BIG DATA MINING AND ANALYTICS
ISSN 2096-0654 03/05 pp121–130
Volume 3, Number 2, June 2020
DOI: 10.26599/BDMA.2019.9020021

Comparative Study of Statistical Features to Detect the Target Event During Disaster

Madichetty Sreenivasulu* and M. Sridevi

Abstract: Microblogs, such as facebook and twitter, have much attention among the users and organizations. Nowadays, twitter is more popular because of its real-time nature. People often interacted with real-time events such as earthquakes and floods through twitter. During a disaster, the number of posts or tweets is drastically increased in twitter. At the time of the disaster, detecting a target event is a challenging task. In this paper, a framework is proposed for observing the tweets and to detect the target event. For detecting the target event, a classifier is devised based on different combinations of statistical features such as the position of the keyword in a tweet, length of a tweet, the frequency of hashtag, and frequency of user mentions and the URL. From the result, it is evident that the combination of frequency of hashtag and position of keyword features provides good classification results than the other combinations of features. Hence, usage of two features, namely, frequency of hashtag and position of the earthquake keyword reduces the event's detection time. And also these two features are further helpful for detecting the sub-events which are used for filtering the tweets related to the disaster. Additionally, different classifiers such as Artificial Neural Networks (ANN), decision tree, and K-Nearest Neighbor (KNN) are compared by using these two features. However, Support Vector Machine (SVM) with linear kernel by using the combination of position of earthquake keyword and frequency of hashtag outperforms state-of-the-art methods. Therefore, SVM (linear kernel) with proposed features is applied for detecting the earthquake during disaster. The proposed algorithm is tested on Nepal earthquake and landslide datasets, 2015.

Key words: disaster; twitter; Support Vector Machine (SVM); statical features

1 Introduction

Micro-blogging is a communication medium used for exchanging a small amount of information such as short length of text or posts among the users[1–3]. The best examples for microblogs are twitter and facebook, etc. Twitter[4] is the most popular among the users for exchanging information because of its real-time nature. Due to the short length of the text, there is an increase in the attention of users and organizations during disasters[5,6]. Different users post different tweets depending on the situation in a large volume and faster rates during disaster[7]. Many organizations or local government rely on the tweets to understand the situation during disaster[8]. These tweets are categorized into two types, namely, tweets related and not related to the disaster based on the tweet information. The tweets need to be detected related to the disaster which includes injured or dead people, missing, trapped or found people, infrastructure and utilities, shelter and supply volunteer or professional services, and caution, etc.[5] The tweets which are not related to disaster represent information that may include spam tweets[9] irrelevant to the disaster. Therefore, detection of the target event in twitter is a challenging task during disasters.

The location of disaster can be predicted based on

• Madichetty Sreenivasulu and M. Sridevi are with the Department of CSE, National Institute of Technology, Tiruchirappalli, Tamilnadu 620015, India. E-mail: 406116004@nitt.edu; msridevi@nitt.edu.

∗ To whom correspondence should be addressed.
Manuscript received: 2019-11-05; accepted: 2019-11-21

© The author(s) 2020. The articles published in this open access journal are distributed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/).

the tweets[10] and the victims in nearby areas can be alerted by sending emails and Short Message Service (SMS). Analyzing such tweets is very helpful to the responders, organizations, and victims for finding the damaged resources and helping the injured people[8, 11]. Performing this task manually is a tedious process due to a large number of tweets posted at the time of disasters. Hence, there is a need for automatic analysis of tweets in the twitter.

The tweets are differentiated from relevant and irrelevant to the disaster based on some limited set of features. Most prior works[5, 12, 13] detect the situational information via twitter during a disaster with the use of pre-processing techniques and features. In this proposed work, sets of statistical features are considered for detecting the tweets related to the disaster and have been applied to a different subset of features of different combinations. The recognized features are extracted from the tweets and given to classification algorithms for detecting the target event during disasters. The classifier is trained with the different combinations of features over tweets posted at the time of disasters and tested with the same set of features.

The contributions of this proposed work are as follows:

(1) Extraction of the statistical features from the tweets uses less number of pre-processing techniques.

(2) Different combinations of statistical features were compared and the best features for earthquake event detection were concluded.

(3) Comparison of Support Vector Machine (SVM) classifier with linear kernel based on proposed features with other classifiers such as decision tree, K-Nearest Neighbor (KNN), Artificial Neural Network (ANN), and random forest.

(4) Comparison of SVM classifier based on the proposed features with standard baselines for datasets such as Nepal earthquake and landslide dataset, 2015.

The rest of the paper is as follows: The related works are discussed in Section 2. Section 3 discusses about the pre-processing techniques and explains extraction of different features. The datasets, experiment results, and performance analysis of the proposed system with various parameters are presented in Sections 4 and 5. Finally, the paper is concluded in Section 6.

