Disaster News Detection in Twitter

Zhixuan Duan (zhidu838) 732A92
Tuesday, March 1, 2022

Abstract

Twitter has become an important communication channel in emergencies, but fake news and disaster news are common on the Internet. This has caused users and emergency organizations to be plagued with messages. Disaster news is harder to detect than fake news because it's not always clear that a person's words are declaring a disaster. In this project, we will build a machine learning model that can predict which tweets are about real disasters and which are not. We will use the tweets of users on the Twitter platform to build a machine learning model.

1 Introduction

Billions of people around the world use social media sites like Facebook and Twitter, posting daily news on social platforms related to topics as diverse as politics, the economy, entertainment, sports and personal stories. Twitter has become one of the most popular ways to communicate during disasters[1]. Geotagged tweets are used to learn about affected areas, which can speed up the spread of disaster news and reduce economic losses and human casualties.

Machine learning is a branch of artificial intelligence, and it is a core technology for realizing artificial intelligence, that is, using machine learning to solve problems in artificial intelligence. Machine learning is through some algorithms that allow computers to automatically "learn" and obtain patterns from data analysis, and then use the patterns to predict new samples.

The following content is organized in the following order. First, we will discuss the work related to fake text and disaster text detection, then we will introduce the data set we use and conduct data set analysis on the text, and finally we will use machine learning to conduct modeling and discuss the accuracy.

2 Related Work

Disaster news detection is a text classification task[2]. The early method of text classification task is based on statistics, such as TFIDF combined with machine learning model. With the popularity of deep learning and GPU, deep learning methods based on word vector are also widely used. For example, TextCNN and TextRNN models can achieve higher accuracy[3][8]. At present, the latest model uses BERT model[9], which is the network structure based on Transformer.

Many researchers have studied this type of disaster classification on the twitter dataset or some specific news sites. Such as financial crisis news detection, hurricane news detection and flood detection[4][6][7]. Related research focuses on the specific crisis category, a text multi-classification task. And most of the existing high-precision models are deep learning models.

3 Dataset

3.1 Data Information

We selected the data set of Natural Language Processing with Disaster Tweets competition on Kaggle platform. This data set is consistent with our mission and contains specific tweet information and dependent variables. Below is a screenshot of the original tweet. The data set is stored in CSV format.

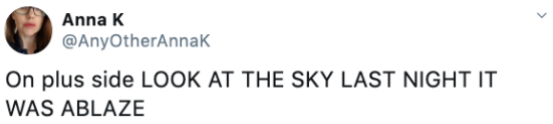


Figure 1. A screenshot of the tweet

The training set contains nearly nine thousand samples. The data contains two questions of training set and test set, and each sample in data set has the following information:

- id - a unique identifier for each tweet
- text - the text of the tweet
- keyword - a particular keyword from the tweet

- target - this denotes whether a tweet is about a real disaster (1) or not (0)

The keyword here is extracted from the original text, and not all samples contain the keyword, so it can be ignored in the modeling process.

3.2 Data Analysis

First we can look at some samples, shown in the table below.

Real disaster tweet	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.
Not disaster tweet	What's up man?
	I love fruits
	Ablaze for you Lord :D

Table 1. Data Example

Next, we counted the categories of the samples. The overall data categories are relatively balanced. The visualization results are shown in the following figure. The no disaster samples in the dataset account for 57%.

