes and word frequencies, and semantic features such as sentiment, emojis and bigrams. Further, we look at the top splits in a decision tree and perform latent semantic analysis in an attempt to uncover lower-dimensional patterns. The analysis reveals some underlying differences between the classes and demonstrates how machine learning can be used on this classification problem.**

I. Introduction

THIS project considers the *Real or Not? NLP with Disaster Tweets*[1] Kaggle competition, where one attempts to classify whether or not a tweet is announcing a real disaster. As stated in the competition; ”Twitter has become an important communication channel in times of emergency“, both by government organisations and individuals. With this comes the challenge of fake news and the general authenticity of the information being spread online. With the capability of quickly and reliably classifying any given tweet as announcing a real disaster or not, one can help improve automatic disaster monitoring.

With any machine learning project, one of the most important steps is exploring the data available. In this competition, we have access to $7,613$ labelled tweets (fake and real). Most of this data is pure text (the content of the tweets), which makes this a natural language processing (NLP) task. Feature extraction is very important and complex when working with textual natural language. There are several methods for representing the semantics of text, including word-frequency based approaches such as TF-IDF and more advanced latent space representations such as word embeddings like *word2vec* [1]. In order to get a better understanding of the existing features in the data and basis for choosing a semantic representation model, we explore both token- and character-level relationships between the data and the two classes.

II. Data Summarization

The data set contains $7,613$ tweets, labelled as fake and real. Excluding the label, the data has three main attributes: *keyword, location, text*. The keyword attribute contains a list of pre-defined keywords that describes the contents of the tweet. The location is provided directly by Twitter’s geo-tag on a tweet and is represented as a string, e.g. *New York City*. The text attribute holds the vast majority of the data, as this is the raw textual content of the tweet. Table I shows the percentage of missing values and unique entries in each attribute.

Attribute	Missing values	Unique entries
keyword	0.8%	61
location	33.3%	2,533
text	0.0%	7,613

TABLE I: Missing and unique values in the data attributes.

As can be seen from the table, a third of the location data is missing. The keyword attribute is only missing a relatively few number of entries, although it should be noted that the number of unique keywords is quite low. Further, Figure 1 shows the proportions of tweets in both classes. There is a slight unbalance in the dataset, however, this is considered low enough to not have any substantial effects on the problem modeling.

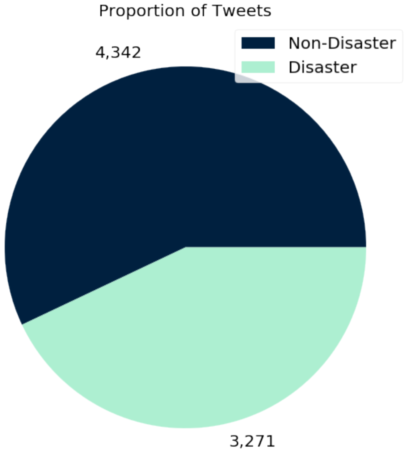

Fig. 1: Class proportions.

III. Quantitative Statistics

Exploring the textual data in the corpus reveals some correlations between the various syntactic and semantic features, and the two classes. Figure 2 shows that there is a small shift in the mean of the average number of words per tweet between the two classes. Non-disaster (fake) tweets are, on average, shorter and have a larger variation in tweet size. This implies that the tweet length could be a useful feature for classification.

Emojis have a prominent presence in social media, also on Twitter. Unfortunately, the text in the dataset is encoded in such a way that emojis can not be distinguished from each other. However, one can still look at the frequency of their use. *A priori* one could argue that a real disaster tweet is often from a mainstream news source and would stray away from using informal language and symbols, such as emojis. Figure 3, showing the frequency of emoji-usage, confirms this hypothesis. Non-disaster tweets use emojis more regularly than real disaster tweets. Since emojis are encoded as a special

