Natural Language Processing with Disaster Tweets

Group 3: Khanh V Huynh, Mahalakshmi Ramasamy, Anne Sesnak, Mohan Singh

Final Report: 574

Professor Mike Stepanovic

March 12, 2023

Table of Contents

Practical Explanation

With large amounts of data existing in unstructured text form, methods such as Natural Language Processing (NLP) can harness that data and help us make sense of it. NLP has many applications, but one of the most widely and practically useful applications is text extraction and classification. Classification using NLP can provide information such as sentiment analysis, author identification, the scope of documents (such as for a trial), email filters, and more.

Smartphones are everywhere and allow the capture and announcement of emergencies and disasters in real time. Smartphone users take to social media to report these events. In this way, Twitter and other social media platforms have become an important communication channel during emergencies and one such source of unstructured text data. This has led to organizations and agencies, such as disaster relief organizations, city governments, and news agencies, becoming interested in monitoring social media for this type of information. The problem however is that it’s not always clear if a post’s content is actually announcing a disaster or not.

This is where NLP comes in. NLP models are able to parse text to differentiate certain language complexities like different words having the same meaning (ablaze, glowing, burning), different expressions having the same meaning (traffic jam, bottleneck, blocked lanes), the same word being used in different contexts (he’s wrecked, he wrecked his car), or different grammar structures supplying the same meaning (twice a week, biweekly, two times per week).

For this project, the group chose to use a Kaggle Competition dataset called, “Natural Language Processing with Disaster Tweets.” The objective of this competition and project is the classification of text in collected tweets to determine if the tweet is about a disaster or potential emergency or not.

This project with NLP is important because citizen reporters are providing valuable real-time insights and this real-time information distribution has many applications. NLP is a great method for this project because it handles unstructured data and can perform classification. The way this process works is by collecting the raw data, preprocessing and cleaning it, then running this data through NLP models.

About the Data

The dataset was supplied directly through Kaggle and includes over 10,000 tweets that have been pre-split into a training set and a testing set. The training set has already been classified to be able to train the model for the testing set. The target variable “target” specifies if a tweet is about a real disaster (1) or not (0).

While the data includes interesting variables like location and keyword, the variable we are focusing on is “text”, which is the raw and unstructured content of the tweet requiring cleaning and preprocessing before it is ready for the NLP models. The objective is a classification of the text as an actual disaster or not. Initial exploratory data analysis of the training data using “groupby” showed that more tweets were classified as not disasters than disasters, so we would expect similar results with the testing data.

The raw data in the “text” variable includes non-text characters, syntax, and punctuation. The Natural Language Toolkit (NLTK) library available in Python was used, specifically the StopWords and PorterStemmer packages to perform preprocessing tasks like removing stop words, removing punctuation, and reducing words to their roots. The group then used Term Frequency Inverse Document Frequency (TFIDF) to convert the text into weighted numerical values for the machine learning algorithms.

Theoretical Explanation

The problem that is handled in this project is classifying an immense amount of tweets posted on Twitter. The classification of the tweets is based on if there is any disaster element associated with the tweet or not.

As this is a classification problem, the following machine learning models are considered: **Logistic regression, Naive Bayes, Neural Network, Random Forest, and Support Vector Machine (SVM)**. The following models are used to get the best accuracy in classifying the tweets in this project.

Logistic regression: This statistical & supervised model is often used to predict and classify binary outcomes (yes or no, true or false, disaster or not, etc). The machine learning field is one of the fields where this algorithm is used. Regression analysis is a type of predictive modeling technique that is used to find the relationship between a dependent variable (usually known as the “Y” variable) and either one independent variable (the “X” variable) or a series of independent variables.

Naive Bayes: This is a very simple yet very efficient & fast model to process a massive amount of data and classify them for the respective problem. There are different types of Naive Bayes algorithms (Gaussian, Multinomial, and Bernoulli). The general use cases for this model are spam filtering, recommendation systems, sentiment analysis, etc.

