Emotions-Based Disaster Tweets Classification: Real or Fake

MARCO ALFONSE[1,2], MARIAM GAWICH[2]
[1]Computer Science Department, Faculty of Computer and Information Sciences,
Ain Shams University,
Abbaseya, Cairo,
EGYPT

[2]Laboratoire Interdisciplinaire de l'Université Française d'Égypte (UFEID LAB),
Université Française d'Égypte,
21 Ismailia Desert Road Ville Shorouk, Cairo,
EGYPT

Abstract: - Social media platforms are considered interactive communication channels between governments, civil society organizations, and the public. During disaster occurrences, social media platforms play a crucial role such as the alertness of people towards the disaster occurrence, its risks, and consequences. They are used as tools to spread real updated information rapidly related to the disaster. Furthermore, social media platforms can facilitate the mobilization of volunteers as well as the organization of campaign donations after the disaster occurrence. Nevertheless, the benefits of social media platforms can be a double-edged sword through the dissemination of unreal information such as rumors or fake disasters. Unfortunately, the public can easily believe unreal information due to the anxiety that they experienced during the occurrence of a past real disaster. This paper presents a model to distinguish between the fake disaster tweets and the real ones. The implementation of this model is established twice; the first implementation involves the use of Machine Learning with the traditional Natural Language Processing techniques on the disaster dataset provided by Kaggle, and the second implementation involves using the emotions that are extracted from the tweets in the classification process. The proposed model achieves an accuracy of 88,34% without the usage of the emotion extraction module while it achieves an accuracy of 89,39 % with the inclusion of the emotion extraction module.

Key-Words: - Artificial Intelligence, Machine Learning, Natural Language Processing, Knowledge Discovery, Sentiment Analysis, Fake Disasters Tweets.

Received: July 28, 2022. Revised: August 29, 2023. Accepted: October 1, 2023. Published: October 18, 2023.

1 Introduction

According to the United Nations Office for Disaster Risk Reduction (UNDRR), [1], a disaster is defined as a severe disturbance of a society or a community during a dangerous phenomena occurrence. The dangerous phenomenon can provoke at least one of the following consequences: human life, loss, economic downturn, environmental loss, and material loss. Without any doubt, social media platforms such as Facebook and Twitter play a crucial role during disaster occurrence and disaster management after its occurrence due to their ability to speed up related information dissemination. These platforms, [2], can provide users with disaster awareness before or during its occurrence. They can facilitate the collection of financial support and raise awareness about the need for donations. Furthermore, the use of social platforms helps people discover their relatives' status and locations during the disaster or after the disaster occurrence. On the other hand, social media platforms can have a negative influence on society due to the dissemination of false information, [3]. Social media platforms allow any user to create and share information in terms of tweet messages or posts, regardless of their validation. The other users can believe the false information and share it because they can't distinguish whether the information is true or fake. Fake disaster tweets can have a negative impact on humans, the government, and the economy. They can affect the health of humans due to the increase in anxiety that can provoke human life loss, especially for those who suffer from diabetes and other harmful diseases. The dissemination of fake disaster tweets can provoke panic among investors, which may lead to the

destruction of their businesses. Furthermore, fake disaster tweets can have a negative influence on public trust in the government. Examples of popular fake tweets disseminated on social media platforms are fake news concerning COVID-19 remedies, [4], and the spread of false tweets concerning a pizza shop that led to a shooting incident in the USA, [5].
To eliminate the dissemination of fake disaster news on social media platforms, tweets, and Facebook posts should be analyzed and verified automatically. For this reason, Machine Learning (ML) and Natural Language Processing (NLP) techniques, [2], [6], [7], [8], are applied to classify whether the tweets involve a true disaster occurrence or not. The verification and analysis of the tweets face many challenges, such as the interpretation of the informal language and its transformation into formal language, the interpretation of slang terms, the emoticons, etc. Another challenge encountered in tweet analysis is sentiment extraction and its classification. Most of the researchers ignored the introduction of sentiment analysis of tweets and its influence on the performance of the classification results. They only applied ML and NLP techniques to tackle the disaster tweet classification.

