Published by
NIGERIAN SOCIETY OF PHYSICAL SCIENCES
Available online @ https://journal.nsps.org.ng/index.php/jnsps

J. Nig. Soc. Phys. Sci. 7 (2025) 2675

Journal of the Nigerian Society of Physical Sciences

Effective tweets classification for disaster crisis based on ensemble of classifiers

Christopher Ifeanyi Eke[a,*], Kholoud Maswadi[b], Musa Phiri[c], Mulenga Mwege[c], Mohammad Imran[d], Dekera Kenneth Kwaghtyo[a], Akeremale Olusola Collins[e]

[a]*Department of Computer Science, Faculty of Computing, Federal University of Lafia, P.M.B 146, Lafia, Nasarawa State, Nigeria*
[b]*Department of Management Information Systems, Jazan University, Jazan 45142, Saudi Arabia*
[c]*School of Engineering and Technology, Mulungushi University, PO Box 80415, Kabwe, Zambia*
[d]*Department of Information Technology, Balochistan University of Information Technology, Engineering and Management Sciences, Airport Road, Baleli, Quetta, Pakistan*
[e]*Department of Mathematics, Faculty of Science, Federal University of Lafia, P.M.B 146, Lafia, Nasarawa State, Nigeria*

Abstract

In the field of disaster management, social media analytics has gained significant recognition. Social media platforms, particularly Twitter, have become an invaluable source for disseminating information during disasters, offering real-time updates on events, crisis reports, and casualty information. However, the deluge of information on social media can also be overwhelming, with a substantial amount of irrelevant content. To address this challenge, researchers leverage machine learning (ML) classifiers to automatically categorize disaster-related tweets. However, ML classifiers, while being effective, also face issues such as overfitting and class imbalance. This study proposes an ensemble-based approach that integrates a variety of linguistic and word embedding features, including Parts-Of-Speech (POS), hashtags, Term Frequency-Inverse Document Frequency (TF-IDF), GloVe, Word2Vec, and BERT. A range of supervised learning algorithms like Decision Trees, Logistic Regression, Support Vector Machines, and Random Forests, were evaluated individually and as part of ensemble methods like AdaBoost, Bagging, and Random Subspace. The results show that combining TF-IDF with word embeddings and using the AdaBoost ensemble model yields superior performance, achieving a classification accuracy of 98.92%. This represents a notable improvement over the conventional standalone classifiers and highlights the advantage of ensemble methods in enhancing model robustness and minimizing overfitting. The proposed approach demonstrates not only high predictive capacity but also scalability for real-time tweet filtering during emergencies. In addition to demonstrating the efficacy of ensemble methods in disaster tweet classification, this study also provides valuable insights for improving social media-based crisis response. It also establishes a foundation for future research, particularly in multi-lingual and multi-disaster scenarios.

DOI:10.46481/jnsps.2025.2675

Keywords: Disaster crisis management, Social media analytics, Twitter, Machine learning classifiers, Ensemble methods, Feature extraction

Article History :
Received: 11 February 2025
Received in revised form: 28 April 2025
Accepted for publication: 30 April 2025
Available online: 26 May 2025

Communicated by: O. Akande

1. Introduction

In recent years, disaster crisis management researchers and emergency practitioners have widely acknowledged the significance of social media analytics [1]. Times of disaster are often characterized by large volumes of message exchange between

*Corresponding author Tel. No.: +234-706-809-0013.
Email address: eke.christopher@science.fulafia.edu.ng (Christopher Ifeanyi Eke)

friends and families trying to notify one another about the ongoing developments. Consequently, this leads to inefficiencies in standard communication technologies due to the significant overload of network lines, which in turn, limits the effectiveness of disaster response teams. Recently, social media has drawn a lot of attention in disaster crisis response [2]. Not only is it making it easier for people to share information, but the process of data collection and analysis has also been simplified by many social media platforms such as Twitter [3].

Twitter has become well known among various organizations and individuals for its value in improving situational awareness during times of disaster crisis [4, 5]. It enables people to post real-time and on-topic information about their status, reports on damage to infrastructure, and information about injured people or loss of life [6]. Disaster crisis-related tweets have proven particularly effective in many human-made or naturally occurring crises such as floods [7], earthquakes [8], nuclear disasters [9] and wildfires [10]. However, while tweets provide important information, they often contain a significant amount of unrelated or irrelevant information. As a result, social media analysts and disaster crisis response teams are faced with the challenge of coming up with mechanisms to prioritize relevant posts while discarding irrelevant ones. This has led to an increase in intelligent technologies for the automatic classification of tweets.

Lately, conventional machine learning (ML) classifiers have gained much popularity in research related to categorizing disaster crisis-related tweets [11]. Conventional ML classifiers have shown remarkable performance in the identification of trustworthy and relevant information related to disaster crisis posts [12]. These classifiers have recorded superior performance results in several areas such as text classification, natural language processing (NLP), speech recognition, and object detection [13, 14]. The main idea of ML classifiers is to learn underlying patterns and make predictions based on historical data. ML classifiers such as k-nearest neighbour (KNN), support vector machines (SVM), Naïve Bayes, and logistic regression (LR) have successfully been applied in the identification of disaster crisis-related tweets [15]. However, single-based ML classifiers are prone to several challenges such as overfitting, especially when the amount of available data is small [16]. Also, ML classifiers are affected by significant class imbalances, where a classifier gives more preference to a class with significantly more examples than others.

Existing studies have shown that one way to enhance the performance of ML classifiers is through the use of ensemble methods [14, 17]. Ensemble methods refer to a learning approach that involves integrating and weighing several base-ML models to identify a classifier that outperforms the rest [18]. The general concept of the ensemble-based approach is to maximize the predictive performance by combining the strengths of several base classifiers. In addition, research has also revealed that most text classification tasks have been handled using general-purpose NLP techniques such as Parts-Of-Speech (POS), n-grams, unigrams, and Global Vectors (GloVe) [19–21].

However, while prior studies have examined the use of machine learning for tweet classification during disasters, few have conducted a detailed comparative analysis of multiple feature representations including TF-IDF, word embeddings, and POS tags combined with ensemble classifiers. Furthermore, existing works often rely on a single model or feature type, which may not generalize well across diverse social media content.

This study addresses this gap by evaluating a range of features and classifiers, with a focus on the effectiveness of ensemble approaches. The study contributes to the field by identifying optimal combinations of features and models that can improve the robustness and accuracy of tweet classification in real-time disaster scenarios. The main contributions of this work are outlined as follows:

- The implementation of three different linguistic features and four-word embedding features for disaster crisis classification on a benchmark dataset.
- The experimentation with the use of ensemble learning methods in conjunction with ensemble feature subsets and classifiers.
- Performance comparison of feature ensemble and classifiers ensemble for disaster crises classification.
- Using the publicly available dataset, the study illustrates that the proposed ensemble-based approach can successfully classify disaster tweets when applied to real-world data.
- The systematic integration, evaluation, and optimization of feature engineering, ensemble learning, and classifier performance on real-world disaster-related tweets.

The rest of the study is organized in the following manner. The second section provides an overview of the related works. The third section outlines the materials and methods used to implement the proposed approach. In the fourth section, the results and discussion of the study are presented. Finally, the fifth section gives the conclusions and suggestions for future works.

2. Review of related works

This section provides a literature review based on feature engineering, conventional machine learning, deep learning, and ensemble classifiers for disaster crisis classification as represented in the subsection below.