2 Related Work

Several studies[5, 10, 14, 15] have been done by many researchers using the micro-blogging service during disaster time. Detecting the target events on micro-blogs plays a very important role during disasters. Different authors used different types of features for identifying the situational information. In Ref. [14], the authors used the Bag-Of-Words (BOW) as features for classifying the situational information and non-situational information. In Ref. [6], the authors developed Artificial Intelligence for Disaster Response (AIDR) based on the features of uni-gram and bi-gram features for classifying the tweets into user-defined categories, and tested the AIDR for classifying the informative and non-informative tweets which are posted in the twitter during 2013 at the time of Pakistan earthquake and achieved 80% of accuracy. In Ref. [16], the authors used *n*-gram features to classify the tweets into pre-incident, during-incident, and post-incident classes. However, there is a need for pre-processing of data when content features are used for proper utilization of features. It is a time-consuming process. Reducing the time for detecting the target event during disasters is one of the important tasks.

In Ref. [10], the authors used three features such as statistical, keyword, and word context for detecting the event. Statistical features represent the position of the earthquake keyword and length of the tweet. Keyword features represent the words in a tweet and word context features represent words before and after the keyword. Out of three methods, statistical features give the good performance when compared to the other two features. However, it works well only for Japanese tweets and does not perform well in English tweets.

In Ref. [5], the authors used pre-processing techniques such as identification of Out Of Vocabulary (OOV) and normalization of OOV method. For identifying OOV words, there is a need for normalization and initial lexicon dictionaries for checking the misspelled words and slang words. It is time-consuming. Hence, there is a need to develop a method which takes less time and more accurate detection.

In Ref. [15], the authors developed a method for finding help requests in social media during disasters. It uses the content and context features for detecting the help request. Content features are specific keywords which are used by the victims for requesting help and they are translated to uni-grams, bi-grams, and trigrams. Another content feature is extracted from the tweets using Latent Dirichlet Allocation (LDA). Context features are the number of URLs, user mentions, and hashtags. It does not mention the special keywords which are included in the content features and also

does not provide information about the detection of the target event. It has been suggested in Ref. [17] that the mined informative words seem to be reliable for detecting the resources during disaster. Informative words include terms related to disaster, human, injured, location, communication, and infrastructure damage. Resources include both availability and requirement of resources. In Ref. [18], the authors proposed a novel retrieval methodology based on the word embeddings for automatic detection of availability and it needs tweets during disaster. Later in Ref. [19], the authors proposed a neural information retrieval model which includes both word-level and character-level embeddings for identifying the availability and needs tweets during disaster, and it outperforms the pattern match techniques. In Ref. [20], the authors used re-ranking feature selection algorithm for detecting the availability and requirement of resources during the disaster. However, the number of tweets related to the sub-events is less and it takes more time for detection using existing methodologies.

During disaster, both relevant and irrelevant tweets are posted on the twitter. Relevant tweets indicate the tweets related to the sub-event and irrelevant tweets indicate the tweets unrelated to the sub-event (which includes both related and unrelated to the disaster). Among the posted tweets, a very less number of tweets is relevant to the sub-event compared to other tweets. Therefore, detecting the sub-events is a time-consuming process using the existing methods. Hence, there is a need to develop a method which consumes less time. In this paper, a method is developed for detecting tweets related to the disaster in less time and is used as pre-processing technique for detecting the sub-event. While detecting the tweets related to sub-event, the proposed method is used for filtering the tweets related to the disaster. Automatically, the number of tweets used for detecting sub-events is reduced. Therefore, it consumes less time for detection of sub-events by applying proposed method in the pre-processing technique.

3 Classification Algorithm

Machine learning algorithms have been effectively utilized in text classification. Machine learning algorithm can be widely classified as decision trees, rule-based methods, perceptron-based methods, statistical learning methods (such as Bayesian networks and Naive Bayes classifier), instance-based classifiers, and SVM. Decision trees and SVM are widely used for text classification problems[21]. Also, KNN is widely used for its low implementation costs on different classification tasks[22,23]. Hence, KNN, decision tree, ANN, and SVM are considered for experimental evaluations.

3.1 Decision tree

A decision tree is one of the predominant methods used for classification. It is simple to follow and offers interpretable outcomes. In Ref. [24], the authors succeeded in the classification and regression trees for classification problems built on the first regression tree algorithm for automatic interaction detection[25]. Classification And Regression Tree (CART) method uses Gini index for selecting the proper attribute to partition the data. Similarly, Iterative Dichotomiser 3 (ID3) algorithm[26] uses the information gain. C4.5[27] uses the gain ratio instead of information gain for attribute selection. However, it reduces adverse effect which is not solvable by information gain.

3.2 Artificial neural network

The artificial neural network is a supervised machine learning algorithm which is inspired by the functioning of biological neurons in the brain and central nervous system[28,29]. Neural network is broadly used classification algorithm and it is trained by the backpropagation. The backpropagation algorithm was first proposed in Ref. [30] and re-invented in Ref. [31]. In Ref. [32], a single hidden-layer network is typically sufficient to interpret any problem in hand. It is suitable to use, however, the initial weights are assigned in a random manner. It causes dissimilarity in the training process and tests result at each time. Another issue is that training process will consume more time.