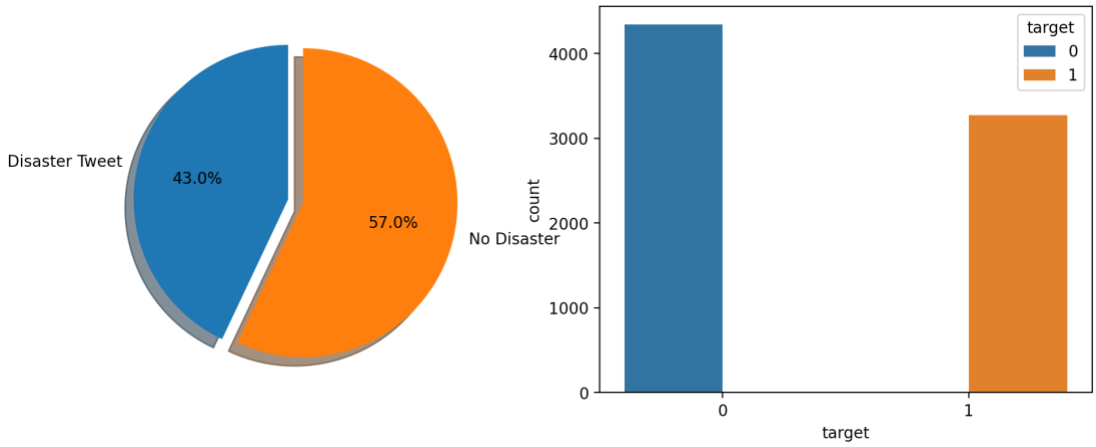


Figure 2. Target distribution

Next, we start to analyze the difference between real disaster text and no disaster text. We can make statistics from the following perspectives:

- the number of characters in a sentence
- the number of words in a sentence
- the number of stop words in a sentence
- the average number of characters per word in a sentence
- the number of capital letters in a sentence
- the number of values in the sentence

The figure below shows the visualization of the number of characters in the real disaster text and no disaster text sentences. It can be seen from the figure that the real disaster text contains more characters. This is also reasonable since disaster texts tend to contain more information and the texts will be longer.

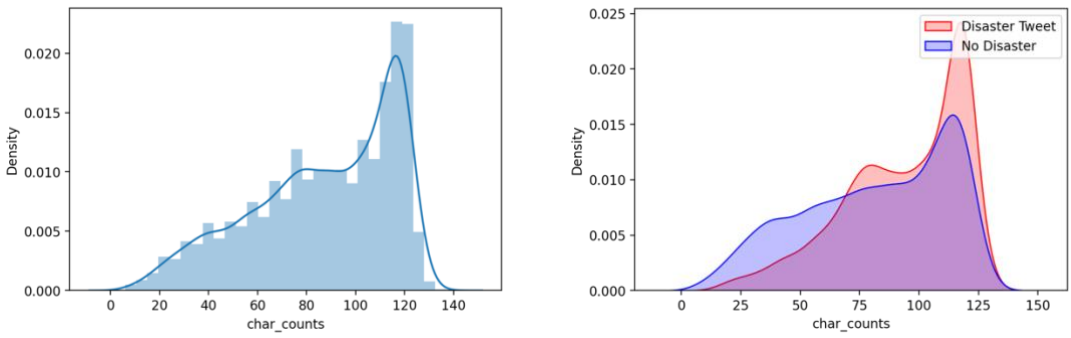


Figure 3. Char count in example

Next, we draw cloud maps for the real disaster text and no disaster text respectively. From the figure, we can see that the hot words are distinguishable.

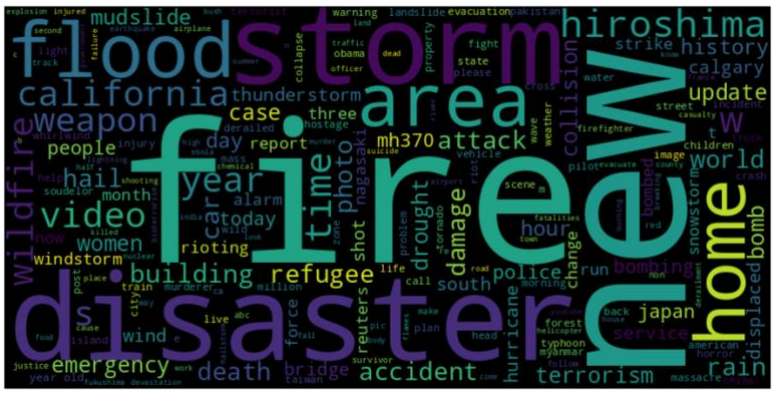


Figure 4. Word cloud for disaster text

Figure 5. Word cloud for no disaster text

It can also be seen from the cloud map that some words appear in both types of text, such as the word fire. These words can have a variety of meanings, which can represent the meaning of fire or environmental emotions.