The objective of this research is to highlight the role of sentiment analysis in improving the performance of ML algorithms to differentiate between fake and real disaster tweets. For this reason, a model was proposed for disaster tweet classification based on ML, NLP, and sentiment analysis. This model has many benefits, it can prevent the spread of false information during the disaster occurrence, and the model can facilitate the support provided to individuals suffering in the face of disasters by providing accurate information to the humanitarian movements, in addition to upholding human dignity during disasters.

This paper is organized as follows: section 2 presents the related work; section 3 demonstrates the proposed model for disaster classification; and section 4 presents the results and discussion. Section 5 contains the conclusion.

2 Related Work

In, [9], the author provided a classification model to identify which tweets are real and which are fake. The proposed model has several stages including, preprocessing the input data using many techniques such as count vector, Term Frequency Inverse Document Frequency (TF-IDF), a continuous bag of words, and a skip-gram vector. The author built a classifier network based on BERT that includes six layers. In this network, he modified some of the hyperparameters of the BERT to minimize the loss. These parameters include random state split, dropout and learning rates, batch size, and finally the optimizer. The data set used in this experiment is given by Kaggle which contains 7613 records used for training and 3263 records used for testing. It is a binary dataset that has two labels, not disaster and disaster. The author applied some of the data cleansing steps for each tweet such as normalization and removing all emails, URLs, HTML tags, emoticons, abbreviations, stop-words, special characters, and punctuation. The author compared his classification performance with some of the traditional machine learning techniques, and he proved that the BERT-based model outperforms other techniques having an overall F1-score equal to 0.8867. In, [10], the authors proposed a classification model that can identify the real tweets from the fake ones based on BERT (Bidirectional Encoder Representations from Transformers). The authors added a dropout layer and another dense layer with a ReLU (rectified linear unit) activation function to the utilized BERT model. The authors used a dataset downloaded from Kaggle that contains 10873 comments, of which 57.03% are not real disasters. Each tweet has an identifier, the text of the tweet, location, keywords, and target. The cleaning of the data is done by removing URLs, HTML tags, special characters, duplicates, special characters, and emails. The data are converted into vectors using TF-IDF, and the linear SVC is utilized as a classifier. The validation accuracy obtained by the proposed model is 79%.

In, [11], the authors propose a methodology that depends on machine learning methods to classify disaster tweets. They proposed two classifiers, which are support vector machine and naive bayes, for their classification system as these two algorithms are frequently used for tweet classification, according to their literature review. They also proposed the inclusion of emoticons in the classification system for better recognition of disaster tweets. They proposed extracting the tweet using an API named Twitter Streaming API. Preprocessing tweets is to be applied to remove noise, repetition, and any unwanted elements from the collected tweets. The authors proposed using the LSTM recurrent neural network to consider emoticons in their classification system, as they mentioned that most models usually remove emoticons from the data in their classification systems. The authors mentioned that they are going to implement this model and compare the results of the proposed algorithms.

In, [12], the authors provided a model for the analysis of disaster tweets. The author proposed a model to identify the informative tweet from the other tweets. Their methodology is based on the following stages: preprocessing, feature extraction, and classification. The preprocessing involves converting all tweets to lowercase letters, removing hashtags, punctuation, URLs, digits, and stop words, removing words that are of length 2, and making all tweets the same length by padding techniques. The feature extraction is performed through the embedding of words using the TF-IDF technique. The authors used the linear SVC as a classification algorithm. They evaluated their method using the F1 score metric, which gives 0.72.