Neural Network: This model is a deep learning model that works similarly to neurons in the human brain. To put it in simpler terms, the neural network is built off of several layers of neurons with better knowledge than the previous state until the desired output is reached. That is, artificial neurons are calculated from the previous layer of raw data or neurons using some calculations. After several iterations of such calculations, the desired output is achieved. One of the disadvantages of NN is that it works more like a black box and deciphering the reason for the output is harder.

Random forest: This model is a supervised learning technique and a popular machine learning algorithm. This algorithm is basically a collection of smaller decision trees and it is iterative, where the input features (hyperparameters) are evaluated and filtered based on the learning. The hyperparameters are essential for making the model faster and increasing the predictive power. The advantage of the random forest algorithm is it avoids overfitting and also has versatile use cases but the disadvantage of the algorithm includes slowness in computing.

Support Vector Machine (SVM): This algorithm is also a supervised learning algorithm that is typically used for text and image classification cases. The SVM algorithm's objective is to find the appropriate hyperplane (in multidimensional space) that classifies the data in the best possible way. Hyperplanes are decision boundaries that help classify the data points. The advantage of SVM is that it works very efficiently when there are several dimensions/features. The main disadvantage of this algorithm is that it is an expensive algorithm in terms of time, memory requirements, and complexity in interpreting the outcome.

Overall Accuracy Results and Discussion of Models

The results below provide us with an evaluation of the performance of each 5 machine learning models on this dataset. The metric used to evaluate the performance of each model is the accuracy score, which measures the proportion of correctly classified instances among all instances in the dataset.

Models	**Accuracy**
Logistic Regression	0.80
Multinomial Naive-Bayes	0.80
Support Vector Machines (SVM)	0.81
Random Forest	0.78
Neural Networks	0.43

From these results, it can be determined that Logistic Regression, Multinomial Naive-Bayes, and Support Vector Machines (SVM) generated a similar result, all at or above 80% accuracy. These models may be considered to have performed well and to have reliable predictions for the dataset. On the other hand, the Random Forest model showed a slightly less accurate score of 78%. This model may not perform as well as the other three models in predicting outcomes for this dataset. Finally, the Neural Networks model showed the lowest accuracy score of 43%. This suggests the complexity of the Neural Networks may not be well suited to this particular dataset and that it requires further optimization to improve the accuracy score.

One reason why these three models performed so well during testing could be that they were well-suited for this particular dataset. For example, logistic regression is a regression model that assumes a linear relationship between the inputs and outputs. Support Vector Machine, on the other hand, uses a function to transform the input data into a higher-dimensional space where it may be linearly separable. Multinomial Naive-Bayes, which is another Bayes' theorem, assumes the independence between the input features and performs well when the features are independent given the class label.

One reason why the Neural Networks did not perform well in this test could be that the overall architecture of the neural network was not properly designed and properly optimized. Neural networks require a lot of training data, and it is very sensitive to the choice of hyperparameters such as the learning rate and the regularization parameters. The model may fail or overfitting of the data will occur if the parameters are not chosen correctly. Careful tuning and optimization are required in order to improve the accuracy of this model.

Overall, the results demonstrated the importance of evaluating and comparing the performance of different machine learning algorithms on a dataset in order to determine the best model for the task at hand. While some models can perform better on certain datasets, they may not perform as well on others.

Applications and Impacts

The application for NLP classification of social media in a real-world disaster management context is great. The power of social media and citizen reporting can be harnessed to collect, process, and analyze crowd-sourced disaster-related data in real-time, faster than traditional methods. Providing critical information to key players in this manner can lead to better response times, more effective emergency management, and a more informed public during a disaster. In addition, the development of web maps or mobile applications can take advantage of individual accounts of the disaster to track changing conditions, inform evacuation decisions, and specify areas to avoid.

Of importance will be the success of the classification model to ensure that content is truly disaster-related. The goal of predicting disaster tweets is to provide early warning signals to emergency responders and the public to help them prepare for and respond to disasters. Predicting disaster tweets can provide early warning signals to emergency responders, allowing them to mobilize resources and respond more quickly to disasters. This can be especially useful in cases where traditional warning systems, such as sirens or emergency broadcasts, may not be effective. Predicting disaster tweets can help emergency responders gain a better understanding of the situation on the ground. By analyzing tweets from people who are directly affected by a disaster, responders can gain insight into the severity of the situation, the areas most affected, and the needs of the affected population. Predicting disaster tweets can help governments and emergency responders communicate more effectively with the public. By monitoring social media, responders can identify misinformation and rumors and respond with accurate information. Predicting disaster tweets can provide valuable data for disaster response planning. By analyzing the language and sentiment of tweets related to disasters, responders can gain insight into the emotional impact of disasters on the affected population, which can inform the development of trauma-informed response plans.