2.1. Feature engineering

Feature engineering is used to extract relevant information from tweet data to improve the performance of classifiers. Neppalli *et al*. [22] first preprocessed the raw tweet data using Natural Language Processing (NLP) techniques, such as tokenization, stemming, and stop word removal to standardize the text and make it easier to analyze. Then the preprocessed tweets were used to extract a set of features that could be used to train and evaluate their classifiers. The combination of traditional

and more complex features to capture both basic and more nuanced aspects of tweet content helped to improve the model's accuracy. The paper by Schnebele *et al.* [23] demonstrates how feature engineering can be used to extract relevant information from limited remote sensing data and improve the accuracy of flood extent estimation during disaster events. By using a combination of pixel-based and object-based features, the authors can capture both basic and more nuanced aspects of the satellite imagery, which helps to improve the accuracy of flood extent estimation. In the paper by Naderi [24], performed feature engineering that includes tokenization, stop word removal, stemming, and the removal of URLs and mentions. Besides, several linguistic and content-based features, such as the number of words, characters, and hashtags, as well as sentiment, readability, and information content were extracted. The authors also used Named Entity Recognition (NER) to extract entities related to disasters, such as locations and event types, and then applied part-of-speech (POS) tagging to identify the function of the words in the tweet. Basu *et al.* [25] presented another way for utilizing Twitter data for the management of post-disaster resource needs and resource availability. Word-level and character-level embeddings were proposed in two separate unsupervised neural retrieval models. They examined several different unsupervised techniques, including pattern matching and information retrieval. Tweets from the 2015 Nepal and portions of India earthquakes, as well as the 2016 Central Italy earthquake, were used to compile the dataset. The unsupervised information retrieval approach proposed by the authors yielded better results compared to other methodologies applied to the dataset of the earthquake that happened in Nepal. The suggested method achieved an accuracy of 0.57 in classifying the Nepal earthquake data and an F1-score of 0.191.

Several research projects have used ML and NLP strategies for disaster management, particularly from the perspective of emergency rescue operations [26, 27]. By combining natural language processing techniques with the machine learning algorithms Nave Bayes and Maximum Entropy, Verma *et al.* [28] were able to identify which tweets contributed to situational awareness during emergencies. Three disasters related data were analyzed, including the 2009-10 floods at Red River, the earthquake in Haiti, in 2010, and the 2009 Oklahoma grass fire. To begin classifying tweets about situational awareness from the three crisis occurrences, they started by building two supervised classifiers, one using Naive Bayes and another using Maximum Entropy. They followed up by analyzing the classifiers' overall performance across all four instances. However, the authors discovered that the classifiers were transferable across the 2009 and 2010 Red River floods, but not between different types of disasters. For instance, the classifier trained on the Haiti earthquake data had poor accuracy performance, in the cases when it was applied to the data of grass fires in Oklahoma, and vice versa, due to the significant differences between the two types of occurrences.

In addition, traditional ML techniques have been used for tweet classification during disasters. For instance, Imran *et al.* [29] proposed a classification framework that combines several traditional machine learning algorithms, including Naive Bayes, Decision Trees, and Random Forest, for extracting information nuggets from tweets during disasters. While the novel combination of algorithms may lead to better performance than using a single algorithm, the study's sample size may not be large enough to provide conclusive results or generalizable findings. Kryvasheyeu *et al.* [30], proposed a DT and RF-based rapid assessment model for disaster damage using social media activity, including tweets. The strength of the proposed method has high precision and recall values, indicating the approach's effectiveness in identifying disaster-related messages and inferring damage. However, the authors relied heavily on the use of location information which may lead to bias in the results in a situation where users are less likely to share their location. Khare *et al.* [31] proposed a method that uses semantic and statistical features of tweets across several languages to classify crisis data. One strength of the paper is that it addresses the issue of language diversity in crisis data classification, which is an important challenge in the field. However, the authors do not provide a detailed evaluation of their method, such as a comparison with existing methods in the field.

Deep learning algorithms have also been extensively used to classify social media content related to crises. For instance, the paper by Burel *et al.* [32], proposes a method called SemCNN, which is a deep and wide CNN model that uses the conceptual semantics of words to detect information categories of tweets related to crises. The evaluation results in the paper indicate that the method performs better than other traditional machine learning approaches. However, the paper does not provide much insight into the generalizability of the approach to other languages or different types of crises. Similarly, Kabir and Madria [33] applied Convolutional Neural Networks for tweet classification and rescue scheduling for effective disaster management. While the paper proposes an interesting approach for tweet classification and rescue scheduling for disaster management, the lack of detailed evaluation and the narrow focus of the study limit its potential impact on real-world disaster management scenarios. Bhoi *et al.* [34] proposed a framework that involves the use of pre-processing, feature engineering, and deep learning techniques, including convolutional neural networks (CNNs) and long short-term memory (LSTM) networks, for social media text classification and sentiment analysis for disaster resource management. Although the proposed framework employs pre-processing and feature engineering techniques to improve the quality of the data, which is crucial for accurate analysis, the authors did not clearly explain the feature engineering techniques used in the framework. In another study, Kundu *et al.* [35] introduced an LSTM-based framework to categorize tweets into these categories (NGOs). The study relied on a dataset including information about the 2015 Nepal earthquake that was received from the International Information Retrieval Forum for the years 2015 and 2017. With an accuracy of 0.9234 and an F1 score of 0.9159, the proposed method outperformed the Bag of Words and TF-IDF methods.

Ensemble classifiers for identifying crisis-related text on social media have also been proposed. For instance, Alshehri and Alahamri [36] proposed a two-stage binary ensemble classifier that incorporates natural language processing and machine

learning techniques. The method used in the study, which combines TF-IDF, psychometrics, and linguistic features, has good performance in tweet identification for situational awareness. However, the classification technique may have poor efficacy due to the poor quality of the labels used. Also, Madichetty [37] proposed a method that uses a majority voting-based ensemble technique that uses algorithms such as gradient boost, bagging, AdaBoost, SVM, and random forest to detect medical resource tweets in times of disaster. The methods produce informative features that have less sparsity, dimensionality, and execution time in contrast with the baseline model. However, the study was limited only to earthquake-related disaster-based datasets.

Previous studies on tweet classification for disaster crises primarily focused on utilizing traditional linguistic and psycholinguistic features, semantic features, word embedding features, and contextual features. In addition to the features utilised common classifiers such as support vector machines, random forests, Naive Bayes, K-nearest neighbour, and decision trees, were employed during the classification process. This study distinguishes itself from earlier research in various aspects. It provides a comprehensive examination of ensemble feature sets derived from diverse contextual and linguistic features. Furthermore, both conventional learning methods and ensemble learners were assessed during the classification phase. The empirical findings also include an evaluation of the predictive performance of deep learning-based algorithms specifically applied to identifying disaster crisis tweets. All empirical analyses were conducted using the Queensland flood dataset.

3. Materials and methods

In this section, a description of the data collection, preprocessing, and feature engineering techniques used in this study is provided. Furthermore, the section also described the classification approach by describing the ML classifiers and the ensemble-based techniques. Finally, the evaluation metrics used to evaluate the performance of the classifiers are discussed. Figure 1 depicts the architectural view of the methodology.

3.1. Data collection

The data collection stage is the first and most crucial step in any classification task. Moreover, the quality of the data collected and its use have a substantial impact on the performance of the classifier. In disaster tweet classification tasks, data is either collected directly from Twitter servers with the aid of a Twitter API or from publicly available datasets. However, one of the major drawbacks of data collected directly from Twitter servers is that it requires a significant amount of manual annotation and validation by experts. In contrast, most publicly available Twitter datasets have already been annotated and filtered, making them relatively convenient. As a result, this study used publicly available real-world Twitter datasets. The datasets were collected during the 2013 Queensland floods. The datasets included millions of tweets gathered using the Twitter streaming API using event-specific keywords and hashtags. The tweets were labelled as either ”relevant” or ”not relevant”. Tweets labelled ”relevant” included crisis response information, such as news of injured or killed persons, infrastructure damage, urgent needs of those impacted, and pleas or offers for contributions. Tweets that did not include any of the aforementioned information, on the other hand, were labelled as ”non-relevant.”

3.1.1. About the dataset

The 2013 Queensland flood tweets of 10,033 randomly selected make up the dataset utilised in this study. The tweets were carefully selected to guarantee a balanced representation of both relevant and irrelevant content. In particular, the dataset has 5,418 tweets categorised as ”not relevant” and 4,615 tweets labelled as “relevant”, indicating a fairly even distribution between the two groups. Because each tweet is represented as a text string, there is a wealth of textual data available for study. This dataset balance is especially beneficial for training and assessing machine learning models since it lessens the difficulties caused by class imbalance and guarantees solid and trustworthy outcomes in tasks like topic modelling, sentiment analysis, and binary classification. The availability of such a well-structured dataset increases the possibility of significant insights into public conversation during the Queensland floods, which serves as a useful resource for scholars and practitioners in the fields of natural language processing and disaster response. The dataset can be accessed through the link below https://crisisnlp.qcri.org.