In, [13], the authors applied many machine learning and deep learning techniques to categorize disaster tweets. They applied their techniques to a data set related to cyclone AMPHAN and NISARGA, and they collected these tweets using Tweepy. They applied different data preprocessing methods including the removal of hashtags, mentions, and URLs, replacing each line and tab break with a space, converting each emoticon into positive and negative, and finally applying lemmatization. For the feature extraction techniques, they applied the glove embedding with the deep learning techniques and count vectorizer, TF-IDF, n-gram, word, and character level for the machine learning techniques. They applied many classifiers, including a classifier based on BERT, bi-directional LSTM, TextCNN, SVC, XG-Boost, logistic regression, SGD, linear SVC, random forest, KNN, AdaBoost, decision tree and Gaussian Naive Bayes. They achieved accuracies that range from 0.51 to 0.72 while categorizing the data into five categories; ‘Important Help Related’, ‘Informative’, ‘Damage and Casualty related’, ‘Emotional’, and ‘Irrelevant’. On the other hand, they achieved accuracies that range from 0.56 to 0.8 while categorizing the data into four categories, ‘Important for Disaster Managers’, ‘Important for Public’, ‘Important for Both’, and ‘Others’. The authors showed that the BERT-based classifier outperforms machine learning and deep learning techniques in both cases.

3 Proposed Model For Tweet Analysis

The idea behind the proposed model of disaster tweet classification, which indicates whether the disaster is real or not, is based on the sentiment analysis (polarity) of each tweet.

The proposed system involves five phases. The first phase is dataset selection, where a benchmark dataset is selected to examine the proposed model. The second phase is dataset preprocessing, which is necessary for cleaning and preparing the data for the next phases as the quality of the data has a great impact on the classification performance. The third phase is emotions extraction, where the dataset is enriched by a new feature extracted from the tweets. This newly added feature enhances the performance of our proposed model. The fourth phase is feature selection, where the input features are converted into a format that can be understood by the applied classifier in the next phase. The final phase is classification, where the input records are given a label by a machine learning classifier, either real or fake. Figure 1 demonstrates the five phases of the proposed model. The model is implemented using Python Language, [14], and the R program, [15].

3.1 Data Selection

The dataset is provided by, [16]. It involves the following features: the id which is a unique identifier for the tweet, the keyword that describes the disaster, the location where the tweet was posted, the text (the tweet itself) and the target (decision class label) that determines whether the disaster is real or not. The dataset contains 11371 records.

3.2 Data Preprocessing

The preprocessing of the Tweet disaster dataset is established through the removal of records that have no values (NaN values) using the Panda library, [17]. The disaster dataset contains 7953 records after the removal of the missing value. Preprocessing also involves noun phrase detection using the TextBlob library, [18], word lemmatization using WordnetLemmatizer, [19], and applying regular expressions to clean the dataset using the RegEx library, [20]. All the previous text processing operations are applied to the Tweet text. Therefore, each Tweet text provided in the disaster dataset is processed and added as a new feature to the dataset.

3.3 Emotions Extraction

In this stage, the newly added feature to the dataset is analyzed using the syuzhet package, [21], of the R program to identify the emotions in the text. These emotions include fear, anticipation, joy, anger, sadness, disgust, surprise, and trust. The Syuzhet package assigns a score for the positivity or negativity of the processed tweet.

