2019

Analyzing Disaster-related Twitter Data and Identifying Panic Triggers for Cyber Disruption Prevention and Emergency Response Enhancement

Nasser A. Assery
North Carolina Agricultural and Technical State University

Analyzing Disaster-related Twitter Data and Identifying Panic Triggers for Cyber Disruption Prevention and Emergency Response Enhancement

Nasser A Assery

North Carolina A&T State University

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Department: Computer Science

Major: Computer Science

Major Professor: Dr. Xiaohong Yuan

Greensboro, North Carolina

2019

The Graduate College
North Carolina Agricultural and Technical State University

This is to certify that the Doctoral Dissertation of

Nasser A Assery

has met the dissertation requirements of
North Carolina Agricultural and Technical State University

Greensboro, North Carolina
2019

Approved by:

Dr. Xiaohong Yuan
Major Professor

Dr. Jung Hee Kim
Committee Member

Dr. Xiuli Qu
Committee Member

Dr. Kaushik Roy
Committee Member

Dr. Xiaohong Yuan
Department Chair

Dr. Yang Li
Committee Member

Dr. Clay S. Gloster, Jr.
Interim Dean, The Graduate College

Biographical Sketch

Nasser Assery is a PhD candidate within the Computer Science program at the North Carolina A&T State University. He received master's degree in Web Development and Security from Royal Melbourne Institute of Technology, Australia, 2012 receiving an outstanding student award. Also, He received his bachelor's degree in Computer Science and Education from King Khaled University, Saudi Arabia, in 2004 with a first-class of honors.

Dedication

I dedicate my dissertation work to my family and many friends. A special feeling of gratitude to my loving parents, Zahra Assery and Ahmad Assery whose words of encouragement and push for tenacity ring in my ears. My sisters and my brothers have never left my side and are very special. I also dedicate this dissertation to my special friend and my colleague Sultan Al Malki who has been a great support and always there when I am in need. I dedicate this work and give special thanks to my friend Sultan Al Asmari for dedicating the time to help me throughout the entire doctorate program. I also would like to dedicate it to my friends who have supported me, and I will always appreciate all they have done.

Acknowledgments

I would like to acknowledge my indebtedness and render my warmest thanks to my supervisor and my committee chairman, Professor Dr. Xiaohong Yuan, who made this work possible. Her friendly guidance and expert advice have been invaluable throughout all stages of the work. I would also wish to express my gratitude to her for the discussions and valuable suggestions which have contributed greatly to the improvement of the work. Thank you, Dr. Yuan, for your countless hours of reflecting, reading, encouraging, and most of all patience throughout the entire process.

I wish to thank my committee members who were more than generous with their expertise and precious time. Thank you, Dr. Kaushik Roy, Dr. Xiuli Qu, Dr. Jung Hee Kim, Dr. Kossi Edoh, and Dr. Li Yang for agreeing to serve on my committee. Each of the members of my Dissertation Committee has provided me extensive personal and professional guidance and taught me a great deal about scientific research. I would like to acknowledge and thank my school division for allowing me to conduct my research and providing any assistance requested.

I am grateful to all of those with whom I have had the pleasure to work with during this and other related projects. I would like to thank the beginning teachers, mentor-teachers and administrators in our school division that assisted me with this project. Their excitement and willingness to provide feedback made the completion of this research an enjoyable experience. Finally, I thank the (yet to come) readers of this dissertation for their interest in my work. I sincerely hope that you can benefit from it.

Table of Contents

List of Figures

List of Tables

Abstract

Twitter has emerged rapidly as an ideal platform for news updates especially during natural disasters. During disasters, people exchange endless amount of information on Twitter. This information may include warnings, evacuation orders, updates etc. Making sense of this information is challenging due to the limitations of available tools to analyze high-volume data. There have been studies done to make use of twitter data as it contains valuable information that has the potential to help improve the efficiency of disaster response. This research presents a framework to extract, automatically label, and classify tweets from two recent disaster events in order to make sense of the data, identify disaster-related tweets, and evaluate their credibility. The framework also includes classifying tweets into disaster-related or not disaster-related, and credible or not credible using learning-based methods.

Many risk-factors associated with panic disorder occur amongst the public during natural disaster. This research presents a panic trigger identification framework to detect triggers that form cyber disruption threats in hurricane disaster data, and reports to emergency responders to mitigate such threats. The results of this research show that automated labeling can be sufficient for labeling tweets in accordance with relevance to the disaster and tweet credibility. For disaster relevance classification, using CountVectorizer word vectorizer has produced features that led to higher accuracies (98% on average) especially when using Decision Tree and Random Forest models. For classifying tweets in terms of credibility, Random Forest and Decision Tree models have given the best predictions with high accuracies (96% on average). For classifying tweets in terms of panic triggers, Random Forest and Decision Tree have given the best predictions with high accuracies (95% on average) when using CountVectorizer features. The contributions of this research include: (1) Two datasets of tweets on hurricanes were collected which will be made

available for future researchers; (2) An automated labeling framework were developed to label disaster tweets into disaster-related and not-disaster-related using dictionary-based technique, and credible and not credible using user-based and content-based features; and (3) A panic trigger detection framework was developed to improve emergency response.

CHAPTER 1

Introduction

The impact of social media platforms, like Twitter, has significantly increased over the past decade. It has significantly supplemented if not replaced more traditional means of communication in many areas of the U.S. (Tracie, 2015). Twitter is one of the world's leading social media platforms with 330 million monthly active users (Clement, 2019). During a disaster and emergencies, people increasingly use microblogging platforms like Twitter during (Abbasi et al., 2013; Alam et al., 2018). This created numerous opportunities to disseminate time-critical information in the form of images, videos and textual messages during disasters and emergencies (Abbasi et al., 2013; Alam et al., 2018; Imran et al., 2018; Starbrid et al., 2010; Vieweg et al., 2010). Twitter postings are useful for a number of crisis response and management tasks such as gaining insights into the situation, identifying urgent needs of the impacted communities, and assessing the severity of damage (Castillo et al., 2016). Hence, many formal disaster response or emergency management organizations have become more interested in finding ways to quickly and easily locate and organize the information that is most useful, which can help track natural disasters in real time and alert first responders to areas that need urgent aid (Vieweg et al., 2014; Kathleen, 2018).

Twitter has become an effective platform for crowdsourcing and spreading critical information; however, making sense of Twitter data is a challenging task due to the lack of tools available to analyze high-volume and high-velocity data streams (Palen et al., 2010; Collins et al., 2016; Das et al., 2016). Therefore, there has been research studies done to make use of disaster Twitter data as it contains information about the disaster, expected damages, resource allocation, evacuation strategies, warning of unsafe areas or situations, notification of someone's safety,

updates about the event and fundraising for disaster relief. Responders use this information to bring the emergency under control and save lives and properties. Responders range from police, fire, and emergency health and weather personnel, to community volunteers. Individuals on the other hand use the information from social media to assess the severity of the hurricane and prepare for evacuation when needed.

On Twitter, people disseminate not only true information but also false information unintentionally (Castillo et al., 2011; Gupta et al., 2012). Thus, several research studies have been conducted for assessing the credibility of information or for detecting false information in a micro-blog, (Castillo et al., 2011; Kawabe et al., 2015; Wassmer et al., 2005), to provide emergency responders with the credibility level of the tweet with critical information that may lead the public to take undesired actions.

Since Twitter has gained a wide adoption over the years as a prominent news source, often disseminating information faster than traditional news media, it plays an important role during crises, provides valuable information to emergency responders and the public, helps reaching out to people in need, and assists in the coordination of relief efforts (Gupta et al., 2014). Using and analyzing Twitter disaster data to categorize and extract credible predictions of the data would provide responders and the public with the best and most reliable knowledge about a certain disaster, and help improve the speed, efficiency, and quality of disaster response.

1.1 Research Problem

During a disaster, there is usually a high volume of posts spread across Twitter. It has been a challenge to identify which of these posts are actually related to the disaster since some users take advantage of the disaster events and promote their products. For example, some people use hashtags like ‘#HurricaneMichael’ in their posts to attract people to see their posts which were not

relevant to the hurricane as shown in Figure 1. Therefore, it is important to filter the collected tweets and eliminate non-related tweets since they can cause a distraction for emergency responders and disaster analysts as they spread abundantly during natural disasters.