3.2. Data pre-processing

Twitter-based datasets are often characterized by noise due to the presence of emojis, symbols, incomplete sentences, slang, invisible characters, ill-formed words or sentences, and non-dictionary-based words [19]. Thus, the data pre-processing in this study applied several steps to pre-process the Twitter-based dataset. Firstly, all text data was converted to lowercase using the lambda function with the join method and split method. Secondly, all URLs were removed from the text data. Three distinct regular expressions were used to remove URLs: one for URLs starting with “HTTPS”, another for those starting with ”HTTP,” and a third for URLs without these prefixes, which may or may not include www. Thirdly, placeholders such as ”link” and ”video” that may have been introduced in the text data during previous cleaning steps were removed. In the fourth step, HTML reference characters such as & and >were removed. The fifth step involved the removal of non-letter characters from the text data. In the sixth step, Twitter handles were removed. The seventh step comprised of tokenizing the text data using the TweetTokenizer object. Thereafter, all punctuation tokens were removed from the tokenized text data, which comprised the eighth step. The final step was to lemmatize the tokenized words using the WordNetLemmatizer object. Data augmentation methods were not used as data was already balanced and labelled.

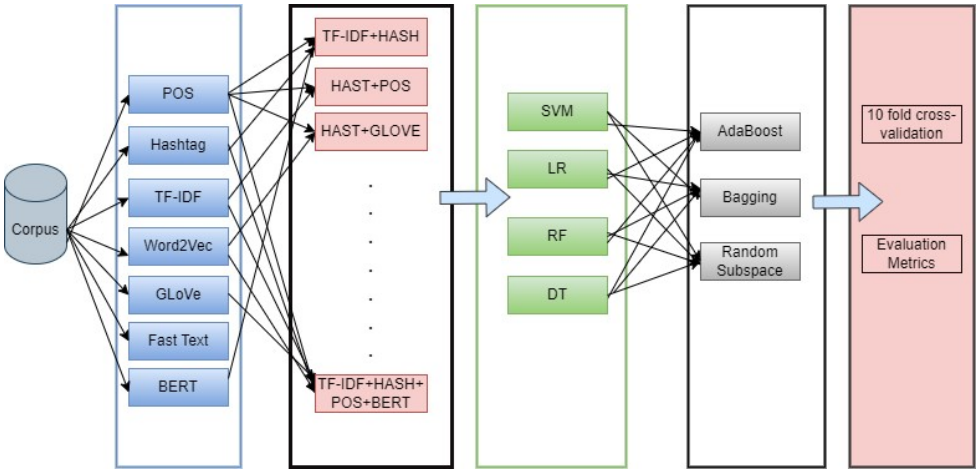

Figure 1: Architecture of the model.

3.3. Feature extraction and selection

In this study, various features were extracted before further classification of the text data. Although the TF-IDF feature was the final feature that was used in the ensemble of classifiers, this section details all the features that were extracted in this research.

(i) Parts of speech (POS) tagging: The first feature extraction technique performed on the Queensland dataset was POS tagging. POS tagging involves assigning a POS label to each word in a sentence, such as noun, verb, adjective, or adverb, among others, to help identify patterns in text data that could improve the machine learning models [38]. To accomplish this, Python code was used to train three POS taggers, a default tagger, a unigram tagger, and a bigram tagger. This was done using the Brown corpus, a large corpus of text data [39]. Thereafter, the code uses the taggers to POS tag the lemmatized words in each dataset. The resulting POS tags were mapped to WordNet POS tags, a more standard format for NLP applications. Finally, counts of the number of occurrences of nouns, verbs, adjectives, and adverbs for each row of data in the Queensland datasets were stored. These counts served as features.

(ii) Hashtags: The second feature extraction technique performed was hash-tagging. Hashtags were extracted and one-hot encoding was applied to the hashtags for the Queensland flood dataset. The one-hot encoded hashtag features were concatenated with the original tweet datasets.

(iii) Term Frequency-Inverse Document Frequency (TF-IDF): The third feature extraction technique performed was the TF-IDF. This is a statistical-based approach that measures the importance of a word in a document [40, 41]. TF-IDF calculates the frequency of occurrence of a word in a particular document and normalizes it in a range between 0 and 1 to eliminate bias between lengthy documents. In this study, the TF-IDF algorithm was used to transform text data into a numerical format by assigning weights to each word or phrase in a document based on its frequency of occurrence in the document and its rarity in the entire corpus. To do this, the TfidfVectorizer class from the Scikit-learn library was used to implement the TF-IDF algorithm. The TF-IDF algorithm was applied to the Queensland dataset. For each dataset, first, the test corpus extracted five documents and applied the TF-IDF algorithm with ngram=1 to generate the features. It then applied the same algorithm to the entire dataset and stored the resulting features in separate variables.

(iv) Word2vec: The fourth feature extraction technique performed was the Word2Vec algorithm. The Word2Vec model converts words into vectors using a cosine similarity formula [42]. To implement this technique, this study used a pre-trained Word2Vec model from the Google News corpus to calculate the similarities between different words such as ”cat” and ”kitten”. The average embedding value and the sentence embedding were defined. These were then used to generate embeddings for the Queensland flood datasets.

(v) Global Vectors for Word Representation (GloVe): The fifth feature extraction technique performed was the Global Vectors for Word Representation (GloVe) model. The GloVe model is used to convert word vectors from a large text corpus into a dense vector space, which can be used to represent word meanings in numerical form [20]. In this research, the glove2word2vec function was imported from the gensim. Scripts module to convert GloVe embeddings to the Word2Vec format. Then a pre-trained GloVe model was loaded and used to generate embeddings for text data from the Queensland flood datasets.

(vi) FastText: The sixth feature extraction technique performed was the FastText model. The FastText model is an extension of the Word2Vec model that can generate embeddings for subword units of words, making it suitable for handling out-of-vocabulary words and capturing more fine-grained information about word meanings [43]. To implement this, a pre-trained FastText model (wiki-news-300d-1M) was loaded and used to generate embeddings

for text data from the Queensland datasets.

(vii) Bidirectional Encoder Representations from Transformers (BERT): The seventh feature extraction technique performed was the BERT model. The BERT algorithm is a pre-trained language model based on the transformer architecture which generates bidirectional representations of text to capture their meaning for downstream NLP tasks [44, 45]. In our study, the BERT model was downloaded from TensorFlow Hub and loaded into a KerasLayer. A 'bert_encode' function, which takes the textual data and tokenizer as input and returns an encoded tensor with tokenized data, padded sequences, and segment IDs for the input sentences was used. Finally, the 'bert_encode' function was then applied to the Queensland datasets.

These features were selected due their ability to capture both the syntactic and semantic properties of tweets [46]. For instance, the POS tags provides grammatical cues, while TF-IDF capture term relevance and word embeddings capture contextual meaning. No automatic feature selection algorithm such as mutual information, and recursive feature elimination was used as features were comprehensively engineered manually [47].

3.4. Classifiers

3.4.1. Supervised learning methods

(i) Decision Tree (DT): DT is a widely used ML algorithm. It handles both classification and regression problems. It has a tree-like structure such that the internal nodes are termed features. While the branches signify a decision concerning the features, the leaf node denotes the result (predicted value) [48]. The DT algorithm constructs the tree-like learning structure through recursive segregation of the dataset using diverse features. Hence, picks the finest features at respective stages of the tree using measures like Gini impurity, or information gain. The main target of DT is to split the data such that impurity at each stage is minimized [49].

(ii) Logistic Regression (LR): LR is a binary classifier that predicts the possibility that a sample data is fit for certain input features [50]. Even though it is named logistic regression, it is a classification-inclined algorithm rather than a regression algorithm. It has a wide range of applicability across fields such as financial analysis, healthcare models, etc. involving binary classifications. It also serves as a benchmark algorithm upon which complex methods are implemented. Logistic regression aims at modelling the relationship involving input features and probabilistic binary outcome generated by a logistic function also called the sigmoid function. The sigmoid function works by mapping real numbers to the binary values of 0 or 1 often described as the actual or predicted class [51].