3.3 K-nearest neighbor

The KNN classifier has been extensively used in different types of classification tasks[22,23]. This classifier has obtained its popularity on its low implementation cost and high degree of classification effectiveness. The KNN classifier is an instant-based learning algorithm. It employs the nearest distance for deciding the category of the new vector in the training dataset[22]. The KNN classifier requires only a small number of training data points and this causes accessibility of the KNN which makes it better than the other classifiers[33]. The most frequently and extensively used distance function for KNN classifier is the Euclidean distance. It is utilized to compute the distance between the training data points and the new unlabeled data points. The foremost step in the classification phase of the KNN is to calculate the

distance to find the nearest neighbors of the alternative input data point[22].

3.4 Support vector machine

The SVM has shown outstanding performance in the text classification tasks[34]. Subsequently, a variety of approaches have been proposed using a linear kernel[35,36] or non-linear kernel[37,38] to enhance the speed for classifying large-scale datasets. Initially, it was designed for binary classification tasks and later it is generalized to multi-category[39] classification tasks. The main goal of the SVM is to determine the maximum margin of hyperplane for binary classification between the two classes. In case of multi-classification, SVM is used in different ways such as one vs. one, one vs. rest, and many vs. many.

3.5 Random forest

Random forest is an ensemble-based technique primarily on the decision trees. It comprises of various kinds of decision trees that can separately take tweet feature vector as input from the sample and generate the vote for each decision tree classifier[24]. In a decision tree, the training samples are divided into a consecutive purer subset at each recursive phase for recursively increasing decision tree. The nodes are divided into several sub-samples as child nodes. The division is based on the feature that minimizes infant (child) node contaminants encircled by all features. If the node contaminant is below the limit, the splitting method will stop and the respective leaf node label will be allocated to the data item as a class label. Finally, random forest generates the majority of output from the various kinds of decision trees.

4 Proposed Work

The existing solutions[5,6,14,16] take more time for pre-processing and feature extraction processes. The proposed algorithm aims to reduce the time consumption of pre-processing and feature extraction process by applying text normalization and stop-word removal, and extracting less number of valuable features. The tweets collected during disasters are preprocessed to remove stop-words and text normalization. The features are extracted from the preprocessed data. The features are fed to the classifier for detecting the relevant and irrelevant tweet from the target event. The proposed method for earthquake event detection is shown in Algorithm 1.

Algorithm 1 Proposed algorithm for target event detection during disaster

(1) Collection of tweet id from datasets.
(2) Crawling the tweets from tweet id through the twitter Application Program Interface (API).
(3) $X = \{\text{tweet-1, tweet-2, } \ldots, \text{ tweet-}n\}$.
(4) For each tweet x in X,
extract the features F1, F2, F3, F4, and F5 from the tweet X.
Where
F1– position of the keyword in the tweet.
F2– length of the tweet.
F3– frequency of the hashtags.
F4– frequency of the @ (user mentions).
F5– frequency of the URL.
(5) Classify the tweets related or unrelated to the earthquake event using SVM, KNN, ANN, and decision tree.
(6) Repeat Step 5 with different combinations of features.

The block diagram of the proposed system is shown in Fig. 1. The steps involved in the proposed scheme are mentioned as follows:

(1) Data collection.
(2) Preprocessing.
  (a) Text normalization and tokenization.
  (b) Stop-word removal.
(3) Feature extraction.
(4) Classification.

The earthquake data is collected from the twitter through the twitter's API. It contains both situational and non-situational information where situational information indicates event related to the earthquake and non-situational information indicates event irrelevant to the earthquake. The collected data has to be preprocessed to remove stop-words and normalize the text by using stop-word removal and text normalization, respectively. These techniques consume less time when compared to stemming, lemmatization, and searching a word in a dictionary for replacement.

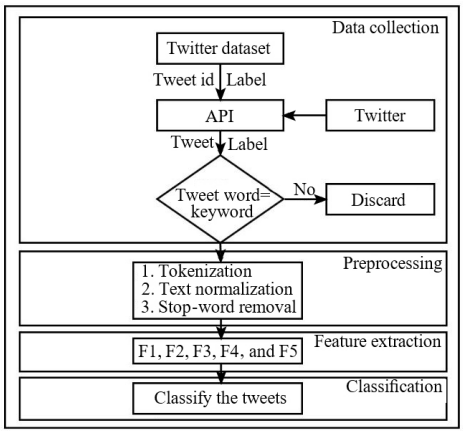

Fig. 1 Overview of the proposed event detection system.

The text normalization process is applied to the collected tweet to convert all tweets into lower case letters, because the same word can appear in a different format, e.g., Earthquake, EarthQuake, and EARTHquake.

The above example shows that the same keyword earthquake presents in a different format. If the text normalization is not applied for above identified keywords, then the keywords such as Earthquake, EarthQuake, and EARTHquake are considered as separate words. Therefore, text normalization is essential for identifying the position of the keyword. After normalization process, tokenization is applied. It is a process of dividing the tweet into multiple words, e.g., Earthquake occurs in the Nepal.

"Earthquake", "occurs", "in", "the", and "Nepal" are the tokens for the example tweet.

The stop-words are the common words which are available in maximum tweets in the twitter. These words do not provide meaning for the tweet while detecting the target events. Hence, they can be removed from the data. The stop-words for the above tweets are "More", "Than", "-", ":", "in", "and", "so", "that", "too", "is", "doing", "a", "by", "the", "of", "with", "you", "all", "me", and "just". These words are removed.