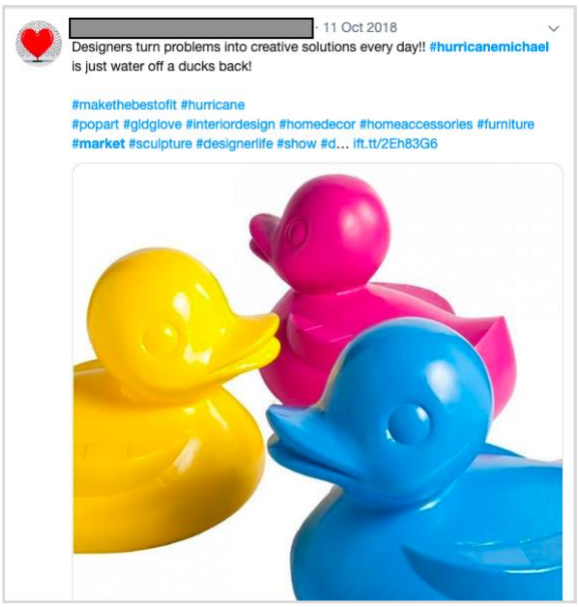

Figure 1. An example of a marketing post that is not related to a hurricane disaster.

Most studies worked on categorizing tweets into disaster-related and not-related have used manual labeling for the data, in which they hired experts to categorize the tweets based on predefined criteria (Huang et al., 2015; Xia et al., 2012; Ito et al., 2015).

When disasters occur, people increasingly post on Twitter about the disaster to learn about the appropriate strategies and to inform their decisions. The problem is that rumors can spread faster on Twitter platform. Some of these rumors can have detrimental consequences for public safety. For example, after both hurricanes Harvey and Irma, false information was spread over social media that immigration status would be checked at evacuation shelters. Rumors like this could affect evacuation decision-making and put both local residents and emergency responders at greater risk. The general public is not very good at differentiating the truth from rumors related to disasters. The public tends to spread rumors without verifying them (Dambroski, 2018).

Therefore, there is the need for tools that analyze the credibility of disaster related data. A few research studies have analyzed Twitter data using manual analysis methods according to predefined credibility criteria (Xia et al., 2012; Ito et al., 2015). Manually labeling large databases in any domain is costly and time-consuming (Schreiner et al., 2006; Conaire et al., 2007; Lu et al., 2019). Therefore, there is need for automated tools to analyze the credibility of disaster-related Twitter data.

Sometimes the information disseminated on Twitter contains triggers and indicators of evacuations that could lead people to panic (Ross et al., 2016), affecting their response and evacuation behavior. For example, Twitter can contain hashtags like "#evacuate", "hurricaneEvacution", "#hurricanePrep", #findSheters and "#flood", or contents that require the public to take evacuation actions. In order to avoid panics, these indicators need to be detected, the credibility of their source needs to be validated, and the emergency responders need to mitigate the risk of panic by providing optimal strategies to handle such situations.

1.2 Summary and Contribution

This work presents a framework to collect tweets related to Hurricane Florence and Hurricane Michael from Twitter API and generates datasets. The datasets created will be made available for researchers who seek to investigate different aspects of these disaster events. Compared with related work, this work presents a labeling framework which automatically labels tweets collected during hurricane disasters into whether a tweet is disaster-related or not disaster-related. This labeling framework could be used to label tweets on future hurricane disasters according to their relevance to the disaster and would speed up the annotation process which could be time consuming and costly when done manually (Conaire et al., 2007; Liu et al., 2010). To generate features for tweet classification, the framework implements tweet classification using

TfidfVectorizer and CountVectorizer features in order to determine which of these word vectorizers would provide better features for classifiers that would produce more accurate classification. Further, for the learning-based system, the framework measures the performance of each classifier from the aspects of accuracy, precision, recall, and f-score. Then a comparison between supervised machine learning classifiers in order to gauge which classification models can produce most accurate predictions.

Moreover, this work presents disaster-tweet credibility evaluation based on user features and tweet content features. For each tweet, attributes like text messages and associated URLs, number of user followers, number of likes, hashtags, etc., were extracted in order to measure the credibility and trustworthiness of disaster-related tweets. A 10-point scoring system is proposed to determine the level of tweet credibility by calculating scores based on the user and tweet features and assigning credibility labels based on the acquired score for each tweet. Based on the features analyzed and the credibility labels, the framework implements supervised machine learning to evaluate the credibility predictions and model performance and conducts a comparison between the classifiers. Finally, this work presents Panic Trigger Identification Framework (PTIF) which is a framework that is able to analyze disaster-related tweets to detect panic triggers, and then classify the tweets based on the triggers identified and the corresponding credibility level for the tweet assigning panic response labels. The framework implements a machine learning classification for the triggered tweets using two kinds of texts vectorizers: CountVectorizer, and TfidfVectorizer to produce features for the classification models used in the experiment, and then conducts a performance comparison between the models in which the accuracies and error rates generated by the classifiers have been analyzed.

The main contributions of this dissertation are as follows:

- Collected and generated hurricane tweets datasets for two recent disaster that will be available for future researchers.
- Developed an automated framework to annotate and classify disaster tweets.
- Developed an automated framework to assess tweet credibility based on user-based features and content-based features.
- Developed an automated panic trigger detection framework to improve the emergency response, and to suggest mitigation strategies for emergency management.

The rest of this dissertation is organized as follows: Chapter 2 discusses what has already been done in the field of the emergency management enhancement and cyber disruption mitigation. Specifically, Chapter 2 covers areas of Twitter disaster-related data classification, tweet credibility analysis, and panic trigger and indicator analysis. Chapter 3 describes in detail the methodology used in the study, presenting the strategies and techniques utilized throughout each iteration of the project. Chapter 4 presents the results of experiment conducted in this project. Finally, Chapter 5 gives a summary of this dissertation as well as discussing directions for future work.

CHAPTER 2

Literature Review

This section presents some backgrounds to the readers in order to put them in the right context by presenting a brief overview of the previous research on Twitter disaster data and what approaches had been pursued to make use of the data. Also, it provides an overview which serves as a baseline description for the different methods used throughout this project. The first part is about the classification and making sense of Twitter disaster-related data and discovering valuable knowledge. The second part reviews the approaches taken to study and evaluate the data credibility. The last part presents a background of the literature on investigating Twitter disaster data to detect panic triggers and indicators to contribute into emergency enhancement which will be concerned by this research work.

2.1 Disaster-related Tweet Classification

During a natural disaster, a large number of messages are often posted on Twitter (Khare et al., 2017). A good percentage of tweets posted about a disaster tend to be irrelevant and unrelated. Some people use current hot trends and events to attract Twitter users to their posts or accounts without any intention to provide informative knowledge or helpful strategies regarding the disaster event occurring (Stowe et al., 2018). Olteanu et al. (2015) mentioned that natural disaster reporting could be classified into three main categories: a) related and informative, b) related but not informative, and c) not related. In this paper, we focus on the identification of disaster-related tweets only, since identifying informative tweets in disaster scenario is a complex task that requires a deeper investigation of the meaning of the information and its dimensions such as the freshness, location, novelty, and the scope of the disaster (Khare et al., 2017).

Supervised machine learning approaches have been used in Disaster-related tweets identification. The majority of supervised machine learning approaches used in this domain depend on linguistic and other statistical attributes that exist within a tweet like part of speech, user mentions, length of the tweet, and number of hashtags etc. (Ghosh et al., 2018; Khare et al., 2018). These supervised machine learning approaches include traditional classification methods such as Logistic Regression, Decision Tree, Support Vector Machines (SVM), Naive Bayes, Conditional Random Fields, etc. (Stowe et al., 2016; Imran et al., 2013; Imran et al., 2014; To et al., 2017; Castillo et al., 2014).

Huang et al. (2015) presented a coding schema for categorizing Twitter messages into different themes according to different disaster stages. They collected tweets about Hurricane Sandy and filtered out the messages that were irrelevant to the disaster. They extracted all hashtags related to Hurricane Sandy from the collected data such as “breakingstorm”, “superstorms”, “hurricanesandyproblems”, and “njpower”. If a tweet did not contain any predefined keywords in either the message or hashtag, it would be excluded from tweets relevant to Hurricane Sandy. Once the relevant tweets were obtained, they manually sampled 2000 relevant tweets, and examined the characteristics and manually annotated tweets into different themes (mitigation, preparedness, emergency response, and recovery) for each tweet. They used several classification algorithms including K-nearest neighbors (KNN), Naïve Bayes, and logistic regression, and performed Ten-fold cross-validation to test the classifier.

Keyword matching is proven to be a reliable technique to filter out disaster related tweets and a faster way to label disaster tweets as disaster related or not disaster related. It has also been used to categorize tweets according the tweet contents and informativeness (To et al., 2017; Ashktorab et al., 2014; De Albuquerque et al., 2015; Verma et al., 2011). Guan (2014) used a

combination of predefined keywords and hashtags to identify and categorize disaster-related tweets. They used Twitter data on Hurricane Sandy and demonstrated the temporal–spatial patterns of Twitter activities particularly near coastal areas and in large urban areas to explore the relationship between hurricane damages and Twitter activities.