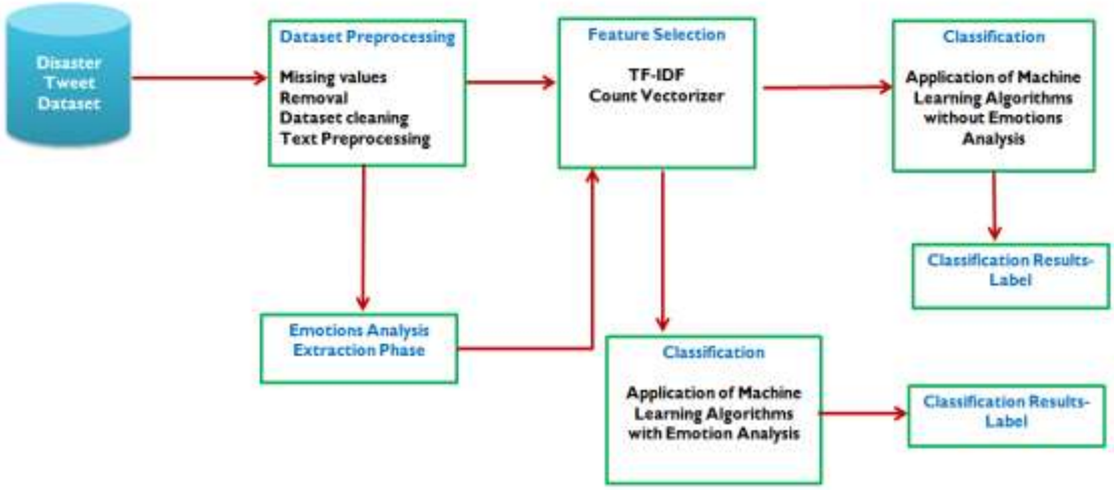

Fig. 1: The Proposed Model for Tweets Disasters Classification

Each Tweet is assigned a score of positivity and negativity. If the positive score of a tweet is greater than the negativity score, the polarity of the tweet will be positive, and vice versa. If the tweet has a positive score that is equal to a negative score, its polarity will be neutral. Furthermore, the polarity will be added to the initial dataset and considered as a new feature called “tweet polarity” that will be taken into consideration in the application of the machine learning algorithms.

Consider the following tweet concerning an accident that occurred in Covina as an example to demonstrate how the syuzhet package works: "How could you leave when I gave you my all □" The syuzhet assigns the following score for each one of the eight emotions: 0 for anger, 0 for anticipation, 0 for disgust, 0 for fear, 0 for joy, 1 for sadness, 1 for surprise, and 0 for trust. The syuzhet assigns 1 for the negativity of this tweet and 0 for its positivity. Consequently, the sentiment analysis result of this tweet is negative. Table 1 summarizes the results of the emotion extraction phase.

Table 1. The experimental Results of the Emotion Extraction Phase

Tweet Polarity	Number of tweet texts
Positive	2030
Negative	3295
Neutral	2628

The sentiment or emotion analysis of the processed tweets demonstrates that 25% of tweets are positive, 41% of tweets are considered negative tweets, and 33% of tweets are neutral. The disaster dataset is populated by the new feature “tweet polarity “generated by the Syuzhet package using the R program.

3.4 Features Selection

This stage is applied twice, as we performed two experiments. In the first experiment, the tweet represents the independent variable, and the target column is used as the output class. Concerning the second experiment, the tweet and its polarity that were detected in the previous phase are taken as independent variables while the target column is used as the output class. For both experiments, two feature extraction techniques are applied: the TF-IDF vectorizer, [22], and the Count Vectorizer, [23]. The TF-IDF is used to determine the term relevance and its occurrence frequency in the tweets. The Count Vectorizer is applied to switch the tweets and their polarity to numeric values to be processed by the classification algorithms.

3.5 Classification

For the two experiments we performed, several supervised machine-learning algorithms were applied to the disaster dataset. The applied supervised machine learning algorithms are: Multinomial Naive Bayes (MNB), Logistic Regression (LR), Linear Support Vector Classification (SVC), K Nearest Neighbors (KNN), Decision Tree (DT), Random Forest (RF), Gradient Boosting (GB), Stochastic Gradient Descent (SGD), Multi-Layer Perceptron (MLP), and Bagging Classifier (BC). The objective of this phase is to identify the new tweet as a fake disaster tweet or a real disaster one.

3.5.1 Multinomial Naive Bayes (MNB)
It is a type of the Naïve Bayesian algorithm, [24], that is widely applied for text classification. The MNB estimates the likelihood of the word occurrence with a class, regardless of the word position in the text. The Laplace smoothing technique is employed by the applied classifier to prevent division by zero while calculating probabilities. Also, it presumes that the prior probability distribution for the data is uniform.

3.5.2 Logistic Regression (LR)
It comes from the statistics domain. Its objective, [25], is to discover the link between the output (class) and the numerical values (input). It involves the use of the sigmoid function to determine the class label. Additionally, it implies the use of the L2 regularization technique, which makes the weights down towards zero to prevent the over-fitting issue.

3.5.3 Linear Support Vector Classification (SVC)
It originated from the Support Vector Machine algorithm; it is suitable to be applied for multi-classification or single classification. The SVC, [26], generates the hyperplanes through an optimization method. A hyperplane is a decision boundary that splits the input data according to the classes. The support vectors are the data points that are closest to the hyperplane. The SVC implies the use of kernels to easily find the hyperplane, which can separate different classes.