(iii) Support Vector Machine (SVM): SVM is also a supervised learning model with the capacity to work with both classification and regression problems. It also handles high-dimensional data involving complex decision limits. Its purpose is to find the best hyperplane which can split data instances into various classes. For linearly discrete data, the hyperplane that can maximize the boundary is determined. SVM takes note of the hyperplane and the data points space from each class. Hyperplane normally defines the workable decision limits in logistic regression [52].

(iv) Random Forest (RF): The RF classifier is a supervised learning method that works like the ensemble models. RF handles both the classification as well as regression problems and works very fine on datasets that are very complex or multi-dimensional. Its randomness makes it robust against overfitting. It also gives it the capability to handle complex and multi-dimensional datasets. Thus, handles both the multi-linear relationships in data features and their target values a process that helps in assessing the feature importance in the dataset about all the features [53, 54].

3.4.2. Ensemble methods

(i) AdaBoost: AdaBoost is a boosting-based ensemble learning algorithm. The algorithm trains the base learning models sequentially, and in each round, a new model is constructed. During the training process, the weight values assigned to misclassified samples are increased with each round, while the weight values assigned to correctly classified instances are decreased. Consequently, the algorithm aims to allocate more rounds to challenging cases that are harder to learn and compensate for classification errors made by previous models [15].

(ii) Bagging: Bootstrap Aggregating simply called Bagging is another ensemble method that groups several models to make predictions. Primarily, bagging focuses on minimizing the variance to enhance model performance. The Bagging algorithm, also known as bootstrap aggregating [55], is another method used to build an ensemble. In this approach, different training subsets are obtained from the initial training set through bootstrap sampling [56]. The outputs generated by the base learning algorithms are then combined using majority voting [15].

(iii) Random Subspace: The Random Subspace method [57] is an additional approach used to create an ensemble. It achieves diversity among the ensemble members by partitioning the feature space. In this algorithm, each classification algorithm operates on different random subsets of the feature space. As a result, the technique effectively reduces overfitting while simultaneously improving predictive efficiency (Onan *et al*., 2016).

3.5. Evaluation measures

(i) Accuracy metric: The accuracy metric measures the ratio of the correctly classified samples to the sum of all samples in the data. It is used in classification tasks where the dataset features are labelled. Thus, to measure accuracy, the model's predictions are compared with the true labels of the samples in the test dataset. The model assigns a

class label to each instance, and the accuracy metric assesses how well the model's predictions match the true labels. The formula is expressed as:

$$Accuracy = \frac{TN + TP}{TP + TN + FP + FN}, \quad (1)$$

where TP = True Positive, TN = True Negative, FN = False Negative, and FP = False Positive.

(ii) Precision: Precision is used to measure the proportion of correctly predicted positive instances out of all samples predicted as positive by a model, often used in situations where the focus is on minimizing false positives. Thus, high precision indicates that the model has a low rate of false positives, making it more reliable for applications where false positives are costly or undesirable [58]. Mathematically, precision can be expressed as follows:

$$Precision = \frac{TP}{TP + FP}. \quad (2)$$

(iii) Recall: Recall, also known as sensitivity or true positive rate, is an evaluation metric commonly used in machine learning to assess the performance of a classifier, particularly in situations where the focus is on minimizing false negatives. Recall measures the proportion of correctly predicted positive instances out of all actual positive instances in the dataset [59]. Recall is denoted mathematically as:

$$Recall = \frac{TP}{TP + FN}. \quad (3)$$

(iv) F1-score: The F1 score provides a balance between precision and recall and is especially useful in situations where both false positives and false negatives are required to be minimized. It is simply the harmonic mean of precision and recall. This penalizes extreme values, making it a suitable metric when there is an imbalance between precision and recall. It ranges from 0 to 1, where 1 represents the best possible F1 score (perfect precision and recall), and 0 represents the worst score (either precision or recall is 0). F1-score is mathematically expressed as:

$$F - score = 2 \times \frac{P \times R}{P + R}, \quad (4)$$

where P denotes Precision and R denotes Recall.

(v) Area Under the Curve (AUC): The AUC evaluation metric measures the overall quality of predictions across different thresholds or decision boundaries [60]. AUC represents the area under the receiver operating characteristic (ROC) curve. The ROC curve is created by plotting the true positive rate (sensitivity or recall) on the y-axis against the false positive rate (1 - specificity) on the x-axis at different classification thresholds. A good model would have an AUC of 1, indicating that it achieves perfect classification between the classes involved.

Table 1: Model parameter settings.

Algorithm	Parameter	Value
Support Vector Machine	Regularization strength	c=0.1
	Penalty	l1
	Optimization formulation	dual=False
	Number of iterations	10000
	Random state	42
Logistic Regression	Inverse regularization strength	Cs=10
	Cross-validation folds	cv=5
	Penalty	l1
	Optimization algorithm	liblinear
	Random state	42
Random Forest	Number of trees	100
	Out-of-bag scores	True
	Warm start	True
	Number of jobs	-1
	Random state	42
Decision Tree	Maximum depth	2, 4, 6
	Minimum samples split	2, 4, 6
	Minimum samples at leaf	1, 2, 3
	Max features	sqrt, log2, None
	Cross-validation folds	fold = 5
	Random state	42

3.6. Experimental procedure

In this section, the experimental procedure employed in the study is presented. The study considered seven primary feature sets, including TF-IDF, HASHTAG, POS, FAST TEXT, GLOVE, W2VEC, and BERT, for feature extraction. These feature sets, along with various Ensemble combinations were utilised in our empirical analysis. This approach resulted in a total of 31 feature sets. To assess the predictive performance of the proposed feature sets in comparison to traditional text representation methods. Also, the evaluation metrics for the unigram model based on term frequency representation are included. Five supervised learning algorithms, including the decision tree algorithm, logistic regression, support vector machines, and random forest algorithm were employed. Furthermore, the ensembles of these classifiers using three ensemble learning techniques: AdaBoost, Bagging, and the random subspace algorithms were used. All the experiments were conducted using the Jupyter Notebook in the Python programming environment. Table 1 outlines the basic parameter settings for both the conventional classifiers and the ensemble learning methods.

4. Experimental results

This section provides the predictive performance results in terms of accuracy, precision, recall, F1-score and AUC values obtained by modelling the proposed features using the conventional supervised learning models and ensemble classifiers. The

result reveals the predictive capabilities of various features like TF-IDF, Hashtag, POS, and word embeddings such as GloVe, FastText, and Word2Vec, including their ensemble combinations when used as feature sets for classifying disaster response tweets.

4.1. Performance of the conventional classifiers

Table 2 presents the classification accuracy and precision results achieved on different feature sets using typical supervised learning algorithms.

As presented in Table 2, it is evident that the logistic regression classifier typically outperforms the other classification algorithms in terms of classification accuracy across many feature sets. Followed by the random forest algorithm. While, decision tree and SVM are the least performed algorithms based on the accuracy metric. Similarly, in terms of precision, random forest outperformed followed by the decision tree algorithm. Both the support vector machine and logistic regression algorithms underperformed across all the features.

Similarly, Table 3 displays the F1-score and AUC performance results obtained from applying the same conventional supervised learning algorithms.

In Table 3, the random forest classifier typically outperforms the other classification algorithms in terms of F1-score values across the several feature sets. Followed by the decision tree and logistic regression algorithms. While SVM appeared to be the least performed algorithm based on the F1-score metric. However, in terms of AUC values, the SVM algorithm demonstrates the highest predictive performance across almost all the feature sets experimented with. The random forest algorithm follows in performance, whereas, the logistic regression and decision tree algorithms tend to underperform across all the feature sets.

Table 4 focuses on the recall values obtained from the conventional classifiers utilised in the study.

Table 4 shows that the logistic regression classifier typically outperforms the other classification algorithms across several feature sets in terms of recall values. Followed by the SVM and logistic regression algorithms. While decision tree appeared to be the least performed algorithm based on the recall metric.