To et al. (2017) used keyword and hashtag matching for identifying relevant messages from social media streams. The authors compared the keyword-matching approach against a learning-based system. Their analysis includes five steps: (a) removing spam from the data, (b) mapping the data to affected and unaffected regions, (c) filtering of irrelevant tweets, (d) tweet sentiment analysis and finally (e) data visualization. They used three types of natural disasters (floods, earthquakes and wildfires) for their comparative study. Their results show that the learning-based technique collected a higher number of relevant tweets compared to the matching-based classification, while matching-based classification collected higher quality relevant tweets (which include higher percentage of disaster related tweets).

Imran et al. (2017) proposed a platform, called AIRD, for automatic detection and classification of disaster-related tweets during disaster events. The system utilizes human intelligence and machine learning for the analysis of large dataset at high speed. The system is able to continuously retrieve disaster-related information. Classification categories were defined through crowdsourcing. The system was tested on an earthquake event in Pakistan in 2013.

Ashktorab et al. (2014), developed a Twitter mining tool called “Tweedr” which extracts useful information from tweets to assist disaster relief workers during disasters. The authors used a combination of classification, clustering and extraction techniques. The classification was used to identify tweets reporting damage or casualties, and clustering was used to merge the tweets that are related to similar events. During the extraction phase, tokens and phrases report useful

information about different classes, such as infrastructure damage, damage types and casualties were used. They used several classification algorithms including sLDA, SVM, and logistic regression, and found Logistic Regression to be the most reliable on several evaluation metrics. For extraction, Conditional Random Fields (CRFs) with several different types of features were used. CRFs are a type of Discriminative classifier and they model the decision boundary between the different classes and used for predicting sequences. They use contextual information from previous labels (Zheng et al., 2015). For clustering, bloom filters (Gupta et al., 2011), and SimHash algorithms (Breitinger et al., 2013) were used. Tweedr was evaluated using data on twelve natural disaster events occurred in North America since 2006. It was shown that Tweedr performed well on predicting several categories such as missing persons, health and hospital infrastructures and electricity loss, etc.

2.2 Tweet Credibility Analysis and Classification

Evaluating the credibility of information is an important part of research on social media. Research on credibility concentrated on source credibility as well as credibility attributed to different media channels (Hovland et al., 1951). In social media, the credibility of the source has a significant effect on the process of acquiring the content and manipulation of the public attitudes, beliefs and reactions (Petty, 2018). As more people rely on social media, especially Twitter, to seek information regarding disasters, Twitter becomes more susceptible to be used to disseminate misinformation and rumors. Therefore, users have the challenge of distinguishing which piece of information is credible. They also need to find ways to assess the credibility of information. This problem becomes critical when the source of information is not known to the user. There have been many research efforts to analyze Twitter data credibility using user behavior analysis.

Castillo et al. (2011) discussed the information credibility of news propagated through Twitter. They used users' profile information and users' behavior to assess the credibility of tweets. They used features from users' posting behavior (tweet and retweet), text, and the network (# of friends and # of followers) to distinguish credible from not credible tweets. They achieved a precision and recall of 70-80% using a decision-tree based algorithm.

Gupta et al. (2012) analyzed the credibility of information in tweets of fourteen high impact news events of 2011 worldwide. Among these events there were 90,237 tweets related to Hurricane Irene. They were able to identify important content-based and user-based features for predicting the credibility of information in a tweet using logistic linear regression. The content-based features were the number of unique characters, swear words, pronouns, and emoticons in a tweet, and user-based features were the number of followers and length of username. The researchers manually labelled the event-related tweets to obtain the ground truth regarding the presence of credible information. The labels included "Definitely Credible", "Seems Credible", "Definitely Incredible", and "I can't Decide". Then they computed the Cronbach Alpha score, which is a tool for measuring internal consistency, i.e., to check the reliability of results obtained by annotators through inter-annotator agreement scoring. Then they selected the majority score for a tweet and discarded all tweets for which all three annotators gave different agreement scores. Then they proposed an automated ranking scheme to output of tweets ordered according to the credibility of information provided in them. They used a combination of supervised machine learning and relevance feedback approach to rank tweets according to information quality in the tweet. They used Ranking SVM algorithm which is an extension of SVM classifier (Joachims, 2002) to build a model for credibility of information in tweets. They succeeded to enhance the performance of their ranking algorithm and showed that extraction of credible information from Twitter can be

automated with high confidence. Based on their analysis, they found that on average 30% of total tweets posted about an event contained information about the event. Fourteen percent was spam and only seventeen percent of the total tweets posted was credible.

Ross et al. (2016) created a general feature set for learning to rank tweets based on credibility and newsworthiness. They gathered features from previous studies that used classifiers to automatically predict credibility. The features included the number of retweets, tweet length, the number of user mentions, the number of URLs, tweet has a URL, etc. They used these features as a starting point for their own feature set. Then, they added two new features to capture when the sentiment of a tweet matches the overall sentiment of the topic it was included in, "differenceFromMeanPositive" and "differenceFromMeanNegative". The tweets that have similar sentiment to the rest of the tweets in the topic will be considered credible. However, if a tweet does not have the same sentiment as the topic overall, it is considered as non-credible or not newsworthy tweet. The researchers ranked the entire feature set for each dataset by listing the ranked features in order from best to worst in order to decide credibility level for each tweet using LibSVM extension, which is a Support Vector Machine library for SVM classification and uses feature selection methods and F-Score to rank the features (Chen et al., 2006).

Ito et al. (2015) collected trendy tweets posted in Japan in April 2014 and hired annotators who were widely distributed by age and sex, to label the credibility of every tweet collected. The labeling depended on four aspects: a) whether the tweet contained opinions or impressions, b) URLs in the tweet, c) credibility of the tweet, and d) the reasons why the annotator thought the tweet was or was not credible. They found that the most important factor in deciding the tweet's credibility was whether a tweet had an information source, whether the topic of a tweet was serious, and whether the user of a tweet is reliable. Moreover, they found that utilizing the "tweet topic"

and “user topic” features obtained from the LDA model were effective when the topic size in the model is appropriate, and the performance was enhanced by using them to recognize reliable trendy topics and users.

Gupta et al. (2014) built TweetCred which is a real-time, web-based system to assess the credibility of content on Twitter. While the system does not determine the truth of stories, it provided a credibility rating. It could be used effectively by emergency responders, firefighters, journalists and general users to determine credibility of Twitter content. They demonstrated that measuring the credibility of Twitter content using automated techniques was possible, and the results are valuable for end users. They collected data about many disaster events such as: Typhoon Haiyan in the Philippines, Cyclone Phailin in India, etc. The system provided a rating from 1 (low credibility) to 7 (high credibility) for each tweet on a user’s Twitter timeline. The score was computed using a supervised automated ranking algorithm, trained manually labeled data obtained using crowdsourcing. They used 45 features including the number of unique characters, the number of retweets, and the ratio friends/followers of the author. They evaluated the performance of TweetCred in terms of response time, effectiveness and usability. They observed that 80% of the credibility scores were computed and displayed within 6 seconds, and that 63% of users either agreed with the automatically generated scores or disagreed by 1 or 2 points (on a scale from 1 to 7). Their main contribution was that the system provided an indication to Twitter users about trustworthiness of tweets in real-time.

2.3 Panic Triggers

Twitter data includes a mixture of real news, informal discussions and rumors. Such discussions may contain information that can trigger panic during disasters. Some people may consider this information from social media more trustworthy than information carried in

traditional media. Many risk-factors associated with post-traumatic stress disorder, major depressive disorder, panic disorder, and generalized anxiety disorder occur amongst the public during natural disasters. During hurricane Harvey, some people felt that official warnings were exaggerating a danger and so causing needless worry or panic, and began to take them seriously (King, 2018). During the hurricane, evacuees were primarily adults who had presented with varied diagnoses, most commonly mood, anxiety and psychotic disorders. There was a significant need for medications and psychosocial support to address preexisting conditions, as well as emerging problems such as insomnia (Storch et al., 2019). Also, Amstadter et al. (2009) highlighted that many risk-factors associated with post-traumatic stress disorder (PTSD), major depressive disorder (MDD), panic disorder (PD), and generalized anxiety disorder occur amongst the public during natural disasters.

