Natural Language Processing With Disaster Tweets

Abhiteja Mandava, Ishaan Bhalla, Sakruthi Avirineni, Yoonjung Choi
CMPE 255 Data Mining
San jose State University

***Abstract* – This paper is to predict disaster based on text from twitter and the problem is a supervised learning, binary classification and natural language processing. For data pre-processing, we have cleaned text from unnecessary text such as URL, Emojis or HTML tags and normalized text by using useful algorithms; tokenizer, stopwords and lemmatization. We created four numerical feature data sets by using Count Vectorizer, Inverse Document Frequency, word2vec and Word2Vec with PCA applied. We trained each numerical data set on different models; Decision Tree, SVM, Logistic Regression. We found each combination how each model has the higher performance on which data set. Then, we modified params of models to increase accuracy. Also, we trained ensemble model.**

***Keywords* – Natural Language Processing, Supervised Learning, Binary Classification, Bag Of Words, Count Vectorizer, TF-IDF, Word2Vec, Decision Tree, Logistic Regression, PCA, SVM, Ensemble, Voting Classifier, Recall, F1, Precision, Accuracy, ROC Curve**

I. Introduction

Social Network Service has been playing a crucial role in communication and Natural Language Processing has been widely used to analyze it and extract potential patterns. Twitter is one of the popular SNS platforms and many tweets has been delivered in emergency situation. Since there are demands for companies to utilize this tweets, we has investigated and developed natural language processes and prediction models to have the best performance.

A. Dataset

The dataset has been collected from the company figure-eight and originally shared on their 'Data For Everyone' website [1]. We found the dataset from Kaggle Competition [2]. It contains 7613 instance with the following features:

Feature	Dtype	Descrition
id	int64	
keyword	object	39 non-values
location.	object	2533 non-valus
text	object	tweetter content
target	int64	0:non-disaster, 1:disaster

Fig. 1. The table shows feature, type, and description.

II. Data Exploration

We use only two features 'text' and 'target'.The figure shows the percentage of feature 'target"s distribution.

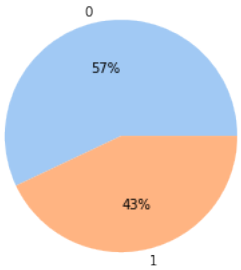


Fig. 2. there are 4342 instances for target(0) and 3271 instances for target(1)

In the Natural Language Processing, there are several common processing steps and many practical algorithms. We need to clean data, normalize data, create numerical feature vector.

A. Data Cleaning

We should remove unnecessary words to make sure it has meaningful values. For cleaning text, we have changed all words to lowercase, removed URL, HTML tags, Emojis, punctuation and ASCII codes.

text
Our Deeds are the Reason of this #earthquake May ALLAH Forgive us all
Forest fire near La Ronge Sask. Canada
All residents asked to 'shelter in place' are being notified by officers.No other evacuation or shelter in place orders are expected
13,000 people receive #wildfires evacuation orders in California
Just got sent this photo from Ruby #Alaska as smoke from #wildfires pours into a school

Fig. 3. The table shows partial of text

CleanText
our deeds are the reason of this earthquake may allah forgive us all
forest fire near la ronge sask canada
all residents asked to shelter in place are being notified by officers no other evacuation or shelter in place orders are expected
13000 people receive wildfires evacuation orders in california
just got sent this photo from ruby alaska as smoke from wildfires pours into a school

Fig. 4. The table shows partial of cleaned text

B. Data Preprocessing

Now, we have a cleaned text set and we should apply some methods to normalize words. NLTK [3] provides easy-to use interfaces for natural language processing

1) Tokenizer: Tokenizers divide strings into lists of substrings. We can use 'tokenize' library to find the words and punctuation in a sentences.