3.5.4 K Nearest Neighbors (KNN)
Being a non-parametric technique, the KNN, [27], does not require any restrictions on the distribution of the data. The classification is established by computing the distance (the Euclidean distance) between the test datum input and the training datum output while taking the customized K value into account. The number of neighbors (K) used by the applied KNN classifier is three. All records in the neighborhood are equally weighted.

3.5.5 Decision Tree (DT)
It is a non-parametric classifier, [28], that doesn't rely on a mathematical model. It is widely applied in several domains such as loan approval and disease diagnosis. Its concept is based on the modeling of a tree that represents the data structure used to classify new cases. In the decision tree, the case is stated in the matter of features/attributes that can be textual or numerical values. According to the decision rules that are inferred from the feature values, the algorithm can predict the output (the decision class).

3.5.6 Random Forest
The random forest algorithm, [29], [30], can be used for classification and regression. Its main idea is to build many decision trees that can be trained on samples of the dataset and find the class label based on the majority of the decisions of these trees. Training multiple trees prevents the model from over-fitting and enhances the classification accuracy. The applied algorithm utilizes 10 different trees and applies the Gini measure to determine the split quality.

3.5.7 Gradient Boosting (GB)
This classifier, [31], utilizes some models (weak models) to produce a strong one. Each weak model is a one-split-point decision tree. During training, the decision tree that produces the minimal error is added to the other weak models. The applied algorithm utilizes 100 boosting stages which makes sure that there is no overfitting. The quality of the split is measured using the mean squared error proposed by Friedman.

3.5.8 Stochastic Gradient Descent (SGD)
It is an optimization algorithm, [32], where each iteration uses a single training data sample, and the weights are adjusted based on the gradient descent value. The SGD runs multiple times until it minimizes the loss function as much as possible. The applied model uses SGD learning to create a regularized linear model (linear Support Vector Machine). Each record's gradient of the loss is computed, and the model updating is executed at a decreasing learning rate.

3.5.9 Multi-Layer Perceptron (MLP)
This algorithm, [33], is a type of artificial neural network that can be used for classification and involves at least three layers: one for input, one is hidden, and the last one is the output layer. The hidden layers can be one or more. This network is trained for many epochs until the error rate reaches an acceptable value. The applied algorithm contains 2 hidden layers and each hidden layer has 10 neurons. The applied activation function is ReLU and the value of alpha is 0.0001. The batch size is 200 and the learning rate is 0.001. The algorithm is trained for 200 epochs.

3.5.10 Bagging Classifier (BG)
It is an ensemble classifier, [34], [35], where samples of the data set are used to train the model in a random order, and the final output is determined by the mean value of the whole data set. This classifier uses a base estimator such as a Support

Vector Machine or K-nearest neighbors to predict the suitable class label. The applied algorithm uses the k nearest neighbor algorithm where the distance is calculated using the Euclidean distance.

4 Results and Discussion

The implementation of the proposed model is performed using the Python language through the Google Colab, [36], and the use of the R program. The data preprocessing phase, feature extraction phase as well and machine learning phase are implemented through Colab whereas the R program is used to implement the emotion extraction phase. The disaster dataset is split into 70% training and 30% testing. The proposed model has been implemented twice. The first implementation involves the data preprocessing phase, the feature selection phase, and the classification phase. The second implementation is based on the data preprocessing phase, the feature selection phase, the emotion analysis phase, and the classification phase. Table 2 demonstrates the results of the classification without the inclusion of the emotion analysis phase. Table 3 demonstrates the results of the classification with the inclusion of the emotion extraction phase.