There has not been much work done in addressing panic problems during natural disasters using Twitter data. According to Stroud et al. (2013), indicators and triggers represent the information and actions that guide incident recognition, response, and recovery. Indicators are usually the measures or predictors of changes in demand and/or resource availability; triggers are decision points. It can be challenging to identify useful indicators and triggers from large and varied sources of available data; therefore, there is the need to understand how indicators can be used to support operational decision making, and to avoid panic. The following are a few examples of indicators and triggers during a crisis:

- Indicators:
    - Impact on community, including transportation and communications infrastructure.
- Triggers:

- Loss of paging and/or cellular service in an area.
- Loss of phone service in a hospital.
- Loss of electrical service in a hospital.
- Loss of water service.
- Closure of transit system.

According to Stroud et al. (2013), indicators may be categorized into predictive versus actionable, and certain versus uncertain. Predictive indicators cannot be directly impacted by actions taken by the agency/facility (e.g. "a hospital receiving notification that a pandemic virus has been detected"); however, actionable indicators are under the control of the facility, for example, "a hospital detecting high patient census". An indicator that is certain requires less analysis before action, while an indicator that is uncertain requires interpretation before action. Understanding the characteristics of the indicators helps inform decision makers about how best to use them. The authors proposed that four steps should be taken in response to an incident:

(1) Identify key response strategies and actions.

The strategies that the facility or agency would use to respond to an incident are identified. Such strategies could include disaster declaration, or establishment of an emergency operations.

(2) Identify and examine potential indicators.

Indicators that inform the decision to initiate actions are identified, for example, "a 911 call", or "witnessing a tornado".

(3) Determine trigger points.

Scripted triggers that may be derived from certain indicators are determined, for example, a mass casualty incident involves >20 victims.

(4) Determine tactics.

Tactics that could be implemented at these trigger points. Scripted triggers may appropriately lead to scripted tactics and a rapid, predefined response. After a facility determines what actions or strategies should be taken during an incident, it should examine indicator data sources that inform the initiation of these actions.

CHAPTER 3

Methodology

This research, a Panic Trigger Identification Framework (PTIF) was developed to extract and to collect tweets during hurricane disaster, to classify the tweets based on their relevance to the disaster, and to evaluate the credibility of the disaster-related tweets, and to identify tweets that contain panic triggers. Figure 2 shows an overview of PTIF. This framework contributes to the enhancement of emergency response and management by detecting the triggers on disaster related tweets that may lead to panic situations.

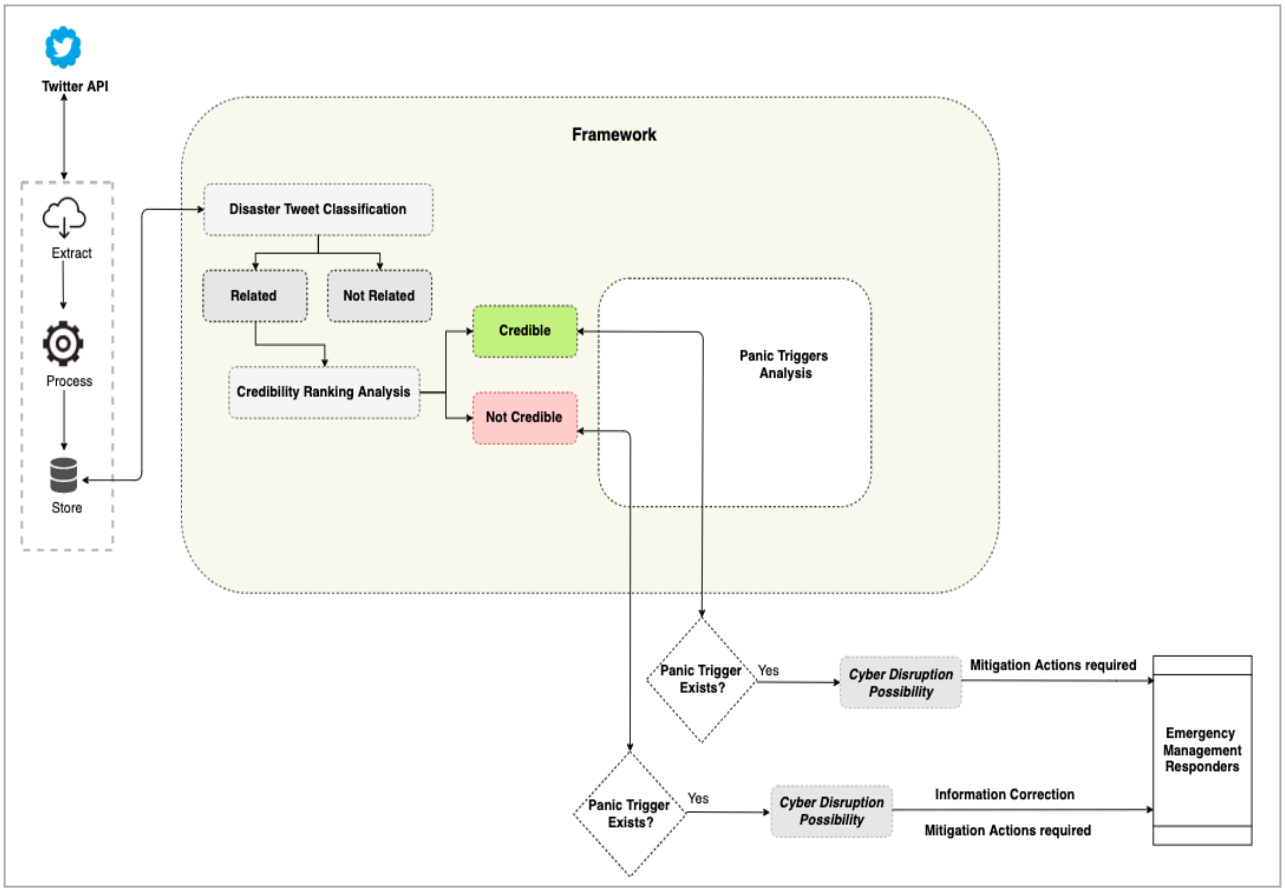

Figure 2. An overview of Panic Trigger Identification Framework (PTIF).

3.1 Tools and libraries

In this project, Python Programming Language is used to develop the PTIF. The following main libraries were used:

- **NTLK** (Natural Language Processing Toolkit). It is a platform for processing natural language data.
- **Regular Expressions** (regex or regexp for short). It is a package for processing text string patterns.
- **Pandas.** It is a data analysis tool for the Python programming language.
- **NumPy.** It is the fundamental package for scientific computing with Python.
- **TF-IDF/Count Vectorizer.** It gives the frequency of the word in textual contents or documents.
- **Scikit-learn.** It provides a range of supervised and unsupervised learning algorithms.
- **Matplotlib.** It is a collection of command style functions that create figures, a plotting area in a figure, and lines in a plotting area.

3.2 Data Collection

In order to conduct the analysis on Twitter data, a premium full archive Twitter API was used to collect historical tweets before, during, and after the time when both hurricane Florence and hurricane Michael happened. Once the premium account was set up, a set of tokens and keys was provided by Twitter API, see Figure 3. These keys and tokens were integrated into the tweet extraction tool. More than 26,000 tweets were collected for both events (10,000 tweets for hurricane Michael and 16,000 tweets for hurricane Florence). The search terms used include “hurricane Michael”,” hurricane Florence”,” hurricane evacuation”,” hurricane warning”, etc. For the purpose of evaluating the credibility of the tweets, specific attributes and entities were extracted

for each tweet object. Table 1 shows the list of the attributes extracted and stored in the hurricane datasets.

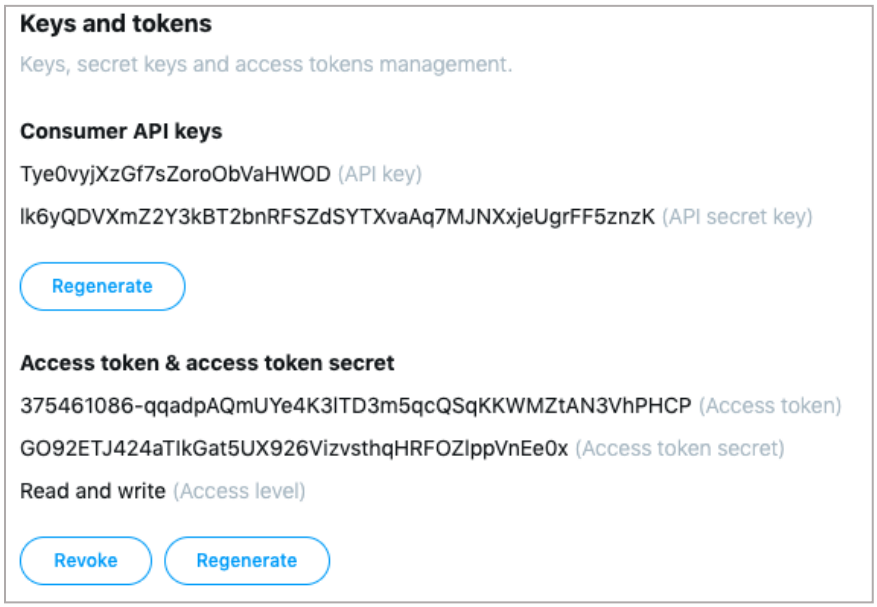

Figure 3. Twitter API tokens used for extracting historical tweets.