[1] https://www.kaggle.com/c/nlp-getting-started/overview

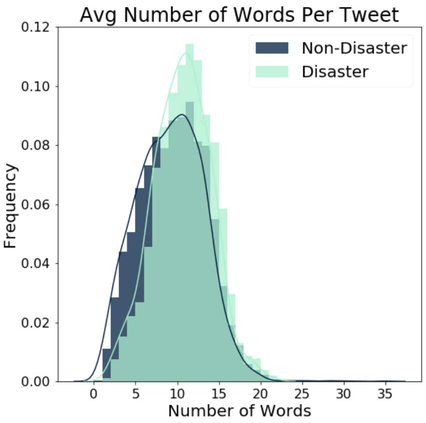

Fig. 2: Distribution of average number of words per tweet.

escape-sequence they can be considered as a semantic token equivalently to a regular word when modeling the problem for classification.

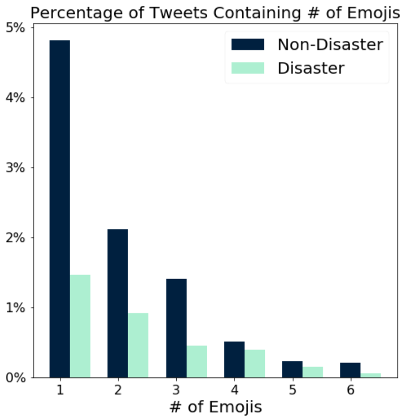

Fig. 3: Frequency of emoji-usage in the two classes.

The use of external links has shown to be a large differentiator between the two classes. 50% of non-disaster tweets contain external links, compared to 78% of disaster tweets. A rationale for this difference is that disaster tweets often refer to a source or proof of the disaster. Looking into the links being used, we can observe a difference in pages that are linked to from the two classes. Figure 4 shows the top 15 referred-to domains by the two classes. Observe that the disaster tweets are linking to official news sites such as `bbc.co.uk`, `latimes.com`, `cnn.com`, while the non-disaster tweets link to other social-medias sites such as `facebook.com` and `instagram.com`, `ebay.com` and various blogs.

When considering the use of punctuation in the tweet texts we observe a few differences between the two classes. Symbols such as { and > appear exclusively in the non-disaster tweets, although at a very low frequency. Other symbols such as `=, / and `` ` `` also vary largely between the two classes. By manual inspection of the tweets that account for these differences there are no obvious reason for the large difference.

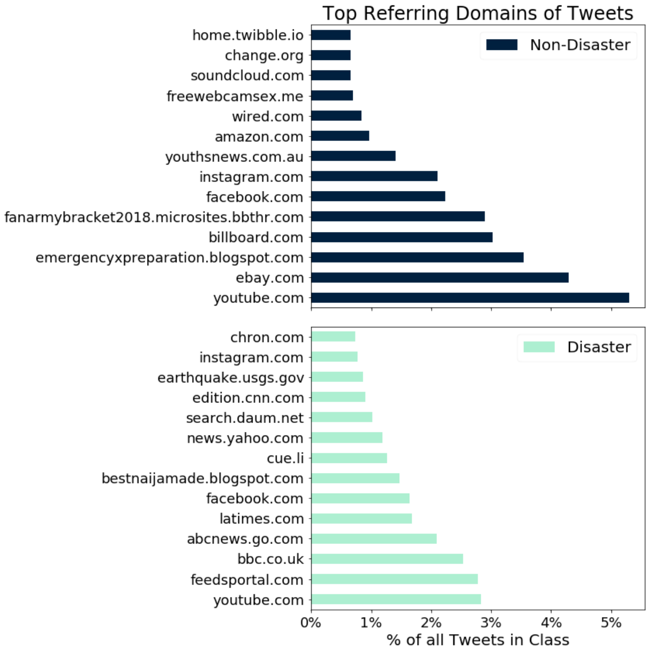

Fig. 4: Top domains for links in tweets.

Since the highest varying punctuation symbols also are the least used symbols, we assume that the large variation is simply due to the sampling bias of such a low frequency of their use.