The output of pre-processing is shown in Fig. 2.

```
1.['breaking','earthquake','devasta
tes','nepal','killing', '1300','n-
ew',
'york','times','new','york',
'[url]', '#khaatumo','#khatumo']
2.['150','dead','#nepal','13','#ind-
ia','#earthquake','strong','tremor-
s','felt','across','india','#dis-
astrous']
3.['#nepalearthquake','indian','med-
ia','great','service','reaching',
'unreachable','difficult','times',
'#proudindian','@rahulkanwal','@dib-
ang','@dibang']
4.['#nepalearthquake','reminded',
'@realsportshbo','featured','deadly',
'risks','sherpa','life', '[url]',
'[url]]
```

Fig. 2 Output of pre-processing.

The third step in the proposed scheme is feature extraction. It plays an important role in detecting the earthquake event. In this work, five statistical features are extracted and given as follows:

(1) Position of the keyword in the tweet (F1). It indicates the occurrence of the keyword in the resultant preprocessed tweet.

(2) Length of the tweet (F2). It counts the number of words in a tweet after the removal of stop-words.

(3) Frequency of hashtags (F3). It counts the number of times hashtag occurs in the tweet. Hashtag is a label that precedes the word and it is used for identifying the topic in the tweet.

(4) Frequency of @(user mentions) (F4). It counts the number of times user mentions occur in the tweets. It is denoted by @. It precedes the username which is used for giving a reply to the other users.

(5) Frequency of the URL (F5). It counts the number of times URL presents in the tweet. URL gives additional information in the form of images, videos, etc.

The five features are extracted from the example tweets shown in Table 1. The extracted features are shown in Table 2. After extraction of all statistical features, they are given to the SVM classifiers for classification of tweets into relevant or irrelevant to the earthquake event because it provides good classification results[10].

5 Experiment and Analysis

This section describes the dataset used for experiment and analysis of the proposed algorithms.

Table 1 Example tweets of relevant and irrelevant to earthquake event.

Tweet No.	Tweet	Relevant and irrelevant to earthquake event
1	Breaking: Earthquake devastates Nepal, killing more than 1300 — *New York Times*: New York [URL] #Khaatumo #Khatumo.	Relevant
2	More than 150 dead in #Nepal and 13 in #India #earthquake so strong that tremors felt across India too #disastrous.	Relevant
3	#Nepal Earthquake Indian media is doing great service by reaching the unreachable in difficult times 'STAY STRONG people of Nepal praying you all #Nepal Earthquake #Proud Indian @rahulkanwal @dibang.	Irrelevant
4	#Nepal Earthquake reminded me @RealSport just featured deadly risks of Sherpa life [URL] [URL].	Irrelevant

Table 2 Feature extraction for the sample tweets.

Tweet No.	Position of the keyword	Length of the Tweet	Frequency of hashtags	Frequency of the user mentions	Frequency of the URL
1	2	14	2	0	1
2	6	12	4	0	0
3	1	12	2	2	0
4	1	10	1	1	2

5.1 Dataset

The proposed algorithm is implemented in Python language[40] and it is tested with tweets collected from the Nepal earthquake dataset and landslide dataset during 2015[41]. The dataset contains the earthquake keywords in 1500 tweets, out of which 750 tweets are related to earthquake event and remaining tweets are not related to earthquake event. 1125 (75%) tweets are used for training the SVM model and 375 (25%) tweets are used for testing the proposed algorithm from the total 1500 tweets. And also experiment is conducted for 4098 tweets of Nepal earthquake dataset 2015, out of which 2976 tweets are used for training the SVM model and 1122 tweets are used for testing. It contains different categories of tweets such as caution and advices, injury and damage, missing and trapped people, etc. The different categories of tweets are combined and considered as related to earthquake event. The landslide dataset contains 1310 tweets related to both target and non-target events.

5.2 Performance measure

The combinations of features, namely, F1, F2, F3, F4, and F5 are given to SVM for classification. The performance of the proposed system can be evaluated by three parameters, namely, Precision, Recall, and F1-score. The parameters that are used in the classifiers are shown in Table 3. The other parameters of the classifier which are not mentioned in Table 3 are taken the default values of scikit[40] package.

5.2.1 Precision

It is known as positive predictive value. It is defined as

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{1}$$

where True Positive (TP) represents the number of tweets predicted correctly related to an earthquake event and False Positive (FP) represents the number of tweets predicted wrongly related to earthquake event.

5.2.2 Recall

It is also known as true positive rate or sensitivity. It indicates the percentage of tweets related to the earthquake event. It is calculated using Eq. (2):

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{2}$$

where False Negative (FN) represents the number of tweets predicted wrongly unrelated to earthquake event.

Table 3 Parameter detail of the classifiers.