Table 1

The attributes extracted and stored in the hurricane datasets

User Attributes	**Tweet Attributes**
username	tweet
user_profile_description	URL_in_Tweet
user_screen_name	tweet_created_date
number_of_follwers	tweet_source
number_of_friends	number_of_retweets
user_account_created_date	number_of_likes
user_likes_count	length_of_tweet
user_posts_count	hashtags_contained_in_tweet

user_account_verified?	

3.3 Data Preprocessing

After the data collection, Natural Language Processing such as Regular Expressions to was applied to clean each tweet by removing contents such as hashtags, URLs, user mentions, special characters, numbers, etc. The aim of the data cleaning process is to remove any unwanted content from the training data, since such data can create noise when implementing the classification. The data preprocessing step is crucial since any noise can affect the performance of machine learning algorithms. Once all tweets were pre-processed and, they were stored in the dataset and used for data annotation, see Figure 4. Data preprocessing includes the following:

- Remove Escaping HTML Characters. Data contains many of entities such as < > &amp.
- Punctuations and numbers. Such as {},?, !, = , &, $, 1, 6, etc. Removing hashtag and user mention symbols like ("#", "@"), and keep the actual words following these symbols.
- Split Attached Word. Some of the tweets have an attached word such as "HurrincaneMichael","BeAwareOfTheStorm" etc. These words must be split to separate words.
- Standardize Words. The users sometimes use words in improper formats. For example, the sentance"it's soooo windy " should be "it's so windy". Such words need to be changed to correct forms.
- Remove Stop-words: Stop-words are filtered out before of natural language data is processed. Stop-words are generally the most common words in a language. Figure 5 shows a list of examples of stopwords. There is no single universal list of stop words used by all-

Natural Language Tools (NLT); there is a built-in stopwords library within NTL in which it contains a list of standard words and the list can be modified and expanded.

- Remove of URLs.
- Removal of Emojis. Emojis are pictographs of faces, objects, and symbols, but the machine learning treats them as a set of Unicode characters, for example," 😊 " is translated as "U+1F60A".

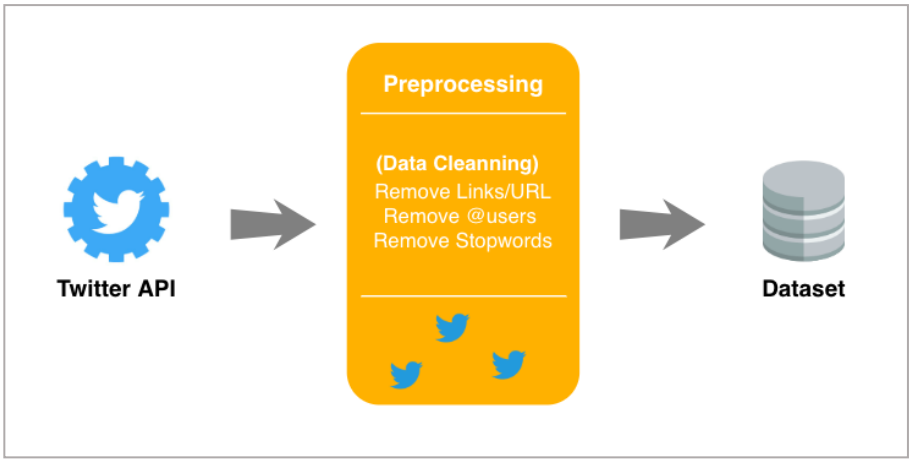

Figure 4. The process for data preprocessing and storing in datasets.

{'ourselves', 'hers', 'between', 'yourself', 'but', 'again', 'there', 'about', 'once', 'during', 'out', 'very', 'having', 'with', 'they', 'own', 'an', 'be', 'some', 'for', 'do', 'its', 'yours', 'such', 'into', 'of', 'most', 'itself', 'other', 'off', 'is', 's', 'am', 'or', 'who', 'as', 'from', 'him', 'each', 'the', 'themselves', 'until', 'below', 'are', 'we', 'these', 'your', 'his', 'through', 'don', 'nor', 'me', 'were', 'her', 'more', 'himself', 'this', 'down', 'should', 'our', 'their', 'while', 'above', 'both', 'up', 'to', 'ours', 'had', 'she', 'all', 'no', 'when', 'at', 'any', 'before', 'them', 'same', 'and', 'been', 'have', 'in', 'will', 'on', 'does', 'yourselves', 'then', 'that', 'because', 'what', 'over', 'why', 'so', 'can', 'did', 'not', 'now', 'under', 'he', 'you', 'herself', 'has', 'just', 'where', 'too', 'only', 'myself', 'which', 'those', 'i', 'after', 'few',

Figure 5. Examples of stop words removed from each tweet.

3.4 Identifying Disaster Related Tweet

3.4.1 Annotating disaster-related tweets

Researchers have used manual annotation to label tweets for machine learning algorithms. Manual annotation can be costly and time consuming. In this research, an automated tweet annotation framework was implemented, as shown in Figure 6. This framework uses a disaster-related term dictionary to automatically label the tweets in both hurricane Michael and Florence datasets. The disaster-related term dictionary contains predefined keywords such as #HurricaneMichael, #Hurricane, #hurricanewarining, #storm, #stormsurge, etc. If a tweet contains at least two keywords from the disaster-related term dictionary, then the tweet is automatically labeled as “Related”. Otherwise, it is labeled as “Not_Related”. Such a framework enables fast labelling for disaster-related tweets.

In order to establish the ground truth for the labeling system, create more accurate labeling and evaluate the accuracy, three participants have manually checked and labeled the first 2,000 tweets in each dataset. Thus, each hurricane related keywords the participants come across were appended to the disaster-related term dictionary while manually labeling the tweets. Then the framework was run on the rest of the tweets. The aspects considered for deciding whether a tweet is related to the disaster or not are as follows:

- Does the tweet contain information about the Hurricane?
- What is the topic of the tweet?
- What are the hashtags contained in the tweet?
- What is the list of keywords the tweet contains?

The quality of the labeling highly depends on the disaster-related terms that the dictionary contains; the more terms included in the dictionary; the tweets that would be identified as disaster-

related tweets. The dictionary used in this research was manually created by a) analyzing 2,000 tweets and extracting the disaster-related keywords from them, b) adopting the keywords used in other work that used dictionary-based approach (To et al., 2017), and c) extracting keywords from hurricane disaster news articles and weather channels. The dictionary was created to cover hurricane disaster related terms.

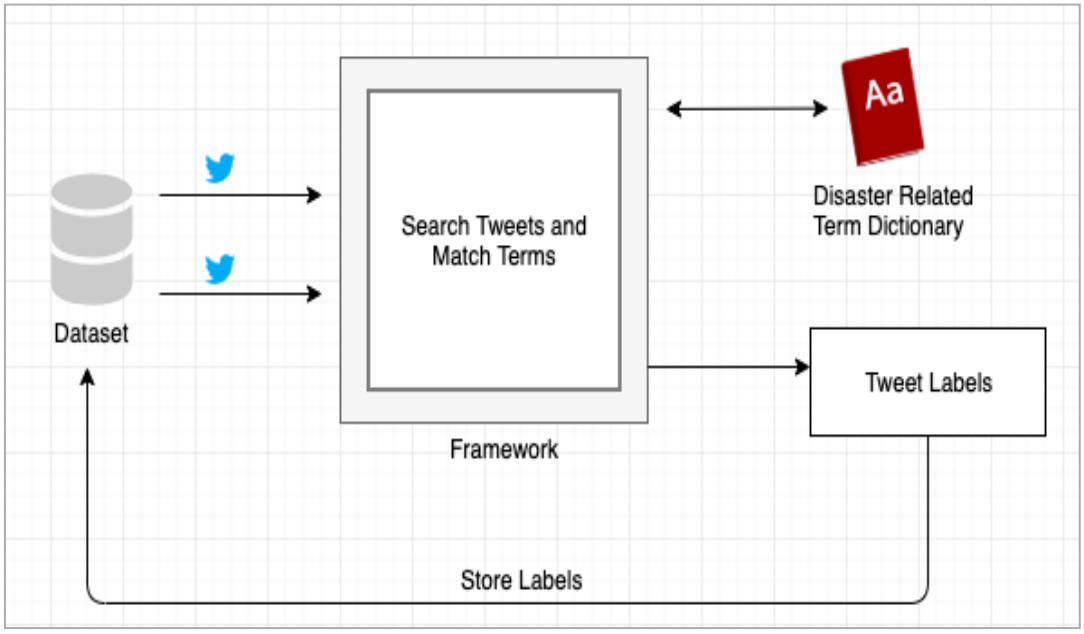

Figure 6. The automated tweet annotation framework.