4.2. Performance of the ensemble classifiers

To enhance the predictive capabilities of the traditional supervised learning methods, an ensemble learning approach was employed. This addresses the third objective of this empirical analysis which is to determine whether ensemble learners can achieve superior predictive results for disaster response classification. In this regard, three well-established ensemble learners, namely bagging (B), AdaBoost (A), and random forest (RS) were leveraged. The experimental results are shown concerning the evaluation metrics utilised. Table 5 presents the predictive performance of the ensemble learners when combined with standard learners and ensembles using the accuracy metric.

As evident in the results presented in Table 5, the employment of ensemble learning techniques generally led to improvements in the evaluation metrics compared to those achieved by standard classification algorithms. In terms of the outcomes obtained from different ensemble learning techniques, the A-LR algorithm consistently demonstrated the most remarkable predictive performance across several feature sets reaching 0.9662 on the TF-IDF+ FST TEXT feature set. A-RF and A-DT followed closely with 0.9651 on the TF-IDF+HASH+GLOVE feature set. Also, the B-RF classifier strongly performed on some feature sets like TF-IDF+HASH+POS reaching 0.9623 on the accuracy metric. The RS-SVM algorithm consistently underperformed across all the feature sets, reaching 0.5404 with HASH and 0.5382 with POS + BERT feature set.

More so, the performance of the ensemble classifiers based on their precision values was assessed as presented in Table 6.

From Table 6, the A-RF and the B-DT algorithms consistently outperform the rest of the algorithms scoring the highest precision values of 0.985 and 0.9892 on TF-IDF, TF-IDF + HASH, and TF-IDF + HASH + BERT feature sets, respectively. The RS-RF algorithm also achieved an outstanding result of 0.9892 on the TF-IDF + HASH + BERT feature set. The least performed algorithms and feature sets include RS-SVM with 0.5402, and 0.6158 precision values on the HASHTAG and POS feature sets respectively.

Similarly, Table 7 presents the performances of the various ensemble learners on the diverse feature sets based on the recall evaluation metric.

As seen in Table 7, A-DT and B-DT returned a recall value of 1.0000 with the HASHTAG feature set. This is followed by the RS-SVM, A-SVM and A-RF algorithms reaching 0.9846, 0.9564 and 0.9436, respectively on the TF-IDF+HASH+POS+BERT feature set combination. The lowest recall value was achieved from A-RF (0.2072) and B-SVM (0.2533) using the HASHTAG feature set.

Again, the F1-score performance of the ensemble learners were also evaluated across all the feature sets as presented in Table 8.

Table 8 indicates that the A-DT algorithm yielded the best result of 0.9672 and 0.9667 F1-score value on TF-IDF + HASH + GLOVE and TF-IDF + POS + GLOVE feature sets, respectively. Following closely, the A-RF algorithm reached 0.9655 F1-score value on the same TF-IDF + HASH + GLOVE feature set. However, on a different feature set (HASH), the A-RF was the least performed algorithm with 0.3415 F1-score value.

Furthermore, the AUC performance across the ensemble learners and the feature sets was also assessed. The experimental results are presented in Table 9.

In terms of the AUC evaluation metric performance as shown in Table 9, the TF-IDF and FastText performed excellently with 0.9581 for A-SVM, 0.9671 for A-LR, and 0.9622 for A-DT. The least performing algorithm B-SVM, achieved a low AUC value of 0.6364.

Overall, across all the ensemble algorithms, the A-RF ensemble model consistently yielded outstanding results with a precision value of 0.9892 using the TF-IDF + FST TEXT feature set combination.

Table 2: Accuracy and precision results for the conventional classifier.

Feature Set	SVM	LR	RF	DT	SVM	LR	RF	DT
TF-IDF	0.9535	0.9640	0.9590	0.9635	0.9714	0.9649	0.9860	0.9861
HASHTAG	0.5620	0.5936	0.5637	0.5598	0.5524	0.7637	0.5533	0.9737
POS	0.6539	0.6528	0.6678	0.6661	0.6686	0.6515	0.6767	0.6626
FAST TEXT	0.8594	0.8594	0.8942	0.8056	0.8616	0.8594	0.9252	0.8047
GLOVE	0.8588	0.8594	0.8893	0.8162	0.8600	0.8594	0.9235	0.8019
W2VEC	0.8505	0.8522	0.8920	0.8311	0.8615	0.8521	0.9333	0.8215
BERT	0.6168	0.6351	0.8245	0.7137	0.6329	0.6349	0.8500	0.7608
TF-IDF +HASH	0.9546	0.9651	0.9635	0.9640	0.9735	0.9658	0.9861	0.9861
HASH+POS	0.7016	0.7027	0.7060	0.6755	0.7202	0.7038	0.7114	0.7335

Table 3: F1-score and AUC performance results for the conventional classifier.

Feature Set	SVM	LR	RF	DT	SVM	LR	RF	DT
TF-IDF	0.9562	0.9641	0.9611	0.9654	0.9802	0.9654	0.9609	0.9650
HASHTAG	0.7104	0.5373	0.7111	0.3176	0.6417	0.6227	0.5263	0.5919
POS	0.6895	0.6505	0.7053	0.7154	0.6993	0.6466	0.6618	0.6565
FAST TEXT	0.8712	0.8591	0.8993	0.8244	0.9267	0.8570	0.8959	0.8022
GLOVE	0.8709	0.8591	0.8942	0.8373	0.9256	0.8571	0.8912	0.8110
W2VEC	0.8615	0.8521	0.8960	0.8488	0.9224	0.8509	0.8947	0.8271
BERT	0.6608	0.6278	0.8345	0.7210	0.6639	0.6244	0.8249	0.7162
TF-IDF +HASH	0.9572	0.9652	0.9654	0.9660	0.9801	0.9664	0.9650	0.9655
HASH+POS	0.7257	0.7030	0.7378	0.6759	0.7736	0.7022	0.7008	0.6798

Table 4: Recall performance results for the conventional classifier.

Feature Set	SVM	LR	RF	DT
TF-IDF	0.9415	0.9640	0.9374	0.9456
HASHTAG	0.9949	0.5936	0.9949	0.1897
POS	0.7118	0.6528	0.7364	0.7774
FAST TEXT	0.8810	0.8594	0.8749	0.8451
GLOVE	0.8821	0.8594	0.8667	0.8759
W2VEC	0.8615	0.8522	0.8615	0.8779
BERT	0.6913	0.6351	0.8195	0.6851
TF-IDF +HASH	0.9415	0.9651	0.9456	0.9467
HASH+POS	0.7313	0.7027	0.7662	0.6267

5. Discussions

The experimental results demonstrate that the AdaBoost-based random forest (A-RF) ensemble model using TF-IDF + FST TEXT achieved the highest result as evaluated with the precision metric. This high performance can be likened to the inherent nature of Bagging ensembles, which integrate multiple weak learners to reduce variance while preventing overfitting. The use of the TF-IDF + FST TEXT feature set further improves its ability to weigh terms according to their document frequency, which is effective in tweet classification.

Similarly, conventional algorithms like SVM and RF also demonstrated high performance results especially when using the TF-IDF feature set. The inclusion of the Part-Of-Speech tagging as a standalone feature yielded lower accuracy. This suggests that it lacks discriminative ability any time it is used in isolation. The experimental results from the study align with those reported in prior studies. Particularly, Onan, *et al.* [15] achieved a promising result ensemble classifier for text classification. Thus, the study reinforces the performance of Bagging in the present study which is in a similar context. Again, the experimental findings resonate with that of ALRashdi and O'Keefe [19], who also reported that conventional classifiers that use the TF-IDF feature set outclassed deep learning models in resource-constrained settings. The use of fused features like TF-IDF + POS + GloVe and TF-IDF + FST TEXT also showed improvements over single feature-based models. Deep learning models are also highly favoured by the fused features. The findings from the current study suggest that well-tuned conventional models are very competitive especially when dealing with small or medium-sized datasets.

This finding has significant implications for emergency response systems and social media monitoring tools. The exceptional performance of Bagging demonstrates that sophisticated models for tackling emergencies can be developed without involving complex deep-learning models. This is advantageous as model interpretability challenges eminent among deep learning techniques as well as resource constraint issues are by this handled. However, this study is not completely free from limitations. First, the dataset size though adequate may not capture the full diversity of tweets across different disaster scenarios, languages or contexts. In addition, the low performance of some models might be a result of the inability to fine-tune them. Thus, future works should take into consideration the use of pre-trained models or properly fine-tuned models on domain-specific and multilingual datasets to enhance the generalisabil-

Table 5: Ensemble classification accuracy performance values.