One plan could be to use a system that ingests tweets, performs the preprocessing and classification, and then saves those tweets that refer to a disaster, and sells the data to companies that would use it. Figure 1 (see Appendix) is an architecture model where our predictor model would sit inside the Twitter ecosystem. Any new tweet would flow into the system through the Write API and land in the ingestor. The ingestor would annotate and tokenize tweets so the data can be indexed.

From the Ingestor Service, the information would flow to Early Bird which stores the search indexes and categorizes data into hashtags. The data would then be passed to the blender where it is asynchronously sent to timeline discovery. The NLP processor will ideally do read operations on streams of data in the Early Bird service and categorize the data into relating to a disaster or not. If it’s a positive match, the tweet will be relayed to potential customers.

Conclusion

Natural Language Processing (NLP) is an interdisciplinary field of study that combines computer science, linguistics, and artificial intelligence to enable machines to understand and process human language. NLP is a critical technology in today's digital age as it is used to extract insights and meaning from vast amounts of unstructured text data. This technology has transformed the way businesses operate, and it has become an essential tool for researchers, healthcare professionals, and other industries. In this project, we explored the importance of NLP and its applications, focusing primarily on information extraction for classification.

Information extraction involves the use of algorithms to extract structured information from unstructured text data and can be used in various industries besides emergency management, for example, finance, healthcare, and law enforcement. In finance, information extraction is used to extract financial data from news articles, enabling traders and analysts to make informed investment decisions. In healthcare, information extraction is used to extract information from medical records, enabling healthcare professionals to make accurate diagnoses and provide appropriate treatments. As we saw in the classification of tweets, information extraction can be used for real-time data classification and dissemination.

In conclusion, our data science project aimed to predict the likelihood of whether a tweet is about a disaster or not using five different classification algorithms: Naive-Bayes, Logistic Regression, Random Forest, Support Vector Machine, and Neural Networks. After thorough data cleaning and preprocessing, we trained and tested each of the algorithms on the dataset, and evaluated their performance based on various metrics such as accuracy and precision. Our results showed that the Neural Network had the lowest accuracy score among the four algorithms, but had the highest error rates for detecting disaster tweets. Logistic Regression had a similar performance to Naive-Bayes, while Support Vector Machine had the highest accuracy. Interestingly, Logistic Regression had the highest accuracy on the training set, but its performance on the testing set was lower than Random Forest and Support Vector Machine. This suggests that Neural Networks may have to overfit the training data, and further tuning may be required to improve its generalization performance.

In summary, each of the five algorithms had its own strengths and weaknesses in detecting disaster tweets. Naive-Bayes and Logistic Regression had better precision and recall for detecting disaster tweets, while Random Forest had a slightly lower accuracy, and Support Vector Machines had higher accuracy. Neural Networks showed promise in their high accuracy on the training set, but, as mentioned, more work is needed to improve their generalization performance. Overall, our project highlights the importance of selecting the right classification algorithm based on the problem at hand and the available data. Further work can be done to improve the performance of the algorithms and explore other techniques such as ensemble methods and deep learning.