3.4.2 Word vectorization

Text Analysis is a major application field for machine learning algorithms. However, raw data, which is a sequence of symbols, cannot be fed directly to the algorithms themselves as most of these classifiers expect numerical feature vectors with a fixed size rather than the raw text documents with variable length. On other words, machine learning algorithms operate on a numeric feature space, expecting input (threshold) as a two-dimensional array where rows are instances and columns are the features. In order to perform machine learning on text, there is the need to transform the tweets into vector representations such that numeric machine learning can

be applied. This process is called feature extraction or vectorization. This numerically representation gives the ability to perform meaningful analytics and also creates the instances on which machine learning algorithms operate. In text analysis, instances are entire documents or utterances, which can vary in length from quotes or tweets to entire books, but whose vectors are always of a uniform length. Each property of the vector representation is a feature. Therefore, the texts to need to be converted to vectors without losing much of the information for better classification accuracies. Thus, stop-words such as “and”, “or”, “hey”, “could”, “can” and “would” etc. are removed from the tweet texts. These words are used to construct sentences; however, such words would create noise that could affect the classification accuracies.

In this research, CountVectorizer and TfidfVectorizer were used to extract features for classification. The two methods are described below.

3.4.2.1 CountVectorizer

It is a method for transforming a document into vectors by counting occurrences of each word in each document, Table 2 shows an example of how CountVectorizer converts words into counts and stored in matrix. Each element in the matrix refers to a word and its count of occurrences in a document. CountVectorizer is a tool provided in Scikit Learn (Pedregosa et al., 2013).

Table 2

An example of the word counts for the frequencies of the terms

Sentences
Sentence 1 = "Apple is a beautiful fruit, monkey eats an apple" Sentence 2 = "monkey eats apples"

Sentence 3 = "monkey eats an apple and a banana"
CountVectorizer conversion
['an', 'and', 'apple', 'apples', 'banana', 'beautiful', 'eats', 'fruit', 'is', 'monkey'] [[1 0 2 0 0 1 0 1 1 0] [0 0 0 1 0 0 1 0 0 1] [1 1 1 0 1 0 1 0 0 1]]

3.4.2.2 TfidfVectorizer

TfidfVectorizer is a tool provided by Scikit Learn (Pedregosa et al., 2013). It is short for Term Frequency-Inverse Document Frequency, which converts a collection of raw text into a matrix of TF-IDF features and count the frequencies of tokens in the raw text. In the context of tweets, features and samples are defined as follows:

- Each individual token occurrence frequency is treated as a feature.
- The vector of all the token frequencies for a given tweet is considered a multivariate sample.

TFIDF converts textual data to a numeric form. The vector value is the product of these two terms: TF (Time Frequency) and IDF (Inverse Document Frequency) (Ghosh et al., 2018). Table 3 shows an example of how TfidfVecctorizer converts word into weights and frequencies and stored into a matrix. Each weight on the matrix represents a word. Relative term frequency is calculated for each term within the document as follows:

(1) *TF (t, d) = (number of times(t)appears in tweet(d))/(total number of terms in tweet(d))*

Next, we get the Inverse Document Frequency (IDF), which measures how important a word is.

(2) *IDF (t, D) = log((total number of tweets(D))/(number of tweets with the terms(t) in it))*

Once the values for TF and IDF are calculated, TFIDF can be calculated as follows:

(3) *TFIDF (t, d, D) = TF (t, d) * IDF (t, d, D)*

Once TF-IDF Vectorizer is instantiated, it will calculate the scores for terms for each tweet in the dataset and convert textual data into numeric form. The TFIDF-transformed data is fit into classification algorithms.

Table 3

An example of the weights for the frequencies of the terms in sentences

Sentences
Sentence 1 = "Apple is a beautiful fruit" Sentence 2 = "monkey eats apples" Sentence 3 = "monkey eats an apple and a banana"
TfidfVectorizer conversion
['an', 'and', 'apple', 'apples', 'banana', 'beautiful', 'eats', 'fruit', 'is', 'monkey'] [0. 0. 0.40204024 0. 0. 0.52863461 0. 0.52863461 0.52863461 0.] [0. 0. 0. 0.68091856 0. 0. 0.51785612 0. 0. 0.51785612] [0.45954 0.45954 0.34949 0. 0.45954803 0. 0.34949812 0. 0. 0.34949812]

3.4.3 Learning-based systems for identifying disaster-related tweets

This phase focuses on comparing the performance of different learning algorithms in identifying disaster related tweets. First matching based method was used to label tweets collected on Hurricane Florence and Hurricane Michael. The performance of using different learning methods using different term vectorizers – TfidfVectorizer and CountVectorizer to identify disaster-related tweets were compared.

The framework for comparing learning-based methods is shown in Figure 7. First, disaster-related data is collected from Twitter and stored. Next, the data is processed and cleaned from any contents that would create noise when transforming the textual data to features or when fitting the features into the classification models. Then the processed data is passed into a labeling module in which each tweet labeled as “Related” or “Not-Related” to the disaster. The next stage is to extract features that can be used in classification. The final stage is to apply machine learning classification models on testing data and measure the performance of each model.

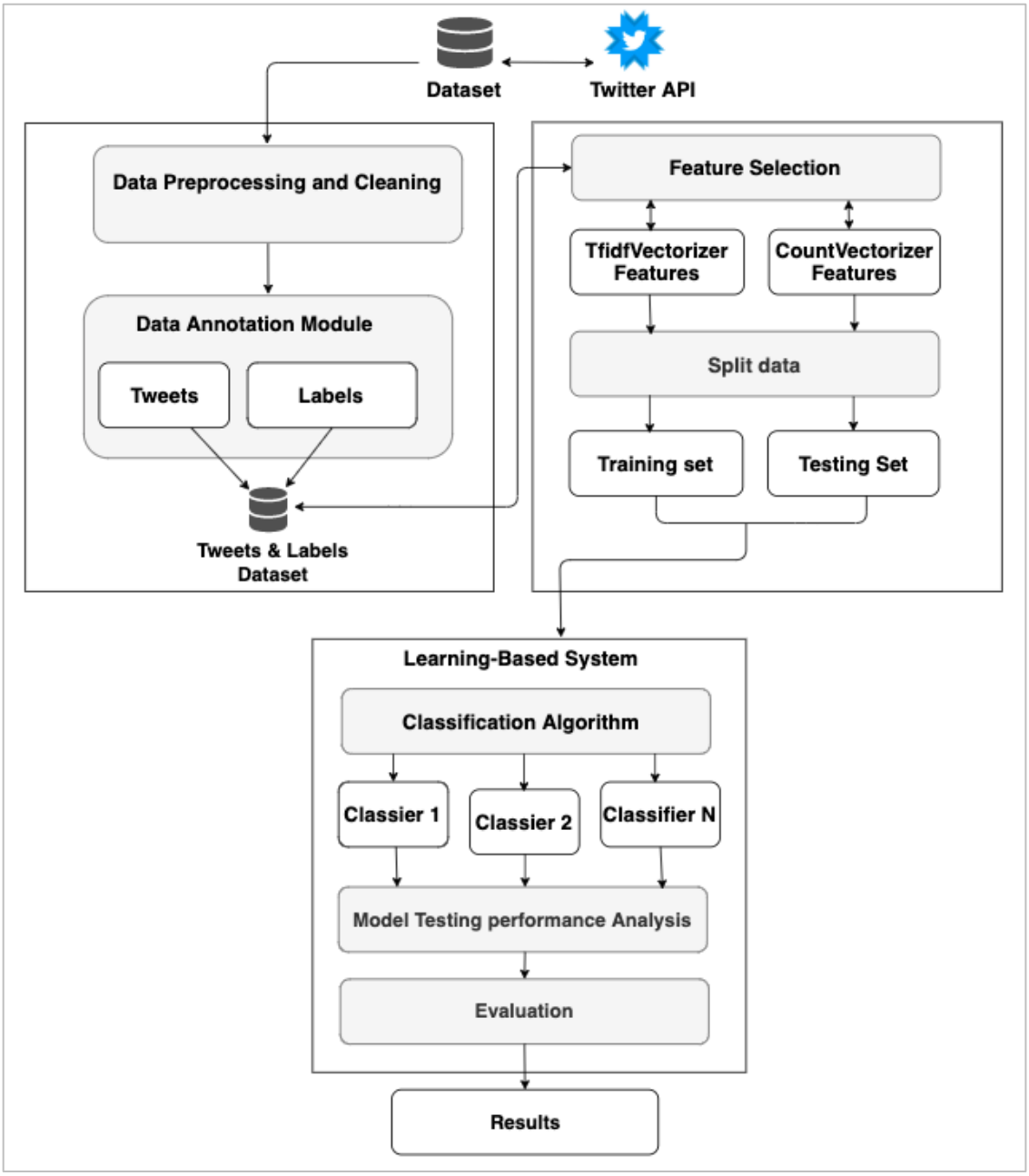

Figure 7. The framework for classifying and comparing learning-based models.