Feature Set	A-SVM	A-LR	A-RF	A-DT	B-SVM	B-LR	B-RF	B-DT	RS-SVM	RS-LR	RS-RF	RS-DT
TF-IDF	0.9537	0.9635	0.9607	0.9618	0.9363	0.9568	0.9596	0.9618	0.5869	0.9081	0.9563	0.9579
HASHTAG	0.5593	0.5681	0.5687	0.5399	0.5914	0.5925	0.5626	0.5399	0.5404	0.5548	0.5626	0.5421
POS	0.6549	0.6689	0.6556	0.6423	0.6578	0.6495	0.6639	0.6539	0.6174	0.6312	0.6689	0.6501
FAST TEXT	0.8364	0.8283	0.8527	0.8128	0.8533	0.8333	0.8865	0.8173	0.8245	0.7907	0.8854	0.8278
GLOVE	0.8405	0.8355	0.8433	0.8267	0.8522	0.8372	0.8843	0.8084	0.8140	0.8389	0.8893	0.8173

Table 6: Ensemble classification precision performance values.

Feature Set	A-SVM	A-LR	A-RF	A-DT	B-SVM	B-LR	B-RF	B-DT	RS-SVM	RS-LR	RS-RF	RS-DT
TF-IDF	0.9688	0.9637	0.9809	0.9778	0.9715	0.9582	0.9892	0.9694	0.5696	0.9133	0.9848	0.9828
HASHTAG	0.7061	0.7616	0.9712	0.5399	0.9611	0.7632	0.5399	0.9612	0.5402	0.7402	0.5527	0.5411
POS	0.6598	0.6706	0.6712	0.629	0.6755	0.6485	0.6471	0.6728	0.6158	0.6296	0.6593	0.644
FAST TEXT	0.8350	0.8282	0.8506	0.8095	0.8615	0.8332	0.8342	0.8515	0.8270	0.7935	0.9201	0.8395
GLOVE	0.8396	0.8355	0.8446	0.8371	0.8590	0.8371	0.8117	0.8528	0.8111	0.8388	0.9282	0.8203

Table 7: Ensemble classification recall performance values.

Feature Set	A-SVM	A-LR	A-RF	A-DT	B-SVM	B-LR	B-RF	B-DT	RS-SVM	RS-LR	RS-RF	RS-DT
TF-IDF	0.9432	0.9635	0.9456	0.9508	0.9087	0.9568	0.9364	0.9395	0.9610	0.9081	0.9333	0.9385
HASHTAG	0.6910	0.5681	0.2072	1.0000	0.2533	0.5925	0.9928	1.0000	1.0000	0.5548	0.9949	1.0000
POS	0.7275	0.6689	0.7097	0.8226	0.7046	0.6495	0.7333	0.7897	0.7744	0.6312	0.8000	0.7867
FAST TEXT	0.8607	0.8283	0.8821	0.8544	0.8677	0.8333	0.8656	0.8256	0.8533	0.7907	0.8626	0.8421
GLOVE	0.8635	0.8355	0.8697	0.8431	0.8687	0.8372	0.8564	0.84	0.8544	0.8389	0.8615	0.8472

Table 8: Ensemble classification F1-score performance values.

Feature Set	A-SVM	A-LR	A-RF	A-DT	B-SVM	B-LR	B-RF	B-DT	RS-SVM	RS-LR	RS-RF	RS-DT
TF-IDF	0.9557	0.9635	0.9629	0.9641	0.9391	0.9569	0.9616	0.9637	0.7153	0.9082	0.9584	0.9601
HASHTAG	0.5618	0.496	0.3415	0.7012	0.4010	0.5356	0.7102	0.7012	0.7014	0.4121	0.7106	0.7022
POS	0.6884	0.6627	0.6899	0.7129	0.6898	0.6455	0.702	0.7113	0.6861	0.6289	0.7229	0.7082
FAST TEXT	0.8476	0.8282	0.8661	0.8313	0.8646	0.8332	0.8917	0.8299	0.8400	0.7887	0.8904	0.8408
GLOVE	0.8511	0.8353	0.857	0.8401	0.8638	0.8371	0.8888	0.8256	0.8322	0.8386	0.8936	0.8335

Table 9: Ensemble classification AUC performance values.

Feature Set	A-SVM	A-LR	A-RF	A-DT	B-SVM	B-LR	B-RF	B-DT	RS-SVM	RS-LR	RS-RF	RS-DT
TF-IDF	0.9547	0.9641	0.962	0.9627	0.9807	0.9586	0.9616	0.9637	0.5781	0.9117	0.9582	0.9596
HASHTAG	0.5557	0.5995	0.6	0.5	0.6364	0.6217	0.5253	0.5	0.5026	0.5163	0.5251	0.5024
POS	0.6461	0.6587	0.6509	0.6267	0.7033	0.6414	0.6579	0.6422	0.6035	0.6250	0.6575	0.6382
FAST TEXT	0.8338	0.8265	0.8502	0.8092	0.9252	0.8318	0.8883	0.8166	0.8970	0.7842	0.8874	0.8266
GLOVE	0.8379	0.8331	0.841	0.8253	0.9246	0.8354	0.8867	0.8057	0.8850	0.8366	0.8917	0.8147

ity of disaster classifiers.

6. Conclusion

Twitter has emerged as a prominent platform for individuals and organizations to disseminate or collect information during times of disaster [61]. It serves as a means for people to share crucial, real-time, and pertinent details such as their well-being, casualties, and the extent of damage caused by the disaster [62]. Additionally, Twitter is frequently utilised to request assistance or extend aid to others. Over recent years, Twitter has proven to be a valuable source of information during various natural and human-induced disaster scenarios, including earthquakes, wildfires, floods, and nuclear incidents. Automatically categorizing tweets related to disaster response poses a substantial challenge in the field of natural language processing. This study introduces a machine learning-based approach for classifying disaster response tweets using a benchmark dataset from the Queensland flood. The extensive examination of various feature sets, classifiers, and ensemble techniques revealed that combining Term Frequency-Inverse Document Frequency (TF-IDF) and word embedding features with ensemble learning methods can produce highly promising predictive outcomes for classifying disaster-related content. By employing ensemble feature combinations along with the Bagging ensemble method of the decision tree algorithm, an impressive classification accuracy of 98.92% was achieved. The experimental findings suggest that employing deep learning techniques can result in superior predictive performance when it comes to satire identification.

This study has demonstrated the effectiveness of ensemble-based machine learning approaches using engineered text features for disaster-related tweet classification. However, future

research could explore more sophisticated Natural Language Processing models such as BERT, RoBERTa and GPT, capable of capturing deeper contextual relationships. Additionally, the extension of this work to multilingual tweet classification would enhance its applicability in global crisis contexts. Developing models that can handle code-switching or mixed language content is particularly important for real-world deployment. Furthermore, short learning and adaptation techniques could be investigated to improve model generalisability in low-resource or novel disaster scenarios.

Data availability

The data utilised in this study, which supports the findings from the research are available based on the request.

References

[1] J. B. Houston, G. Hawthorne, M. F. Perreault, E. H. Park, M. Goldstein Hode, M. R. Halliwell, S. E. Turner McGowen, R. Davis, S. Vaid, J. A. McElderry & S. A. Griffith, "Social media and disasters: a functional framework for social media use in disaster planning, response, and research", Disasters **39** (2015) 1. https://doi.org/10.1111/disa.12092.

[2] L. Xukun & D. Caragea, "Improving disaster-related tweet classification with a multimodal approach", in *ISCRAM 2020 Conference Proceedings–17th International Conference on Information Systems for Crisis Response and Management*, 2020. https://par.nsf.gov/servlets/purl/10204504.

[3] H. Li, D. Caragea, C. Caragea & N. Herndon, "Disaster response aided by tweet classification with a domain adaptation approach", Journal of Contingencies and Crisis Management **26** (2018) 16. https://doi.org/10.1111/1468-5973.12194.