Serial No.	Classifier	Parameter
1	ANN	(1) Number of layers = 4 (2) Number of hidden layers = 2 (3) Number of neurons in the first hidden layer = 5 (4) Number of neurons in the second hidden layer = 2 (5) Activation function = "relu"
2	SVM	(1) kernel = "linear" (2) Regularization parameter (C) = 1
3	Random forest	(1) Attribute selection criteria = "gini" (2) Maximum depth = 2
4	Decision tree	(1) Attribute selection criteria = "gini" (2) Maximum depth = 2
5	KNN	(1) Number of nearest neighbors (K) = 5 (2) Power parameter (P) = 2

5.2.3 F1-score

A measure that combines Precision and Recall is called as harmonic mean of Precision and Recall or traditional F-measure or balanced F-score. It can be computed using Eq. (3):

$$\text{F1-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

The measures mentioned above are calculated by using two features out of five extracted features in which one feature is kept constant and the second feature is changed. The calculated values for different combinations of features are given in Table 2. It is inferred from Tables 4–6 that the F1 and F3 features give more Recall than the other combinations.

Table 4 Comparison of two different combinations of features for Radial Basis Function (RBF) kernel. (%)

Feature	Recall	Precision	F1-score
F1, F2	72	57	64
F1, F3	**80**	**57**	**67**
F1, F4	57	54	55
F1, F5	68	58	63
F2, F3	71	57	64
F2, F4	79	57	64
F2, F5	56	62	59
F3, F4	53	58	55
F3, F5	63	58	61
F4, F5	59	61	60

Table 5 Comparison of three different combinations of features for RBF kernel.

(%)

Feature	Recall	Precision	F1-score
F1, F2, F3	71	58	63
F1, F2, F4	68	56	62
F1, F2, F5	69	60	64
F1, F3, F4	71	60	65
F1, F3, F5	65	57	61
F1, F4, F5	74	58	65
F2, F3, F4	69	58	63
F2, F3, F5	57	62	59
F3, F4, F5	62	60	61

Table 6 All features used to detect the earthquake event in RBF kernel.

(%)

Feature	Recall	Precision	F1-score
F1, F2, F3, F4, F5	68	59	63

Instead of considering two features, three features are taken at a time and tested with the proposed algorithm, and the results are evaluated and tabulated in Table 5. All features are considered and tested with the proposed algorithm, and results are evaluated and tabulated in Table 6. From the observation of Tables 4–6, Recall is high whenever the position of the keyword and frequency of the hashtag features are used, which indicates high detection of the earthquake event. Low Precision value indicates detection of an irrelevant earthquake event. Whenever the feature of the frequency of user mentions is combined with other features, its performance is less for the detection of earthquake event.

Figure 3 shows the result of different combinations of features. It is observed that F1 and F3 features alone give better recall value. From the analysis, it is found that two extracted features (F1, F3) give more precision than other feature combinations. However, the proposed algorithm detects the earthquake event in a faster manner during disaster time because of usage of less number of pre-processing techniques and features.

The experiment is conducted for a proposed method using SVM with three different kernels, namely, linear, RBF, and Sigmoid for the different combinations of features and the results are shown in Figs. 3 and 4. From the result, it is inferred that the linear SVM kernel gives the best recall value when compared to the other kernels. Hence, it is concluded that the data items for the experiment are linearly separable. In case of precision, all the kernels give the same precision value.

One more test is conducted with 4098 tweets of Nepal earthquake 2015 dataset. From the observations of Tables 7–9, the features with the combination of the frequency of URL give high precision value and features with the combination of all features give the highest precision value. However, they do not give high recall and F1-score value. Therefore, it is concluded that the combination of proposed features such as frequency of hashtags and position of the keyword performs better

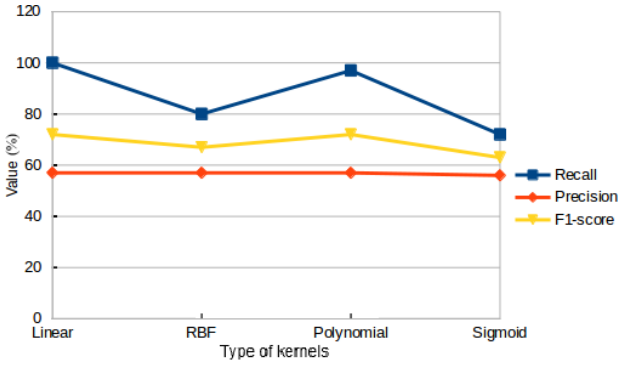

Fig. 4 Comparison of SVM classifier with different kernels.

Table 7 Comparison of two different combinations of features in linear kernel.

(%)

Feature	1500 tweets			4098 tweets		
	Recall	Precision	F1-score	Recall	Precision	F1-score
F1, F2	100	55	71	100	59	74
F1, F3	**100**	**55**	**71**	**100**	**62**	**77**
F1, F4	100	55	71	100	60	75
F1, F5	70	63	67	100	61	76
F2, F3	100	55	71	100	60	75
F2, F4	100	55	71	100	61	75
F2, F5	70	63	67	100	59	74
F3, F4	100	55	71	100	61	76
F3, F5	70	63	67	100	61	76
F4, F5	70	63	67	100	59	74

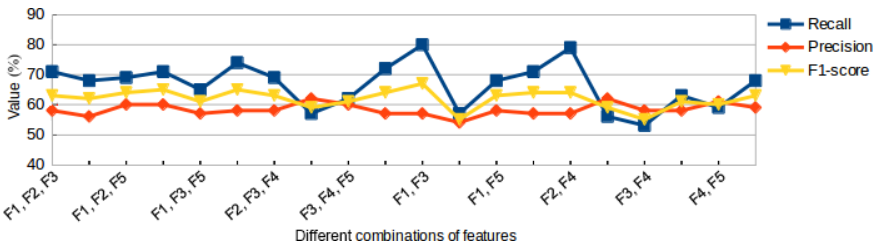

Fig. 3 Overall performance of the classifier with different combinations of features.