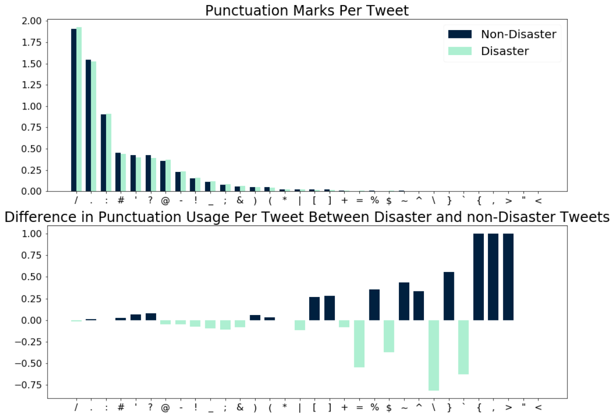

Fig. 5: Use of punctuation per tweet and differences in use between classes.

Figure 6 illustrates some of the different linguistic features of the two classes by showing the distribution and difference in part-of-speech tags per tweet. Notably, the disaster tweets more often use numbers and proper nouns. One explanation of this difference would be the larger amount of references to statistics (numbers) and official bodies and names (proper nouns) in real disaster coverage. However, the difference is very minimal for all POS-tags, so this could simply be an arbitrary difference due to noise. The `en_core_web_sm`

POS-tagger in `spacy` was used for tagging.

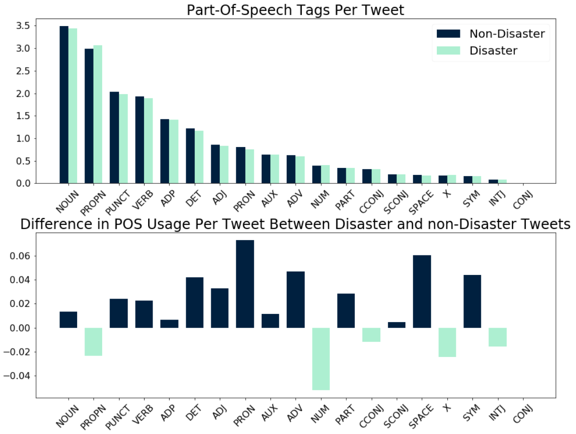

Fig. 6: Distribution of POS tags per tweet and differences between classes.

Using a Hamilton distance of 1 between two tokens and the WordFrequency project[2] dictionary to correct spelling, we evaluate the number of spelling mistakes per tweet, per class. Figure 7 shows the number of spelling mistakes for both classes. We observe that except for tweets with a single spelling mistake, disaster tweets contain more mistakes than non-disaster tweets. These differences are very small and lay within the margin of error to be expected from this approach to detecting spelling mistakes.

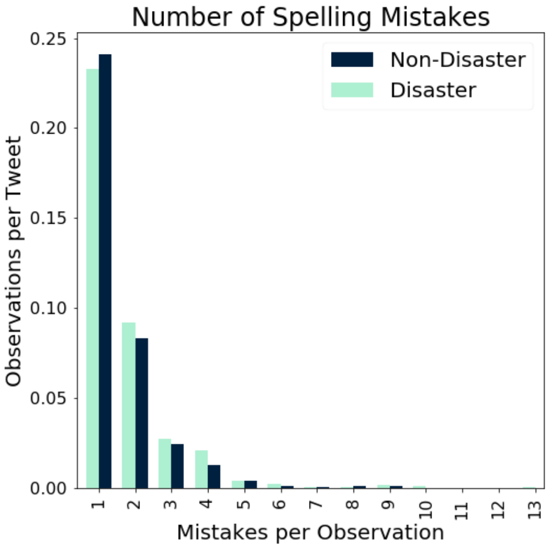

Fig. 7: Frequency of spelling mistakes per tweet.