For this phase, the framework used different supervised machine learning models suitable for classification with discrete features (e.g., word counts for text classification) and binary classification to identify disaster-related tweets such as Naive Bayes, Support Vector Machine

(SVM), k-Nearest Neighbors (KNN), Logistic Regression, Random Forest, and Decision Tree. These models are briefly described below.

3.4.3.1 Naive Bayes

It is a machine learning algorithm that uses probabilistic classification based on Bayes theorem with an assumption of independence among features. This algorithm performs classification tasks well and learns quickly in numerous real-world supervised classification problems (Chandel et al., 2016). The algorithm requires a smaller number of training data compared with other algorithms (Deekshatulu et al., 2013). This classification method is used to calculate the probability of a specific tweet belonging to each class. The class which has maximum probability is expected to be the class of that tweet. The Naive Bayes theorem is as follows:

(4) $P(Y/X) = \frac{P\left(\frac{X}{Y}\right).P(Y)}{P(X)}$

3.4.3.2 Support Vector Machine (SVM)

It is a discriminative supervised machine learning algorithm that is formally defined by a separating hyperplane. It has the capability to predict and analyze regression and to classify a dataset (Polat et al., 2017; Ashktorab et al., 2014). In other words, given labeled training tweet data, the algorithm outputs an optimal hyperplane which categorizes new examples in two-dimensional space (disaster-related, or not related). This hyperplane is a line dividing a plane in two parts where each class lays in either side. SVM is basically a linear classification approach based on two classes (Boukenze et al., 2016). In this experiment, the learning of the hyperplane is done in linear SVM which is done by transforming the problem using a linear kernel.

For the kernel the equation for predicting the class of a new input using the dot product between the tweet input (x) and each support vector (xi) is calculated as follows:

(5) $f(x) = B(0) + sum(ai * (x , xi))$

This is equation involves calculating the inner classes of a new input vector (*x*) with all support vectors in training data. The coefficients *B(0)* and *ai* (for each input) must be estimated from the training data by the learning model (Patel, 2017). For example, the inner product of the vectors *[2, 3]* and *[5, 6]* is *2*5 + 3*6* or *28*.

3.4.3.3 K-Nearest Neighbor (KNN)

It is a supervised learning algorithm. It is a popular, simple, highly efficient and effective algorithm for pattern recognition (Chandel et al., 2016). It is good for a large number of records and fast to train the models. First, the model computes a distance value between the tweet input to be classified and every tweet in the training dataset. Then the model picks *k* data points from the training data that are nearest to the test data (the items with the *k* lowest distances). Then it conducts a "majority vote" among those data points to predict the class of the test data. The dominating classification in that pool is decided as the final classification (Soni et al., 2018).

In KNN, the nearest class is recognized by using different distance measurements such as Manhattan distance, Minkowski distance, Euclidean distance, and Hamming distance. The distance formulas are given as follows:

(6) *Euclidean-Distance (X , Y)* $= \sqrt{\sum_{k=1}^{n}(x_k - y_k)^2}$

(7) *Manhattan-Distance (X , Y)* $= \sum_{k=1}^{n} |x_k - y_k|$

(8) *Minkowski-Distance (X , Y)* $= \sum_{k} (|x_k - y_k|)^q|)^{\frac{1}{q}}$

(9) *Hamming-Distance (X , Y)* $= \sum_{k=1}^{n} d(x_k - y_k)$

$$d(x_k - y_k) = \begin{cases} 1 & si \quad x_k \neq y_k \\ 0 & si \quad x_k = y_k \end{cases}$$

Here, X and Y are the objects to be compared, in which $X = \{x_1, x_2, \dots, x_k\}$ and $Y = \{y_1, y_2, \dots, y_k\}$, with dimension n (attribute number), and x_k and y_k denote the k^{th} attributes of X and Y respectively (Bonet et al., 2008).

3.4.3.4 Logistic Regression

It represents a probabilistic machine learning model that is used to find the probability of a certain class or event existing such as pass/fail, win/lose, alive/dead or healthy/sick. In our case, the event is 'Related' or 'Not Related' tweets. Logistic regression is appropriate when the dependent variable is binary (0/ 1, True/ False, Yes/ No) in nature (Raghuwanshi et al., 2017). In this research, 1 represents the related tweets and 0 represents the non-related ones. Logistic regression is used to describe data and explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables. Here the value of Y ranges from 0 to 1 and it can be represented by the following equation:

(10) $$odds = \frac{Probability\ of\ event\ occurring}{Probabilty\ of\ event\ not\ occuring} = \left(\frac{P}{1-P}\right)$$

p is the probability of presence of the characteristic of interest. For example, when p is larger than 0.5, then the odds will be the input belongs to one class.

3.4.3.5 Decision Tree

It is a classification technique that can handle both numerical and categorical data. It is a graph that follows a branching method to exhibit every possible outcome for a decision (Nair et al., 2017). A tree is constructed in a top-down recursive divide and conquer manner. All the training samples are placed as the root element then partitioned recursively based on the selected attribute. Decision tree consists of two nodes namely leaf nodes and decision nodes. A decision node or internal node contains two or more sub-branches, in which each branch denotes the test on a particular attribute. The leaf node contains a resulting classification decision or class label.

The Classification techniques of Decision tree are simple and fast, and the tree format is simple and easy to understand.

3.4.3.6 Random Forest

It is an ensemble learning method that consists of a construction of multiple decision trees called Forests. Individual decision trees are generated during the training time using randomly selected attributes in each node, in which the split is determined. Using Random Forest for classifying a tweet, the tweet is passed to all the trees in the forests. Each tree will give a classification result. That is, all the decision trees will give their vote on the classification individually. Then the algorithm chooses the most popular class voted. This Algorithm takes less time to predict the output in comparison with other classification models (Nair et al., 2017).

3.4.4 Model Evaluation Metrics

In this research, the following metrics were used to compare the performance of different learning-based methods in classifying disaster data: Confusion Metrics, Binary classification tests, and Receiver operating characteristic (ROC).

3.4.4.1 Confusion Matrix

A confusion matrix is an *n x n matrix*, where *n* is the number of classes to be predicted. For binary classification problems, the number of classes is two; thus, the confusion matrix has two rows and columns. The rows of the confusion matrix represent the target classes while the columns represent the output classes. The diagonal cells in each table show the number of cases that were correctly classified, and the off-diagonal cells show the incorrectly classified cases as showing in Table 4.

Once all the instances are classified, the predicted results are compared to the actual values. Important parameters that can be derived from the confusion matrix, which are helpful to

understand the information that the matrix provides. Some of the most important ones are the classification Accuracy (ACC), the Error Rate (ER), the sensitivity or True Positive Rate (TPR), and the specificity or True Negative rate (TNR), (Salman et al., 2018, Damousis et al., 2012; Elamvazuthi et al., 2018). All of them are calculated by using the values TP, TN, FP and FN written in the confusion matrix. These values are defined below:

- **True Positive (TP):** The predicted label is "Related", and the actual is "Related".
- **True Negative (TN):** The predicted label is "Not Related", and the actual is "Not Related".
- **False Positive (FP):** The predicted label is "Related", but the actual is "Not Related".
- **False Negative (FN):** The predicted is label "Not Related", but the actual is "Related".