Table 2. The Model Performance Without Using Emotion during the Classification Process

Classifier	Accuracy	Performance			
		Class	Precision	Recall	F1-Score
MNB	0.8830678960603521	0	0.97	0.89	0.93
		1	0.54	0.81	0.65
LR	0.8834870075440067	0	0.97	0.90	0.93
		1	0.55	0.81	0.65
SVC	0.8696563285834031	0	0.94	0.90	0.92
		1	0.57	0.72	0.64
KNN (K is 3)	0.8310980720871752	0	0.99	0.83	0.90
		1	0.20	0.81	0.32
DT	0.8461860854987426	0	0.93	0.89	0.91
		1	0.53	0.64	0.58
RF	0.8725901089689857	0	0.99	0.87	0.93
		1	0.39	0.93	0.55
GB	0.8461860854987426	0	0.99	0.84	0.91
		1	0.26	0.88	0.41
SGD	0.8654652137468567	0	0.94	0.90	0.92
		1	0.59	0.69	0.63
MLP	0.8616932103939648	0	0.93	0.90	0.91
		1	0.60	0.67	0.63
BC	0.8088851634534786	0	1.00	0.81	0.89
		1	0.04	1.00	0.08

Concerning the first implementation of the proposed model, the classification takes into consideration the processed tweets and the target (decision class) without applying the emotion extraction phase. The LR achieves the highest accuracy of 88.34%, which is almost the same as the Multinomial Naïve Bayes (MNB) algorithm with an accuracy of 88.30%. Furthermore, the linear Support Vector Classification (SVC) achieves accuracy above 85%, which is better than other classifiers.

Table 3. The Model Performance With The Usage Of The Emotion Analysis During The Classification Process

Classifier	Accuracy	Performance			
		Class	Precision	Recall	F1-Score
MNB	0.8822296730930428	0	0.97	0.90	0.93
		1	0.51	0.78	0.61
LR	0.893964794635373	0	0.97	0.91	0.94
		1	0.56	0.80	0.66
SVC	0.8818105616093881	0	0.94	0.92	0.93
		1	0.61	0.71	0.66
KNN (K is 3)	0.8445096395641241	0	0.99	0.84	0.91
		1	0.19	0.86	0.31
DT	0.8524727577535625	0	0.93	0.89	0.91
		1	0.51	0.62	0.56
RF	0.8755238893545683	0	0.99	0.87	0.93
		1	0.36	0.92	0.51
GB	0.8608549874266554	0	0.99	0.86	0.92
		1	0.28	0.89	0.43
SGD	0.8763621123218777	0	0.94	0.91	0.93
		1	0.58	0.70	0.63
MLP	0.863788767812238	0	0.92	0.92	0.92
		1	0.63	0.63	0.63
BC	0.8243922883487007	0	1.00	0.82	0.90
		1	0.05	1.00	0.09

Concerning the second implementation of the proposed model, the classification takes into consideration the processed tweets, tweet polarity (a new feature), and the target (the decision class). The accuracy of most of the applied classifiers with the inclusion of the emotion extraction phase is enhanced compared to the first implementation results. The increased accuracy scores of the SVC, Gradient Boosting (GB), and Bagging Classifier (BC) reflect the impact of emotion extraction on the performance of the classification. The LR achieves the highest accuracy, 89.39%. In addition, the MNB and the SVC have achieved an approximately similar accuracy rate, which is 88%.
Table 4 presents a comparison between our proposed model and the available literature.

Table 4. A Comparison between the Proposed Model and Other Models

Model	Methodology	Performance
[9]	BERT	F1-score=88%
[10]	BERT	Accuracy=79%
[11]	Support Vector Machine, Naive Bayes, and LSTM	No Implementation
[12]	linear SVC	F1-score=0.72
[13]	BERT	Accuracy=72%
The proposed model	NLP+ Emotion Analysis + Machine Learning	Accuracy=89,39%

As shown in Table 4, the overall performance of the applied classifiers on the disaster dataset is better than the previous research models, as including the emotion analysis phase in addition to applying the NLP techniques to the data in the preprocessing phase has a salient positive impact on the performance of the proposed model.

5 Conclusion

The disaster tweets can have a great impact on the decision-making of many people and organizations. Therefore, it is necessary to have an automated model that can identify the real disaster tweets from the fake ones. Most existing classification models don't take into consideration the emotional analysis in the classification process. This paper provides a new classification model that considers emotion analysis during the classification process. What distinguishes the proposed model from other models is the introduction of an emotion extraction phase that gives the ability to discover new knowledge from the tweets. Furthermore, the enrichment of the disaster dataset with the new feature (tweet polarity) has a positive impact on the performance of classifiers. The model achieves an accuracy rate of 89.39%, which is greater than the accuracy of existing models. The proposed model doesn't consider the processing of the emotions' icons (emoticons), which may lead to lower performance if the model is provided with a tweet that contains a lot of emoticons. For this reason, in future work, the authors are going to include another phase to handle the emoticons, which can result higher accuracy score.