4 Experiment

4.1 Model

In this section we introduce the model of multiple choice, we have chosen three text processing methods. We use countvector and tfidfvector from sklearn to convert text to a matrix, and logistic regression was chosen for training and prediction.

In countvector and tfidfvector, there are two parameter are highly related to model performance, that is ngram_range and max_features. ngram_range controls the lower and upper boundary of the range of n-values for different word n-grams or char n-grams to be extracted. max_features controls the max features in vocabulary.

We also use word2vec word vectors to map to text, then use logistic regression to complete training and prediction. Word2vec is a tool for converting words into vector form. Through word2vec, text can be simplified into vector space, and the similarity in the vector space can be calculated to represent the semantic similarity of the text.

A simple way to study word2vec is to find the closest word to a user-specified word. There are two main learning algorithms in word2vec:

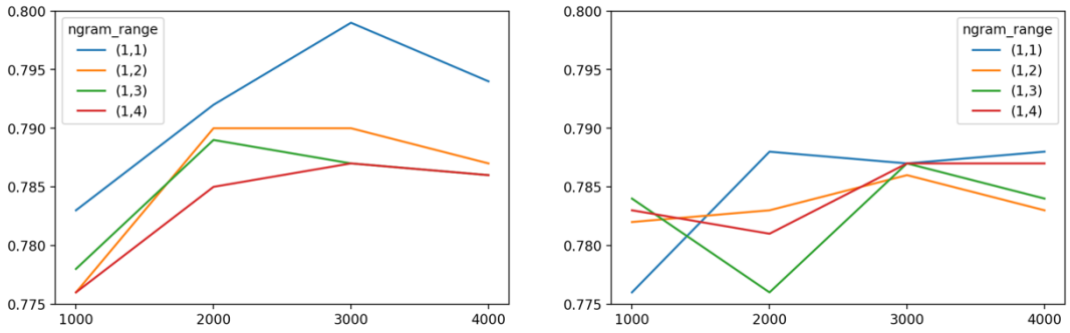
continuous bag of words and continuous skip grammar. We use pretrained vectors trained on part of the Google News dataset (about 100 billion words). The model contains 300-dimensional vectors for 3 million words and phrases.

4.2 Evaluation

We divide 25% of the training set into the validation set, and then use AUC for evaluation. To reduce the precision fluctuation caused by data division, we repeated all experiments 5 times and then took the average AUC.

4.3 Result

We first choose countvector and tfidfvector for experiments, as shown in the figure below, we compare the impact of ngram_range and max_features on model accuracy.



Tfidfvector countvector

Figure 6. AUC

It can be seen from the experimental results that the accuracy of tfidfvector is better than that of countvector. In addition, ngram_range can get the highest accuracy when the value is (1,1). This means only unigrams have the best effect.

For each word of the sentence, a vector can be obtained through word2vec, and the sentence can be spliced into a matrix. If the sentences contain different words, you will get sentences of different dimensions. If sentence A contains 20 words, the resulting matrix has a dimension of 20*300, and sentence B contains 30 words, and the resulting matrix has a dimension of 30*300.

To encode sentences into the same dimension, we average the matrix according to the word dimension to get the sentence vector, and then

use the sentence vector to send it into logistic regression to complete the training. Through the above steps, we can get an accuracy of about 0.78, which is lower than that of tfidfvector.

We also experimented with the effect of text cleaning on model accuracy by converting text to lowercase and replacing abbreviated words in it. Here, tfidfvector is selected for training and verification. The model results are as shown in the figure below. From the figure, the result from clean data is lower than the raw data.

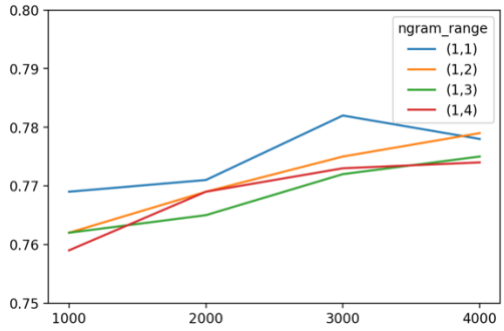


Figure 7. AUC(tfidfvector with clean data)

We summarize the results of all AUC in the table below.