TokenizedText
['our', 'deeds', 'are', 'the', 'reason', 'of', 'this', 'earthquake', 'may', 'allah', 'forgive', 'us', 'all']
['forest', 'fire', 'near', 'la', 'ronge', 'sask', 'canada']
['all', 'residents', 'asked', 'to', 'shelter', 'in', 'place', 'are', 'being', 'notified', 'by', 'officers', 'no', 'other', 'evacuation', 'or', 'shelter', 'in', 'place', 'orders', 'are', 'expected']
['13000', 'people', 'receive', 'wildfires', 'evacuation', 'orders', 'in', 'california']
['just', 'got', 'sent', 'this', 'photo', 'from', 'ruby', 'alaska', 'as', 'smoke', 'from', 'wildfires', 'pours', 'into', 'a', 'school']

2) Stopwords: We should remove commonly used words (such as "the", "a", "is", "in").

RemoveStopWords
['deeds', 'reason', 'earthquake', 'may', 'allah', 'forgive', 'us']
['forest', 'fire', 'near', 'la', 'ronge', 'sask', 'canada']
['residents', 'asked', 'shelter', 'place', 'notified', 'officers', 'evacuation', 'shelter', 'place', 'orders', 'expected']
['13000', 'people', 'receive', 'wildfires', 'evacuation', 'orders', 'california']
['got', 'sent', 'photo', 'ruby', 'alaska', 'smoke', 'wildfires', 'pours', 'school']

3) Stemming: Stemming is the process of producing morphological variants of a root/base word. For example, words such as "Likes", "liked", "likely" and "liking" will be reduced to "like" after stemming. There are different algorithms for stemming. Porter Stemmer is one of them and is a basic stemmer. Its' advantages are straightforward and fast to run.

PorterStemmer
['deed', 'reason', 'earthquak', 'may', 'allah', 'forgiv', 'us']
['forest', 'fire', 'near', 'la', 'rong', 'sask', 'canada']
['resid', 'ask', 'shelter', 'place', 'notifi', 'offic', 'evacu', 'shelter', 'place', 'order', 'expect']
['13000', 'peopl', 'receiv', 'wildfir', 'evacu', 'order', 'california']
['got', 'sent', 'photo', 'rubi', 'alaska', 'smoke', 'wildfir', 'pour', 'school']

4) Lemmatization: Lemmatization is the process of grouping together the inflected forms of a word so they can be analysed as a single item, identified by the word's lemma, or dictionary form. Both stemming and lemmatization are word normalization techniques, but we can find the word in dictionary in case of lemmatization. For example, original words 'populated' changed 'popul' in Stemming, but it is not changed in lemmatization. Lemmatization is more better performed than Stemming [4]. We decided to apply lemmatization.

LemmatizedText
['deed', 'reason', 'earthquake', 'may', 'allah', 'forgive', 'u']
['forest', 'fire', 'near', 'la', 'ronge', 'sask', 'canada']
['resident', 'asked', 'shelter', 'place', 'notified', 'officer', 'evacuation', 'shelter', 'place', 'order', 'expected']
['13000', 'people', 'receive', 'wildfire', 'evacuation', 'order', 'california']
['got', 'sent', 'photo', 'ruby', 'alaska', 'smoke', 'wildfire', 'pours', 'school']

5) Data Visualization: After normalized text, we made data visualization by using word cloud. In disaster tweet's words, we can discover disaster related words; suicide, police, news, kill, attack, death, california, storm, flood. In the other hand, the non disaster tweets shows that time, want, great, feel, read, but also killed, injury or emergency are found.

Fig. 5. target(1) disaster tweets' words

Fig. 6. target(0) non disaster tweets' words

C. Create feature to numerical values

Bag of Words model is a simplified representation used in natural language processing. A text is represented as the bag of its words, diregarding grammer and describes the occurrences of words with in a document.

1) CountVectorizer: CountVectorizer can be used for bag of words model. This convert a collection of text documents to a matrix of token counts.