References:

[1] United Nations Office for Disaster Risk Reduction (UNDRR), https://www.undrr.org/terminology/disaster (Accessed Date: Jul. 3 2023).

[2] V. Linardos, M. Drakaki, P. Tzionas, and Y. Karnavas, "Machine learning in disaster management: Recent developments in methods and applications," Machine Learning and Knowledge Extraction, vol. 4, no. 2, pp. 446–473, 2022. doi:10.3390/make4020020

[3] S. Madichetty, S. Muthukumarasamy, and P. Jayadev, "Multi-modal classification of Twitter data during disasters for humanitarian response," Journal of Ambient Intelligence and Humanized Computing, vol. 12, no. 11, pp. 10223–10237, 2021. doi:10.1007/s12652-020-02791-5

[4] M. N. Alenezi and Z. M. Alqenaei, "Machine learning in detecting COVID-19 misinformation on Twitter," Future Internet, vol. 13, no. 10, p. 244, 2021. doi:10.3390/fi13100244

[5] C. Kang and A. Goldman, "In Washington Pizzeria attack, fake news brought real guns," The New York Times, https://www.nytimes.com/2016/12/05/business/media/comet-ping-pong-pizza-shooting-fake-news-consequences.html (Accessed Date: July 9 2023).

[6] A. Kruspe, J. Kersten, and F. Klan, "Review article: Detection of actionable tweets in crisis events," Natural Hazards and Earth System Sciences, vol. 21, no. 6, pp. 1825-1845, 2021. doi:10.5194/nhess-21-1825-2021

[7] M. Zhang, "Applications of deep learning in news text classification," Scientific Programming, vol. 2021, pp. 1–9, 2021. doi:10.1155/2021/6095354

[8] D. Gupta, N. Narang, M. m, and M. Sood, "Disaster tweets classification," SSRN Electronic Journal, 2022. doi:10.2139/ssrn.4020410

[9] A. D.Le."Disaster Tweets Classification using BERT-Based Language Model". arXiv preprint arXiv:2202.00795. 2022

[10] A. K. Ningsih and A. I. Hadiana, "Disaster tweets classification in disaster response using bidirectional encoder representations from transformer (Bert)," IOP Conference Series: Materials Science and Engineering, vol. 1115, no. 1, p. 012032, 2021. doi:10.1088/1757-899x/1115/1/012032

[11] K. Asinthara, M. Jayan, and L. Jacob, "Classification of disaster tweets using machine learning and Deep Learning Techniques," 2022 International Conference on Trends in Quantum Computing and Emerging Business Technologies (TQCEBT),

2022. doi:10.1109/tqcebt54229.2022.10041629

[12] A. Kumar and S. Reddy, "A detailed analysis on disaster tweet analysis using Deep Learning Techniques: DTWEET," 2022 Fourth International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT), 2022. doi:10.1109/icerect56837.2022.10060783

[13] S. Sharma, S. Basu, N. K. Kushwaha, A. N. Kumar, and P. K. Dalela, "Categorizing disaster tweets into actionable classes for disaster managers: An empirical analysis on Cyclone Data," 2021 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME), 2021. doi:10.1109/iceccme52200.2021.9591063

[14] Python Language- "Welcome to Python.org", Python.org, 2022. [Online]. Available: https://www.python.org/. (Accessed Date: 22 July 2023).

[15] R Program - "The R project for statistical computing," R, https://www.r-project.org/ (Accessed Date: 24 July 2023).

[16] Disaster tweets- Kaggle, https://www.kaggle.com/datasets/vstepanenko/disaster-tweets (Accessed Date: 22 July 2023).

[17] Pandas - Python Data Analysis Library. Pandas.pydata.org. (2022). (Retrieved on: 24 July 2023), from https://pandas.pydata.org/.