Appendix

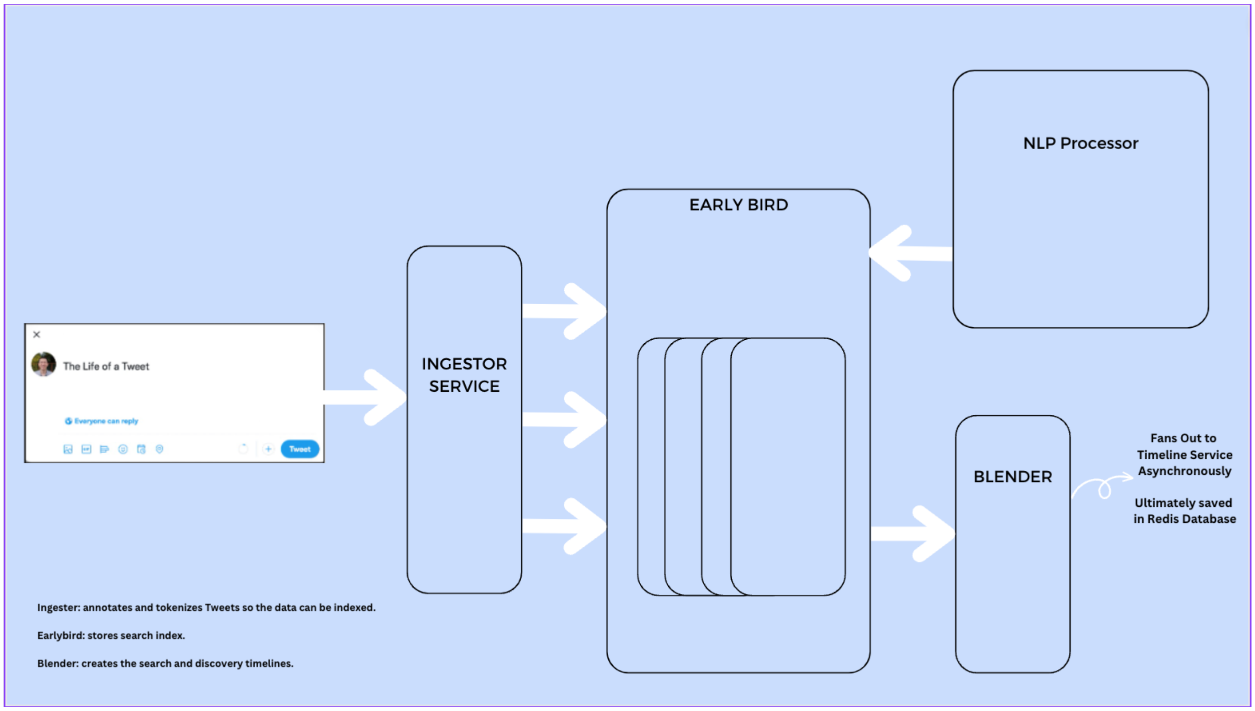
Source

Natural language processing with disaster tweets. Kaggle. (n.d.). from https://www.kaggle.com/competitions/nlp-getting-started/overview

Code Output

IMT 574 Group 3 Final Project Code Notebook

Figure 1 - Tweet Prediction Architecture Model



Ingester: annotates and tokenizes Tweets so the data can be indexed.

Earlybird: stores search index.

Blender: creates the search and discovery timelines.

References

Bandyopadhyay, H. (2023, March 2). Data Cleaning in Machine Learning: Steps & Process [2023]. Retrieved March 12, 2023, from https://www.v7labs.com/blog/data-cleaning-guide

Fleming, E. (n.d.). *Out of the Sea; Identifying Critical Tweets During Disasters*. Identifying Critical Tweets During Disasters. Retrieved March 12, 2023, from http://www.eamonfleming.com/projects/disaster-tweets.html

Howard, J. (2022, May 16). *Getting started with NLP for Absolute Beginners*. Kaggle. Retrieved March 12, 2023, from https://www.kaggle.com/code/jhoward/getting-started-with-nlp-for-absolute-beginners

IQVIA. (n.d.). *How does natural language processing (NLP) work?* Linguamatics. Retrieved March 12, 2023, from https://www.linguamatics.com/how-does-nlp-work

Mazereeuw, M., Quintero, A., & Barve, A. (n.d.). *Real-time flood mapping for Disaster Management Decision Support*. MIT Tata Center. Retrieved March 12, 2023, from https://tatacenter.mit.edu/portfolio/real-time-flood-mapping-for-disaster-management-decision-support/

PyCoach, T. (2021, June 9). *7 NLP techniques you can easily implement with python*. Medium. Retrieved March 12, 2023, from https://towardsdatascience.com/7-nlp-techniques-you-can-easily-implement-with-python-dc0ade1a53c2