[4] W. Zhai, "A multi-level analytic framework for disaster situational awareness using Twitter data", Computational Urban Science **2** (2022) 23. https://doi.org/10.1007/s43762-022-00052-z.

[5] K. Maswadi, A. Alhazmi, F. Alshanketi & C. I. Eke, "The empirical study of tweet classification system for disaster response using shallow and deep learning models", Journal of Ambient Intelligence and Humanized Computing **15** (2024) 3303. https://doi.org/10.1007/s12652-024-04807-w.

[6] A. Kumar & J. P. Singh, "Location reference identification from tweets during emergencies: A deep learning approach", International Journal of Disaster Risk Reduction **33** (2019) 365. https://arxiv.org/pdf/1901.08241.

[7] J. A. de Bruijn, H. C. Winsemius, M. J. Wanders, E. J. M. van den Berg & H. H. G. Savenije, "Improving the classification of flood tweets with contextual hydrological information in a multimodal neural network", Computers & Geosciences **140** (2020) 104485. https://doi.org/10.1016/j.cageo.2020.104485.

[8] W. Gata, F. Amsury, N. K. Wardhani, I. Sugiyarto, D. N. Sulistyowati & I. Saputra, "Informative tweet classification of the earthquake disaster situation in Indonesia", in *2019 5th International Conference on Computing Engineering and Design (ICCED)*, 2019, pp. 1–6. http://dx.doi.org/10.1109/ICCED46541.2019.9161135.

[9] R. Thomson, N. Ito, H. Suda, F. Lin, Y. Liu, R. Hayasaka, R. Isochi & Z. Wang, "Trusting tweets: the Fukushima disaster and information source credibility on Twitter", in *ISCRAM*, 2012. https://www.emknowledge.org.au/ISCRAM2012/proceedings/112.pdf.

[10] S. A. Morshed, K. M. Ahmed, K. Amine & K. A. Moinuddin, "Trend analysis of large-scale Twitter data based on witnesses during a hazardous event: a case study on California wildfire evacuation", World Journal of Engineering and Technology **9** (2021) 229. https://doi.org/10.4236/wjet.2021.92016.

[11] H. Li, D. Caragea & C. Caragea, "Combining self-training with deep learning for disaster tweet classification", in *The 18th International Conference on Information Systems for Crisis Response and Management (ISCRAM 2021)*, 2021. https://par.nsf.gov/servlets/purl/10308599.

[12] J. Qadir, A. Ali, R. ur Rasool, A. Zwitter, A. Sathiaseelan & J. Crowcroft, "Crisis analytics: big data-driven crisis response", Journal of International Humanitarian Action **1** (2016) 1. https://doi.org/10.1186/s41018-016-0013-9.

[13] C. I. Eke, A. A. Norman, L. Shuib & H. F. Nweke, "Sarcasm identification in textual data: systematic review, research challenges and open directions", Artificial Intelligence Review **53** (2020) 4215. https://doi.org/10.1007/s10462-019-09791-8.

[14] A. Mohammed & R. Kora, "An effective ensemble deep learning framework for text classification", Journal of King Saud University-Computer and Information Sciences **34** (2022) 8825. https://doi.org/10.1016/j.jksuci.2021.11.001.

[15] A. Onan, S. Korukoğlu & H. Bulut, "Ensemble of keyword extraction methods and classifiers in text classification", Expert Systems with Applications **57** (2016) 232. https://doi.org/10.1016/j.eswa.2016.03.045.

[16] O. Sagi & L. Rokach, "Ensemble learning: a survey", Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery **8** (2018) e1249. http://dx.doi.org/10.1002/widm.1249.

[17] C. I. Eke, A. A. Norman, L. Shuib & Z. A. Long, "Random forest-based classifier for automatic sarcasm classification on twitter data using multiple features", Journal of Information Systems and Digital Technologies **4** (2022) 205. file:///C:/Users/hp/Downloads/205.pdf.

[18] L. Rokach, *Ensemble learning: pattern classification using ensemble methods*, World Scientific, Singapore, 2019, pp. 1–300. http://dx.doi.org/10.1142/11325.

[19] R. ALRashdi & S. O'Keefe, "Deep learning and word embeddings for tweet classification for crisis response", arXiv preprint arXiv:1903.11024, 2019. [Online]. https://arxiv.org/abs/1903.11024.

[20] C. I. Eke, A. Norman, L. Shuib, F. B. Fatokun & I. Omame, "The significance of global vectors representation in sarcasm analysis", in *2020 International Conference in Mathematics, Computer Engineering and Computer Science (ICMCECS)*, 2020, pp. 1–7. http://dx.doi.org/10.1109/ICMCECS47690.2020.246997.

[21] T. Sahni, C. Chandak, N. R. Chedeti & M. Singh, "Efficient Twitter sentiment classification using subjective distant supervision", in *2017 9th International Conference on Communication Systems and Networks (COMSNETS)*, 2017, pp. 548–553. https://arxiv.org/pdf/1701.03051.

[22] V. K. Neppalli, C. Caragea & D. Caragea, "Deep neural networks versus naive Bayes classifiers for identifying informative tweets during disasters", in *Proceedings of the 15th Annual Conference for Information Systems for Crisis Response and Management (ISCRAM)*, 2018. https://par.nsf.gov/servlets/purl/10204522.

[23] E. Schnebele, G. Cervone, S. Kumar & N. Waters, "Real time estimation of the Calgary floods using limited remote sensing data", Water **6** (2014) 381. https://doi.org/10.3390/w6020381.

[24] N. Naderi, *Computational analysis of arguments and persuasive strategies in political discourse*, University of Toronto (Canada), 2020. https://utoronto.scholaris.ca/server/api/core/bitstreams/91f8c5fa-6fe8-4e7d-b9a6-444da6c95370/content.

[25] M. Basu, A. Shandilya, P. Khosla, K. Ghosh & S. Ghosh, "Extracting resource needs and availabilities from microblogs for aiding post-disaster relief operations", IEEE Transactions on Computational Social Systems **6** (2019) 604. http://dx.doi.org/10.1109/TCSS.2019.2914179.

[26] S. Kumar, X. Hu & H. Liu, "A behavior analytics approach to identifying tweets from crisis regions", in *Proceedings of the 25th ACM Conference on Hypertext and Social Media*, 2014, pp. 255–260. https://doi.org/10.1145/2631775.2631814.

[27] H. Purohit, C. Castillo, F. Diaz, A. Sheth & P. Meier, "Emergency-relief coordination on social media: Automatically matching resource requests and offers", First Monday **19** (2014). http://dx.doi.org/10.5210/fm.v19i1.4848.

[28] S. Verma, G. Vieweg, W. J. Corvey, L. Palen, J. H. Martin, M. Palmer, A. Schram & K. M. Anderson, "Natural language processing to the rescue? Extracting 'situational awareness' tweets during mass emergency", in *Proceedings of the International AAAI Conference on Web and Social Media*, 2011, pp. 385–392. https://doi.org/10.1609/icwsm.v5i1.14119.

[29] M. Imran, S. Elbassuoni, C. Castillo, F. Diaz & P. Meier, "Extracting information nuggets from disaster-related messages in social media", ISCRAM **201** (2013) 791. https://idl.iscram.org/files/imran/2013/613_Imran_etal2013.pdf.

[30] Y. Kryvasheyeu, H. Chen, N. Obradovich, E. Moro, P. Van Hentenryck, J. Fowler & M. Cebrian, "Rapid assessment of disaster damage using social media activity", Science Advances **2** (2016) e1500779. https://www.science.org/doi/pdf/10.1126/sciadv.1500779.

[31] P. Khare, G. Burel, D. Maynard & H. Alani, "Cross-lingual classification of crisis data", in *The Semantic Web–ISWC 2018: 17th International Semantic Web Conference, Monterey, CA, USA, October 8–12, 2018, Proceedings, Part I 17*, 2018, pp. 617–633. https://doi.org/10.1007/978-3-030-00671-6_36.