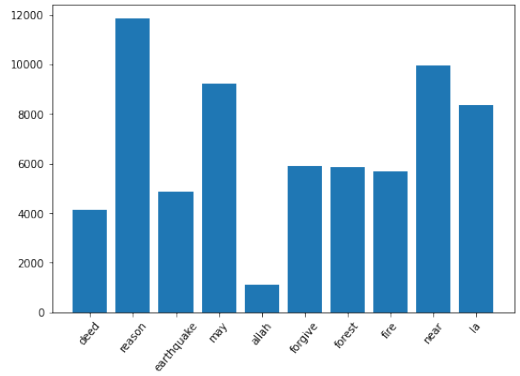


Fig. 7. Bar chart shows the number of ten words from dictionary

2) TF-IDF: The Term Frequency Inverse Document Frequency is a measure of whether a term is common or rare. It gives weight more to a term that occurs in only a few documents.

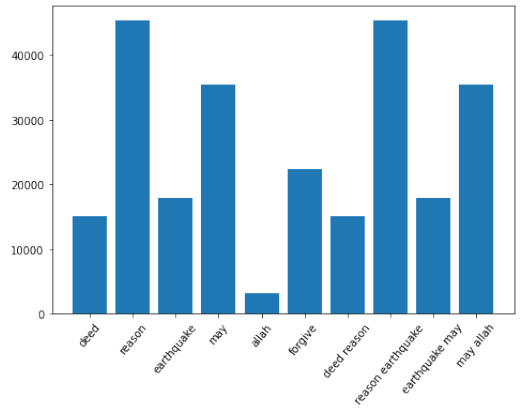
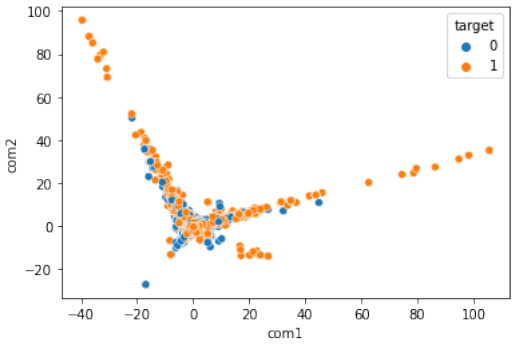


Fig. 8. Bar chart shows the number of ten words from dictionary

3) Word2Vec: The word2vec uses a neural network model to learn word associations from a large corpus of text.

4) Word2Vec with PCA applied: As principal component analysis is a strategy to reduce dimension, we applied PCA with 100 components on feature set from word2vec. The below figure is shown when applying PCA with 2 components.



III. METHODS

1) DecisionTree:

2) LogisticRegression:

3) SVM:

4) Ensemble: Ensemble methods are techniques that create multiple models and then combine them to produce improved results. Ensemble methods usually produces more accurate solutions than a single model would.

5) Results:

IV. CONCLUSION

Este artigo foi integralmente editado conforme as normas apresentadas para submissão de artigos em português.

REFERENCES

[1] OPPEN, URL: `https://appen.com/datasets-resource-center`.

[2] Kaggle, "Natural Language Processing with Disaster Tweets", URL: `https://www.kaggle.com /competitions/ nlp-getting-started/data`.

[3] NLTK, "Natural Language Toolkit", URL: `https://www.nltk.org/index.html`.

[4] Baeldung, "Naturalstemming-vs-lemmatization", URL: `https://www.baeldung.com/cs/stemming-vs-lemmatization`.

[5] T. A. Lipo, M. D. Manjrekar, "Hybrid Topology for Multilevel Power Conversion", US Patent 6 005 788, 21 Dec. 1999.

[6] *IEEE Recommended Practices and Requirements for Harmonic Control in Electrical Power Systems*, IEEE Std. 519-1992, 1993.

[7] SW Technologies, "SWDV Converter", Online, 2001, URL: `www.sw.com.br`.

[8] I. Barbi, *Etude de Onduleurs Autoadaptifs Destines a la Alimentation de Machines Assynchrones*, Ph.D. thesis, Institut National Polytechnique de Toulouse, Toulouse, França, 1979.

[1]–[8].