Table 8 Comparison of three different combinations of features in linear kernel.

(%)

Feature	1500 tweets			4098 tweets		
	Recall	Precision	F1-score	Recall	Precision	F1-score
F1, F2, F3	100	55	71	100	60	75
F1, F2, F4	100	55	71	100	60	75
F1, F2, F5	70	63	67	71	65	68
F1, F3, F4	100	51	71	100	60	75
F1, F3, F5	70	63	67	71	65	68
F1, F4, F5	70	63	67	71	65	68
F2, F3, F4	100	55	71	100	60	75
F2, F3, F5	70	63	67	71	65	68
F3, F4, F5	70	63	66	71	65	68

Table 9 All features for detecting the earthquake event in linear kernel.

(%)

Feature	1500 tweets			4098 tweets		
	Recall	Precision	F1-score	Recall	Precision	F1-score
F1, F2, F3, F4, F5	70	63	66	71	65	68

in the cases of Recall and F1-score than the other combinations with length of tweet, frequency of user mentions, and frequency of the URL. Later comparison is made between 1500 and 4098 tweets on the parameters such as Recall, Precision, and F1-score. SVM classifier with a linear kernel based on the proposed features also works well for 4098 tweets. The plot is made for the measured metrics in Fig. 5. It is observed that the increase in some observation leads to increase in Precision and F1-score values. SVM classifier (linear kernel) is compared with different classifiers such as KNN, decision tree, ANN, and random forest and it is tabulated in Table 10. SVM classifier (linear kernel) is

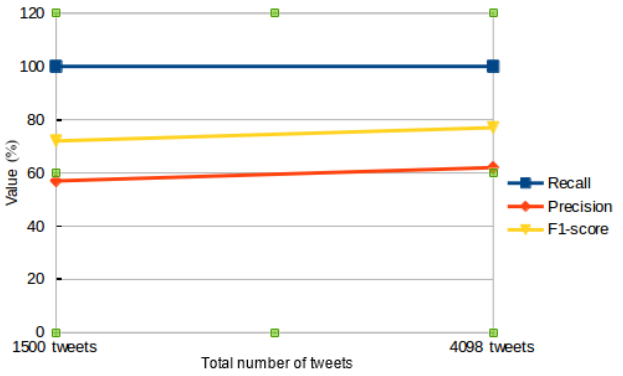

Fig. 5 Comparison of different parameters by varying the number of tweets using the SVM classifier with linear kernel based on the F1 and F3 features.

compared with different state-of-the-art methods such as SVM classifier with uni-gram features similar to baseline-1[6] and baseline-2[10]. It outperforms the state-of-the-art models and the results are tabulated in Table 11 for Nepal earthquake dataset and landslide dataset. The proposed algorithm outperforms the BOW model for two datasets.

6 Conclusion

Users are interacted with the real-time events such as earthquakes and floods through social media. Millions of tweets or messages which are related and unrelated to the disaster are posted on social media during disaster. A scheme is proposed in this paper to detect the tweets related to the disaster using SVM classifier by employing different combinations of statistical features. From the analysis, it is evident that the position of the earthquake keyword and frequency of hashtag features provide better results than the other combinations. Hence, the

Table 10 Comparison of different classification algorithms with proposed features.

(%)

Classifier	1500 tweets			4098 tweets		
	Recall	Precision	F1-score	Recall	Precision	F1-score
K-nearest neighbor	58	57	58	83	58	68
Decision tree	65	57	61	91	59	72
Neural network	89	55	68	100	60	75
SVM	100	55	71	100	62	77
Random forest	92	52	66.4	93	59	72

Table 11 Comparison of proposed method with state-of-the-art methods such as baseline-1[6] and baseline-2[10] on different disaster datasets.

(%)

Dataset	Recall			Precision			F1-score		
	Baseline-1	Baseline-2	Proposed method	Baseline-1	Baseline-2	Proposed method	Baseline-1	Baseline-2	Proposed method
Nepal earthquake 2015	75	90	**100**	69	61	62	72	72.71	**77**
Landslide 2015	65	80	**97**	63	53	54	64	56.81	**69**

time taken for detecting the tweets related to the disaster is reduced by using two features. The results suggest that the SVM classifier with proposed features outperforms the BOW model. The proposed technique can be applied in pre-processing stage of detecting the sub-events such as infrastructure damage, resource detection, helping requests, and so on. In future, the proposed features can be applied to detect different disasters in different languages using different methods to check the scalability and compatibility of the algorithm. And it can also be used in cross-domain (training with one disaster event and testing with another disaster event) applications.