Text	Model	AUC
Raw data	tfidfvector + LogisticRegression	0.80
Raw data	countvector + LogisticRegression	0.785
Raw data	word2vec + LogisticRegression	0.780
Clean data	tfidfvector + LogisticRegression	0.785

Table 2. AUC results table

In the above results, the accuracy of word2vec is lower than we thought, which may be caused by the following reasons. First, the pre-trained word2vec dictionary contains limited words, while tweets often contain more complex or colloquial words. Second, word2vec does not pay attention to the capitalization of words, but the case of text in tweets can represent emotional information.

4.4 Visualization

ELI5[5] is a Python package that helps debug machine learning classifiers and interpret their predictions. Here we will use it to

visualize the model, we first show the words that work on the real disaster text.

Weight?	Feature
+0.798	hiroshima
+0.748	fires
+0.705	california
+0.688	storm
+0.645	suicide
+0.633	fire
+0.620	train
+0.618	disaster
... 9682 more positive ...	
... 11946 more negative ...	
-0.570	you
-0.929	<BIAS>

Figure 8. Model test

We can see that these words have the meaning of disasters, such as storm, suicide and fire.

With the help of ELI5, we can also get the prediction results of the model for a single sample. As shown in the figure below, the first is a visualization of the not disaster text, and the second is a visualization of the real disaster text. Among them, the green color words represent the meaning of real disaster, and the red color words represent the meaning of not disaster.

y=Not Disaster (probability **0.537**, score **-0.149**) top features

Contribution?	Feature
+0.929	<BIAS>
-0.780	Highlighted in text (sum)

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

y=Real Disaster (probability **0.883**, score **2.026**) top features

Contribution?	Feature
+2.955	Highlighted in text (sum)
-0.929	<BIAS>

.@norwaymfa #bahrain police had previously died in a road accident they were not killed by explosion https://t.co/gfjfgtodad

Figure 9. Model visualization

5 Conclusion

This project introduces how to use text modeling to detect disaster news. We introduce the background of the problem, data sources and

data distribution laws. We also tried different text feature encodings and used AUC to evaluate the final accuracy. From the accuracy comparison results, tfidfvertor can bring the best accuracy, and finally we also use ELI5 to visualize the model prediction results.

To improve the accuracy of the model, the following perspectives can be considered. First, try a more powerful model, such as BERT or TextCNN, because such models will bring more powerful model accuracy. You can also consider adding the modeling of tweet location information. Disaster tweets often have location information.

References

[1] Sakaki, T., Okazaki, M., & Matsuo, Y. (2010, April). Earthquake shakes twitter users: real-time event detection by social sensors. In Proceedings of the 19th international conference on World wide web (pp. 851-860).

[2] Tomonto, Matthew. A Calamitous Imagination: Disaster Images, Fake News, and Challenges to Journalistic Objectivity. Diss. MA Thesis, Aristotle University of Thessaloniki. http://ikee. lib. auth. gr/record/304586/files/GRI-2019-24234. pdf, 2019.

[3] Ajao, Oluwaseun, Deepayan Bhowmik, and Shahrzad Zargari. Fake news identification on twitter with hybrid cnn and rnn models." Proceedings of the 9th international conference on social media and society. 2018.

[4] Biradar, Shankar, Sunil Saumya, and Arun Chauhan. Combating the infodemic: COVID-19 induced fake news recognition in social media networks." Complex & Intelligent Systems (2022): 1-13.

[5] Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. Why should i trust you? Explaining the predictions of any classifier." Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. 2016.

[6] Chee-Hong CHAN, Aixin SUN, Lim and Ee Peng, "Automated online news classification with personalization", 4th International Conference on Asian Digital Libraries (ICADL) Research Collection School Of Information Systems. INK, 2001.

[7] Abeer Khaleq, Ra and Ilkyeun, Twitter Analytics for Disaster Relevance and Disaster Phase Discovery: Volume 1", Proceedings of the Future Technologies Conference (FCT), pp. 401-417, 2018.

[8] Guo, Bao, et al. Improving text classification with weighted word embeddings via a multi-channel TextCNN model. Neurocomputing 363 (2019): 366-374.

[9] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.