IV. Sentiment analysis

As we have seen from the investigation into the use of external links, many of the real disaster tweets are referring to large mainstream news sources. News is supposed to be objective and carry little to no sentiment. Using VADER Sentiment Analysis[3], a rule-based sentiment analysis tool that is specifically made for social media text, we investigate the correlation between sentiment and subjectivity, and classes. Figure 8 shows a distribution and density of sentiment and subjectivity scores. Note that the two classes are laid on top of each other in both figures. A negative sentiment value naturally corresponds to negative sentiment, positive values correspond to positive sentiment and 0 corresponds to a neutral or non-sentimental tweet. A subjectivity score of 0 corresponds to an objective text, where 1 corresponds to maximum subjectivity.

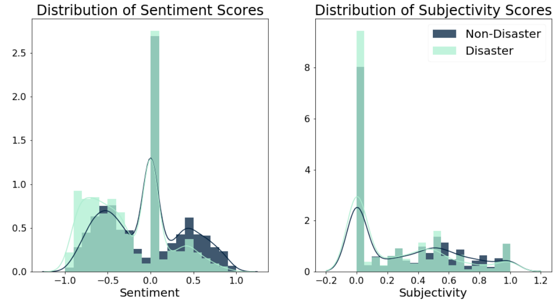

Fig. 8: Distribution and sentiment

Observe that disaster tweets are more frequently objective and neutral. Additionally, non-disaster tweets carry more positive sentiment while disaster tweets are negative more often. Non-disaster tweets are also slightly skewed towards being more subjective than real disaster tweets. These observations all follow the intuition of disaster tweets being either mostly neutral and objective (reported news) or carry a negative sentiment.

Class	Negative	Neutral	Positive	Subjective
Non-Disaster	0.132	0.766	0.102	0.324
Disaster	0.173	0.777	0.049	0.265

TABLE II: Mean values of negative, neutral and positive sentiment as well as mean subjectivity of tweets for each class.

Table II shows the mean values of sentiment and subjectivity for the two classes. These values further quantifies the differences. Disaster tweets are 20% more objective, 71% less positive and 27% more negative than non-disaster tweets.

V. Feature Exploration

To explore some of the most prominent textual and semantic features in the dataset, we consider a few models that are suitable for extracting said features. Figure 9 shows the first few nodes in a decision tree trained based on word frequency. The tree classifies the tweets with an F1-score of 0.76. Notably, from the top of the tree we observe that words such as *suicide, California, killed, ISIS, bomb* and *west* are immediately recognised as features contributing to a good split. These are also tokens that one would consider

[2] https://github.com/hermitdave/FrequencyWords

[3] https://github.com/cjhutto/vaderSentiment

as substantial in differentiation between a disaster and non-disaster tweet. This shows that the data contains useful features for class separation.

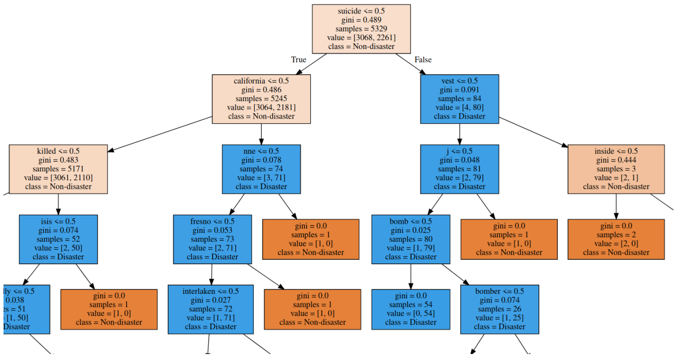

Fig. 9: First few nodes of a decision tree achieving an F1 score of 0.76.

However, the vocabulary size of the textual data is over 7000 unique words. Data with such a high dimensionality can cause some problems when attempting to extract useful features. Using PCA, we attempt to reduce the dimensionality (substantially) and visualise the results to see if there are any evident patterns, even at 2- and 3D. Figure 10 shows a 2D PCA projection of the textual data. Other than the slightly higher variation in the non-disaster data, including a few outliers, there are no evident patterns that helps separate the two classes. The same results are evident in the 3D projection of the data.