References

[1] Z. C. Miao, K. Chen, Y. Fang, J. H. He, Y. Zhou, W. J. Zhang, and H. Y. Zha, Cost-effective online trending topic detection and popularity prediction in microblogging, *ACM Trans. Inf. Syst.*, vol. 35, no. 3, p. 18, 2017.

[2] N. Pervin, F. Fang, A. Datta, K. Dutta, and D. Vandermeer, Fast, scalable, and context-sensitive detection of trending topics in microblog post streams, *ACM Trans. Manage. Inf. Syst.*, vol. 3, no. 4, p. 19, 2013.

[3] M. Sreenivasulu and M. Sridevi, A survey on event detection methods on various social media, in *Recent Findings in Intelligent Computing Techniques*, P. K. Sa, S. Bakshi, I. K. Hatzilygeroudis, and M. N. Sahoo, eds. Singapore: Springer, 2018, pp. 87–93.

[4] H. Kwak, C. Lee, H. Park, and S. Moon, What is twitter, a social network or a news media? in *Proc. 19^{th} Int. Conf. World Wide Web*, Raleigh, NC, USA, 2010, pp. 591–600.

[5] M. Imran, P. Mitra, and C. Castillo, Twitter as a lifeline: Human-annotated twitter corpora for NLP of crisis-related messages, arXiv preprint arXiv: 1605.05894, 2016.

[6] M. Imran, C. Castillo, J. Lucas, P. Meier, and S. Vieweg, AIDR: Artificial intelligence for disaster response, in *Proceedings of the 23rd International Conference on World Wide Web*, Seoul, Korea, 2014, pp. 159–162.

[7] M. Imran, C. Castillo, F. Diaz, and S. Vieweg, Processing social media messages in mass emergency: A survey, *ACM Comput. Surv.*, vol. 47, no. 4, p. 67, 2015.

[8] S. Vieweg, C. Castillo, and M. Imran, Integrating social media communications into the rapid assessment of sudden onset disasters, in *Proc. 6^{th} Int. Conf. Social Informatics*, Barcelona, Spain, 2014, pp. 444–461.

[9] S. Madisetty and M. S. Desarkar, A neural network-based ensemble approach for spam detection in Twittern, *IEEE Trans. Comput. Social Syst.*, vol. 5, no. 4, pp. 973–984, 2018.

[10] T. Sakaki, M. Okazaki, and Y. Matsuo, Tweet analysis for real-time event detection and earthquake reporting system development, *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 4, pp. 919–931, 2013.

[11] B. Takahashi, E. C. Jr. Tandoc, and C. Carmichael, Communicating on twitter during a disaster: An analysis of tweets during Typhoon Haiyan in the Philippines, *Comput. Human Behav.*, vol. 50, pp. 392–398, 2015.

[12] K. Rudra, S. Banerjee, N. Ganguly, P. Goyal, M. Imran, and P. Mitra, Summarizing situational tweets in crisis scenario, in *Proc. 27^{th} ACM Conf. Hypertext and Social Media*, Halifax, Canada, 2016, pp. 137–147.

[13] K. Rudra, S. Ghosh, N. Ganguly, P. Goyal, and S. Ghosh, Extracting situational information from microblogs during disaster events: A classification-summarization approach, in *Proc. 24^{th} ACM Int. Conf. Information and Knowledge Management*, Melbourne, Australia, 2015, pp. 583–592.

[14] S. Verma, S. Vieweg, W. J. Corvey, L. Palen, J. H. Martin, M. Palmer, A. Schram, and K. M. Anderson, Natural language processing to the rescue? extracting "situational awareness" tweets during mass emergency, in *Proc. 5^{th} Int. Conf. Weblogs and Social Media*, Barcelona, Spain, 2011, pp. 385–392.

[15] T. H. Nazer, F. Morstatter, H. Dani, and H. Liu, Finding requests in social media for disaster relief, in *Proc. 2016 IEEE/ACM Int. Conf. Advances in Social Networks Analysis and Mining*, Davis, CA, USA, 2016, pp. 1410–1413.

[16] S. R. Chowdhury, M. Imran, M. R. Asghar, S. Amer-Yahia, and C. Castillo, Tweet4act: Using incident-specific profiles for classifying crisis-related messages, in *Proc. 10^{th} Int. ISCRAM Conf.*, Baden-Baden, Germany, 2013, pp. 1–5.

[17] M. Sreenivasulu and M. Sridevi, Mining informative words from the tweets for detecting the resources during disaster, in *Proc. 5^{th} Int. Conf. Mining Intelligence and Knowledge Exploration*, Hyderabad, India, 2017, pp. 348–358.

[18] M. Basu, K. Ghosh, S. Das, R. Dey, S. Bandyopadhyay, and S. Ghosh, Identifying post-disaster resource needs and availabilities from microblogs, in *Proc. 2017 IEEE/ACM Int. Conf. Advances in Social Networks Analysis and Mining*, Sydney, Australia, 2017, pp. 427–430.