[32] G. Burel, H. Saif & H. Alani, "Semantic wide and deep learning for detecting crisis-information categories on social media", in *The Semantic Web–ISWC 2017: 16th International Semantic Web Conference, Vienna, Austria, October 21–25, 2017, Proceedings, Part I 16*, 2017, pp. 138–155. https://oro.open.ac.uk/51726/1/322.pdf.

[33] M. Y. Kabir & S. Madria, "A deep learning approach for tweet classification and rescue scheduling for effective disaster management", in *Proceedings of the 27th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2019, pp. 269–278. http://dx.doi.org/10.1145/3347146.3359097.

[34] A. Bhoi, S. P. Pujari & R. C. Balabantaray, "A deep learning-based social media text analysis framework for disaster resource management", Social Network Analysis and Mining **10** (2020) 1. https://link.springer.com/article/10.1007/s13278-020-00692-1.

[35] S. Kundu, P. Srijith & M. S. Desarkar, "Classification of short-texts generated during disasters: a deep neural network based approach", in *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 2018, pp. 790–793. http://dx.doi.org/10.1109/ASONAM.2018.8508695.

[36] A. Alshehri & S. Alahamri, "An ensemble learning for detecting situational awareness tweets during environmental hazards", in *2019 IEEE International Systems Conference (SysCon)*, 2019, pp. 1–8. https://doi.org/10.1109/SYSCON.2019.8836814.

[37] S. Madichetty, "Identification of medical resource tweets using majority voting-based ensemble during disaster", Social Network Analysis and Mining **10** (2020) 66. https://doi.org/10.1007/s13278-020-00679-y.

[38] A. Chiche & B. Yitagesu, "Part of speech tagging: a systematic review of deep learning and machine learning approaches", Journal of Big Data **9** (2022) 10. https://doi.org/10.1186/s40537-022-00561-y.

[39] A. Priyadarshi & S. K. Saha, "Towards the first Maithili part of speech tagger: Resource creation and system development", Computer Speech & Language **62** (2020) 101054. https://doi.org/10.1016/j.csl.2019.101054.

[40] N. N. A. Sjarif, N. F. M. Azmi, S. Chuprat, H. M. Sarkan, Y. Yahya & S. M. Sam, "SMS spam message detection using term frequency-inverse document frequency and random forest algorithm", Procedia Computer Science **161** (2019) 509. https://doi.org/10.1016/j.procs.2019.11.150.

[41] C. I. Eke, A. A. Norman & L. Shuib, "Multi-feature fusion framework for sarcasm identification on Twitter data: A machine learning based approach", PLoS One **16** (2021) e0252918. https://doi.org/10.1371/journal.pone.0252918.

[42] D. Jatnika, M. A. Bijaksana & A. A. Suryani, "Word2vec model analysis for semantic similarities in English words", Procedia Computer Science **157** (2019) 160. https://doi.org/10.1016/j.procs.2019.08.153.

[43] Z. Quan, Z.-J. Wang, Y. Le, B. Yao, K. Li & J. Yin, "An efficient framework for sentence similarity modeling", IEEE/ACM Transactions on Audio, Speech, and Language Processing **27** (2019) 853. https://cszjwang.github.io/sub_pages/pps/TALSP19.pdf.

[44] C. I. Eke, A. A. Norman & L. Shuib, "Context-based feature technique for sarcasm identification in benchmark datasets using deep learning and BERT model", IEEE Access **9** (2021) 48501. http://dx.doi.org/10.1109/ACCESS.2021.3068323.

[45] A. Alhazmi, R. Mahmud, N. Idris, M. E. Mohamed Abo & C. I. Eke, "Code-mixing unveiled: enhancing the hate speech detection in Arabic dialect tweets using machine learning models", PLoS One **19** (2024) e0305657. https://doi.org/10.1371/journal.pone.0305657.

[46] A. Yusuf, R. Dima & S. Aina, "Optimized breast cancer classification using feature selection and outliers detection", Journal of the Nigerian Society of Physical Sciences **3** (2021) 298. https://doi.org/10.46481/jnsps.2021.331.

[47] P. U. Emmoh, C. I. Eke, T. Moses & A. Ovre, "Feature selection techniques for high-dimensional data analysis: applications, challenges, and future directions", Nigerian Journal of Technological Development **22** (2025) 201. https://doi.org/10.63746/njtd.v22i1.2943.

[48] S. Tangirala, "Evaluating the impact of GINI index and information gain on classification using decision tree classifier algorithm", International Journal of Advanced Computer Science and Applications **11** (2020) 612. http://dx.doi.org/10.14569/IJACSA.2020.0110277.

[49] Z. Mohammadi-Pirouz, K. Hajian-Tilaki, M. Sadeghi Haddat-Zavareh, A. Amoozadeh & S. Bahrami, "Development of decision tree classification algorithms in predicting mortality of COVID-19 patients", International Journal of Emergency Medicine **17** (2024) 126. https://doi.org/10.1186/s12245-024-00681-7.

[50] P. U. Emmoh & T. Moses, "A feature selection and scoring scheme for dimensionality reduction in a machine learning task", Journal of the Nigerian Society of Physical Sciences **5** (2025) 2273. https://doi.org/10.46481/jnsps.2025.2273.

[51] X.-S. Yang, *Introduction to algorithms for data mining and machine learning*, Academic Press, Cambridge, MA, 2019, pp. 1–300. http://dx.doi.org/10.1016/C2018-0-02034-4.

[52] O. Okwuashi & C. E. Ndehedehe, "Deep support vector machine for hyperspectral image classification", Pattern Recognition **103** (2020) 107298. https://doi.org/10.1016/j.patcog.2020.107298.

[53] Y. Al Amrani, M. Lazaar & K. E. El Kadiri, "Random forest and support vector machine based hybrid approach to sentiment analysis", Procedia Computer Science **127** (2018) 511. https://doi.org/10.1016/j.procs.2018.01.150.

[54] L. Zhu, D. Qiu, D. Ergu, C. Ying & K. Liu, "A study on predicting loan default based on the random forest algorithm", Procedia Computer Science **162** (2019) 503. https://doi.org/10.1016/j.procs.2019.12.017.

[55] L. Breiman, "Bagging predictors", Machine learning **24** (1996) 123. https://doi.org/10.1007/BF00058655.

[56] D. O. Oyewola, E. G. Dada, J. N. Ndunagu, T. A. Umar & A. Sa, "COVID-19 risk factors, economic factors, and epidemiological factors nexus on economic impact: machine learning and structural equation modelling approaches", Journal of the Nigerian Society of Physical Sciences **3** (2021) 395. https://doi.org/10.46481/jnsps.2021.173.

[57] T. K. Ho, "The random subspace method for constructing decision forests", IEEE Transactions on Pattern Analysis and Machine Intelligence **20** (1998) 832. https://www.ehu.eus/ccwintco/uploads/4/45/Presetacion-ibarandiaran-2012-01-27.pdf.

[58] M. Hossin & M. N. Sulaiman, "A review on evaluation metrics for data classification evaluations", International Journal of Data Mining & Knowledge Management Process **5** (2015) 1. http://dx.doi.org/10.5121/ijdkp.2015.5201.

[59] D. K. Kwaghtyo & C. I. Eke, "Smart farming prediction models for precision agriculture: a comprehensive survey", Artificial Intelligence Review **56** (2023) 5729. https://doi.org/10.1007/s10462-022-10266-6.

[60] M. Mourad, M. El-Seoud, A. El-Sayed, H. El-Bassiouny & H. El-Bahnasawy, "Machine learning and feature selection applied to SEER data to reliably assess thyroid cancer prognosis", Scientific Reports **10** (2020) 5176. https://doi.org/10.1038/s41598-020-62023-w.

[61] D. T. Nguyen, S. Joty, M. Imran, H. Sajjad & P. Mitra, "Applications of online deep learning for crisis response using social media information", arXiv preprint arXiv:1610.01030 (2016). https://doi.org/10.48550/arXiv.1610.01030.

[62] S. E. Vieweg, *Situational awareness in mass emergency: a behavioral and linguistic analysis of microblogged communications*, University of Colorado at Boulder, 2012. https://www.proquest.com/openview/540ee2ba902309c5ad7314438e06ea42/1?cbl=18750&pq-origsite=gscholar.