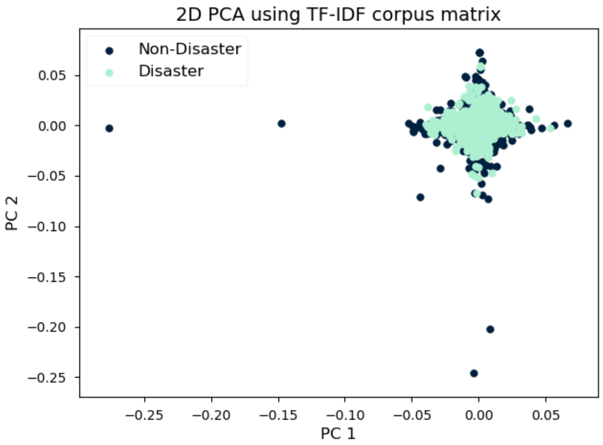

Fig. 10: 2 dimensional PCA using a TF-IDF corpus matrix.

From a probabilistic view, we can build a basic language model using the n-gram model, specifically a 2-gram (bigram) model. Using this Markov chain model we can investigate the most frequent bigrams from two models built on each data class. Figure 11 shows the top 10 bigrams for both classes, which illustrates some expected differences. Although the two classes share some frequent bigrams such as *burning building(s)*, the disaster model mostly contains bigrams that are clearly related to disasters. While the non-disaster language model contains what can be seen as more arbitrary bigrams of words. The disaster tweets also have a clearer variation in the frequency of its top 10 bigrams, where a few specific disasters are overrepresented, such as the Malaysian airlines disaster.

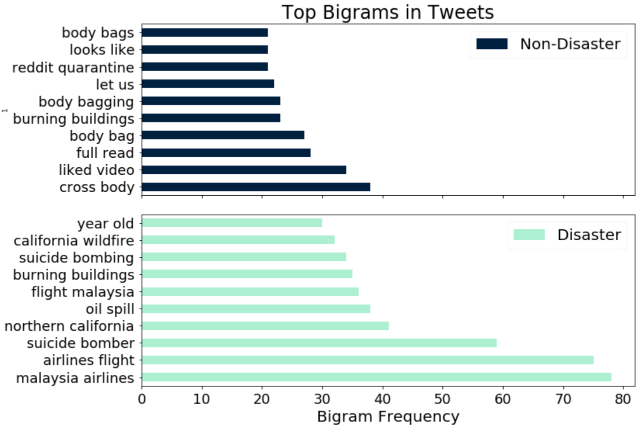

Fig. 11: Top 10 Bigrams.

VI. Conclusion

In the data exploration part of this project, we analysed the data layer by layer like peeling an onion. In the beginning, we utilized quantitative statistics to generate the number of words per tweet, frequency of emoji-usage, top (website) domains for links in tweets, use of punctuation, part-of-speech tags and the frequency of spelling mistake to observe the differences between the two classes using various characteristics and elements of the text. Fortunately, we did observe some patterns worth mentioning between the two classes. We observed that the tweets labelled as real tend to include less emojis, link to official news sites, and the distribution of the average words in disaster class shifts to larger number compared to non-disaster class. These mentioned patterns interested us to investigate the sentiment of the tweeters as real disaster tweets are more likely to be originated from news or a serious/negative emotion. Thus, the distribution of sentiment and subjectivity scores of tweets were analysed to see if we could have a clearer insight.

However, the results are still not obvious enough to differentiate the disaster and non-disaster classes as tweets data are very noisy. Therefore, we tried to extract the prominent features by using TF-IDF and a decision tree. The vectors processed by TF-IDF and PCA seems to have no clear evidence to classify the two classes as it is the standard way of performing a latent semantic analysis (LSA). In comparison, the decision tree model achieved an F1-score of 0.76, which is a good result and the power of machine learning is uncovered. In the next stage of the project, we will be using word embedding method such as word2vec, BERT and sequence-to-sequence RNN to find out the patterns between tiny differences which are hard to be found by the existing methods.

References

[1] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” *Advances in Neural Information Processing Systems*, vol. 26, 10 2013.