[19] P. Khosla, M. Basu, K. Ghosh, and S. Ghosh, Microblog retrieval for post-disaster relief: Applying and comparing neural IR models, arXiv preprint arXiv: 1707.06112, 2017.

[20] M. Sreenivasulu and M. Sridevi, Re-ranking feature selection algorithm for detecting the availability and requirement of resources tweets during disaster, *International Journal of Computational Intelligence & IoT*, vol. 1, no. 2, pp. 207–211, 2018.

[21] M. Ikonomakis, S. Kotsiantis, and V. Tampakas, Text classification using machine learning techniques, *WSEAS Trans. Comput.*, vol. 4, no. 8, pp. 966–974, 2005.

[22] E. H. Han, G. Karypis, and V. Kumar, Text categorization using weight adjusted k-nearest neighbor classification, in *Proc. 5^{th} Pacific-Asia Conf. Knowledge Discovery and Data Mining*, Hong Kong, China, 2001, pp. 53–65.

[23] J. He, A. H. Tan, and C. L. Tan, On machine learning methods for Chinese document categorization, *Appl. Intell.*, vol. 18, no. 3, pp. 311–322, 2003.

[24] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. Belmont, CA, USA: Wadsworth, 1984.

[25] J. N. Morgan and J. A. Sonquist, Problems in the analysis of survey data, and a proposal, *J. Am. Stat. Assoc.*, vol. 58, no. 302, pp. 415–434, 1963.

[26] J. R. Quinlan, Induction of decision trees, *Mach. Learn.*, vol. 1, no. 1, pp. 81–106, 1986.

[27] J. R. Quinlan, *C4.5: Programs for Machine Learning*. Amsterdam, Netherlands: Elsevier, 2014.

[28] W. S. McCulloch and W. Pitts, A logical calculus of the ideas immanent in nervous activity, *Bull. Math. Biophys.*, vol. 5, no. 4, pp. 115–133, 1943.

[29] F. Rosenblatt, The perceptron: A probabilistic model for information storage and organization in the brain, *Psychol. Rev.*, vol. 65, no. 6, pp. 386–408, 1958.

[30] P. Werbos, Beyond regression: New tools for prediction and analysis in the behavior science, Ph.D. dissertation, Harvard University, Cambridge, MA, USA, 1974.

[31] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, Learning internal representations by error propagation, Technical report, University of California, San Diego, CA, USA, 1985.

[32] K. Hornik, M. Stinchcombe, and H. White, Multilayer feedforward networks are universal approximators, *Neural Netw.*, vol. 2, no. 5, pp. 359–366, 1989.

[33] R. Gutierrez-Osuna, *CS 790: Selected Topics in Computer Science: Introduction to Pattern Recognition*. Dayton, OH, USA: Wright State University, 2002.

[34] T. Joachims, Text categorization with support vector machines: Learning with many relevant features, in *Proc. 10^{th} European Conf. Machine Learning*, Chemnitz, Germany, 1998, pp. 137–142.

[35] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter, Pegasos: Primal estimated sub-gradient solver for SVM, *Mathematical Programming*, vol. 127, no. 1, pp. 3–30, 2011.

[36] C. J. Hsieh, K. W. Chang, C. J. Lin, S. S. Keerthi, and S. Sundararajan, A dual coordinate descent method for large-scale linear SVM, in *Proc. 25^{th} Int. Conf. Machine Learning*, Helsinki, Finland, 2008, pp. 408–415.

[37] I. W. Tsang, J. T. Kwok, and P. M. Cheung, Core vector machines: Fast SVM training on very large data sets, *J. Mach. Learn. Res.*, vol. 6, pp. 363–392, 2005.

[38] A. Rahimi and B. Recht, Random features for large-scale kernel machines, in *Proc. 20^{th} Int. Conf. Neural Information Processing Systems*, Vancouver, Canada, 2008, pp. 1177–1184.

[39] C. W. Hsu and C. J. Lin, A comparison of methods for multiclass support vector machines, *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 415–425, 2002.

[40] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Scikit-learn: Machine learning in Python, *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[41] M. Imran, P. Mitra, and C. Castillo, Twitter as a lifeline: Human-annotated twitter corpora for NLP of crisis-related messages, in *Proc. 10^{th} Int. Conf. Language Resources and Evaluation*, Paris, France, 2016.

Madichetty Sreenivasulu received the BTech degree and the MTech degree in computer science and engineering from the Jawaharlal Nehru Technological University, Ananthapur, India in 2013 and 2015, respetively. He is currently pursuing the PhD in computer science and engineering at the National Institute of Technology, Tiruchirappalli, India. He worked as a lecturer in CSE Department, IIIT RK Valley and SVNE, Tirupathi, Andhra Pradesh from 2015 to 2016. His research interests include information retrieval, social media analysis, natural language processing, and machine learning.

M. Sridevi received the PhD degree from National Institute of Technology, Tiruchirappalli in 2015 and the ME degree in computer science and engineering from Annamalai University, India in 2006. She is an assistant professor in the Department of Computer Science and Engineering, National Institute of Technology, Tiruchirappalli, and has 13 years of teaching and research experience. Her research interests include image processing, parallel algorithms, and soft computing techniques.