[18] S. Loria, P. Keen, M. Honniba, R. Yankovsky, D. Karesh, E. Dempsey, "Textblob: simplified text processing.", Secondary TextBlob: simplified text processing, 3, 2014.

[19] WordnetLemmatizer, NLTK, https://www.nltk.org/_modules/nltk/stem/wordnet.html (Accessed Date: 24 July 2023).

[20] J. Friedl, Mastering Regular Expressions, 3rd Edition. [S.l.]: O'Reilly Media, Inc., 2006.

[21] M. Jockers, "Extracts Sentiment and Sentiment-Derived Plot Arcs from Text [R package syuzhet version 1.0.6]", Cran.r-project.org, 2022. [Online]. Available: https://cran.r-project.org/web/packages/syuzhet/index.html. (Accessed Date: 24 July 2023).

[22] S. Tongman & N. Wattanakitrungroj, "Classifying Positive or Negative Text Using Features Based on Opinion Words and Term Frequency-Inverse Document Frequency". In 2018 5th International Conference on Advanced Informatics: Concept Theory and Applications (ICAICTA), pp. 159-164, IEEE,2018.

[23] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R.Weiss, V.Dubourg, J.Vanderplas, A.Passos, & D.Cournapeau, "Scikit-learn: Machine learning in Python", the Journal of machine Learning research, vol. 12, pp.2825-2830, 2011

[24] A. M Kibriya, E. Frank, B. Pfahringer & G. Holmes, 2004," Multinomial naive bayes for text categorization revisited". In Australasian Joint Conference on Artificial Intelligence, pp. 488-499, Springer, Berlin, Heidelberg, 2004.

[25] J. Brownlee," Master Machine Learning Algorithms: discover how they work and implement them from scratch", Machine Learning Mastery. Chapter 13: Logistic Regression, pp.51-55, 2016a.

[26] J. Novakovic, & A. Veljović, "C-support vector classification: Selection of kernel and parameters in medical diagnosis". In 2011 IEEE 9th International Symposium on Intelligent Systems and Informatics, pp. 465-470, 2011.

[27] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov," Neighbourhood components analysis". In Advances in neural information processing systems, pp. 513-520, 2005.

[28] T. Hastie, T. Robert, and F. Jerome," The elements of statistical learning: data mining, inference, and prediction". Springer Science & Business Media, 2009.

[29] J. Brownlee," Master Machine Learning Algorithms: discover how they work and implement them from scratch", Machine Learning Mastery. Chapter 28: Bagging and Random Forest, pp.126-129, 2016b

[30] G. Biau, 2012. Analysis of a random forests model. The Journal of Machine Learning Research, vol 13, no.1, pp.1063-1095,2012.

[31] J. Friedman, "Greedy function approximation: A gradient boosting machine.", The Annals of Statistics, vol. 29, no. 5, pp. 1189-1232, 2001.

[32] L. Bottou," Large-scale machine learning with stochastic gradient descent", In Proceedings of 19th International Conference on Computational Statistics (COMPSTAT'2010), pp. 177-186, Physica-Verlag HD, 2010.

[33] H. Ramchoun, M. Amine, J. Idrissi, Y. Ghanou and M. Ettaouil, "Multilayer Perceptron: Architecture Optimization and Training", International Journal of Interactive Multimedia and Artificial Intelligence, vol. 4,

no. 1, pp. 26-30, 2016. Available: 10.9781/ijimai.2016.415.
[34] J. Brownlee, Master Machine Learning Algorithms: discover how they work and implement them from scratch. Machine Learning Mastery. Chapter 28, pp.26-127, 2016c.
[35] L. Breiman, "Bagging predictors", Machine Learning, vol. 24, no. 2, pp. 123-140, 1996. Available: 10.1007/bf00058655.
[36] Google Colab, https://colab.research.google.com/ (Accessed Date: 24 July 2023).

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

The authors equally contributed to the present research, at all stages from the formulation of the problem to the final findings and solution.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

No funding was received for conducting this study.

Conflict of Interest

The authors have no conflicts of interest to declare.