The classification accuracy is the ratio of instances correctly classified, and it can be calculated using the following formula:

(11) $$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

The error rate, which is the ratio of instances misclassified, is given by:

(12) $$Error\ Rate = \frac{FP+FN}{TP+TN+FP+FN}$$

The sensitivity, which is the portion of actual positives which are predicted as positives, we use the following expression:

(13) $$Sensitivity\ (TPR) = \frac{TP}{TP+FP}$$

Finally, the specificity, which is the portion of actual negatives predicted as negative, is calculated as follows:

(14) $$Specificity\ (TNR) = \frac{TN}{TN+FN}$$

Table 4

Confusion Matrix and performance metrics

<table>
<tr><td rowspan="2">Actual</td><td colspan="3">Predicted</td></tr>
<tr><td>Positive</td><td>Negative</td><td rowspan="3">Accuracy =
$\frac{TP + TN}{TP + TN + FP + FN}$</td></tr>
<tr><td>Positive</td><td>TP</td><td>FN</td></tr>
<tr><td>Negative</td><td>FP</td><td>TN</td></tr>
<tr><td></td><td>Precision =
$\frac{TP}{TP + FP}$</td><td>Recall =
$\frac{TP}{TP + FN}$</td><td>F-score =
$2 * \frac{Recall * Precision}{Recall + Precision}$</td></tr>
</table>

3.4.4.2 Receiver operating characteristic (ROC)

ROC curve is a testing method for binary classification problems. It provides a comprehensive and visually attractive way to summarize the accuracy of predictions and compare the performance of classification models (Hajian-Tilaki, 2013).

By varying the value of the decision threshold τ between 0 and 1, we obtain a set of different classifiers for which we can calculate their specificity and their sensitivity. The points of a ROC curve represent the values of those parameters for each of the values of the decision threshold.

3.4.4.3 Acceptance and rejection rates

The False Acceptance Rate (FAR) is the measure of the likelihood that a classification model accepts incorrect instances while predicting when it should be rejected. FAR is the ratio of the number of false acceptances divided by the number of identification attempts.

(15) $$FAR = \frac{Number\ of\ incorrect\ instances\ accepted}{Total\ of\ instances}$$

The False Rejection Rate (FRR) is the measure of the likelihood that a classification model incorrectly rejects correct instances while predicting when it should be accepted. FRR is the ratio of the number of false rejections divided by the number of identification attempts.

(16) $$FRR = \frac{Number\ of\ correct\ instances\ rejected}{Total\ of\ instances}$$

Equal error rate (EER) is used to predetermine the threshold values for its FAR and FRR. When the rates are equal, the common value is referred to as the equal error rate. The value indicates that the proportion of FAR is equal to the proportion of FRR. The lower the EER value, the higher the accuracy of the classification model (Salve, 2018; Salem, 2019).

(17) $$EER = \frac{FAR+FRR}{2}$$

For each classification phase in this research the Acceptance Rate, and the Rejection Rates represents the classes corresponding to the features the classifiers use. For example, in this research the instances are: for disaster-related tweet identification the instances represent the classes: “related or “not related”, for the credibility analysis the instances represent the classes: “credible” and “not credible”, and for panic-triggering tweet the instances represent the classes “Mitigation”, “Mitigation_and_Correction”, and “No_Triggers Contained”. Hence, the FAR, FRR and EER were measured for each phase. In general, for classification systems, the main performance indicator is the ROC curve, which is a plot of True Acceptance Rate which is TAR=1-False Rejection Rate against False Acceptance Rate (FAR), which is computed as the number of false instances classified as positive among all instances. The closer the curve is to the top left corner, the better.

3.5 Tweet Credibility Analysis

3.5.1 Tweet credibility annotation

In order to annotate the tweets and assign credibility labels for each tweet, a framework was implemented in which user-based features and content-based features were considered to evaluate the credibility and trustworthiness of each disaster-related tweet and to assign a credibility label. The framework uses a 10-point-scale credibility rating system in which tweets with score between 5-10 points are considered “Credible”, and tweets with scores lower than 5 points are considered “Not Credible”. The rating of each tweet depends on points given to the tweet based on the combination of user-based features and content-based features as explained below.

In this step, tweet attributes were analyzed to extract new features for calculating the credibility of each tweet. These new features were stored in the datasets as new attributes corresponding to the tweets analyzed. These new attributes were used as features to classify the tweets into “Credible” and “Not Credible”. There are two types of features which are explained below.

3.5.1.1 User-Based Features

- ***Verified User Account***

In user-based analysis, first we check if the user’s account is verified by Twitter. If so, it is considered as “Credible” and is given the max points, since the verified badge for any Twitter accounts confirms that all tweets coming from that profile are credible (Pinegar, 2018). Figure 8 shows an example of a Twitter verified account.

Figure 8. An example of verified Twitter account.

- ***Trusted username***

The usernames and account description are checked to see if they contain trusted information sources using a dictionary of trusted sources. For example, if the username or description contains trusted names such as “”ABC News “Weather News Channel”, “news channel”, or “breaking news”, the user gets credibility points.

- ***Slang in user profile description***

If the username and description have slang or swear words, the framework deducts credibility points are deducted. Figure 9 show the flow of the dictionary-based system to check the username and description for trusted sources.

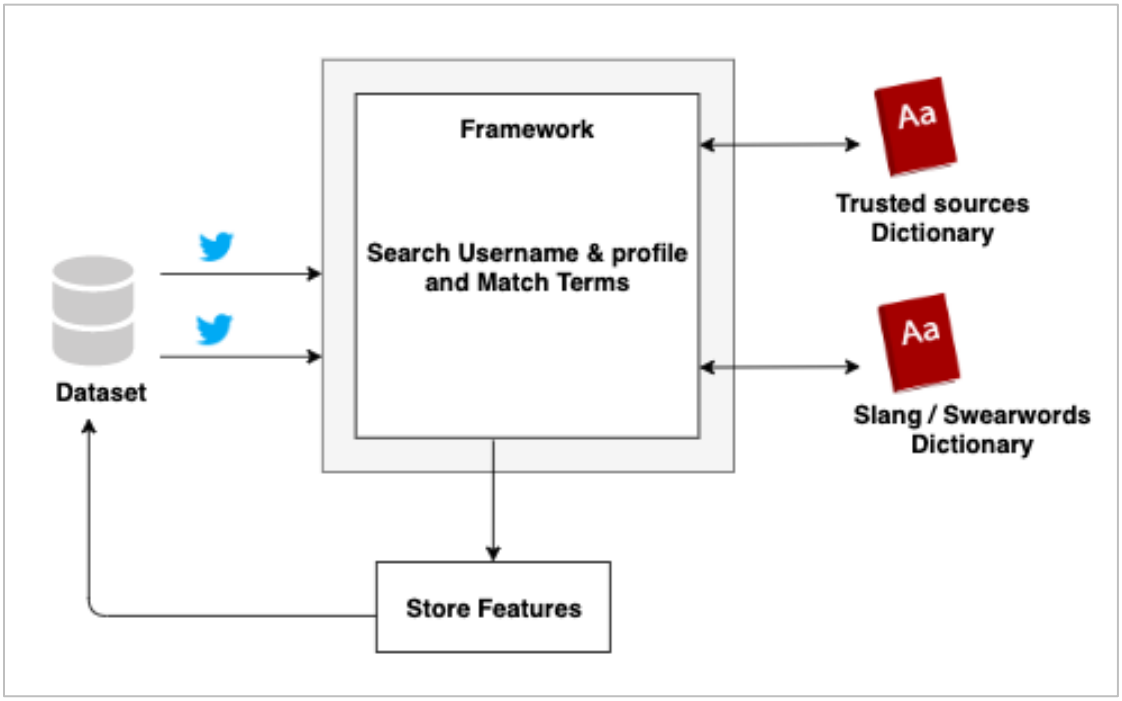

Figure 9. An overview of the dictionary-based analysis to identify user-based features for credibility.

- ***User Follower/Following Ratio***

The popularity and influence of a user measured by the follower/following ratios, assigning high credibility scores to high ratio is used as a feature for credibility (Zhang, 2015). Table 5 shows the interpretation of the ratio calculated. Follower/Following Ratio is calculated using the following formula (Parsons, 2017):

(18) $$Follower/Following\ Ratio = \frac{Number\ of\ Followers}{Total\ Following}$$

Table 5

The Follower/Following ratio indicators

Follower/Following Ratio	Influence Description	Category

<0.5	Users that are inexperienced with Twitter and are spamming followers in hope for being followed.	Spammer
0.5-1	Users that are likely to be using Twitter automation tools but are following the wrong people or has poor quality content leading to poor number of followers.	Suspicious
1-2	Users that have some success with Twitter automation tools but need to focus on other strategies to drive more followers.	Normal
2-10	Users that are either master of Twitter automation tools or has incredible content to grow their account.	Micro Influencer
10+	Users are likely to be micro-celebrities or rising stars that are popular on other social media channels.	Influencer

Figure 10 show how credibility score is assigned to a tweet based on a user-features. Before starting the process, the overall credibility score is initiated to zero. Then the scoring is pursued as follows:

- When the user is verified by Twitter the user gets a "+10" points directly as Twitter confirms the user's credibility. However, if the user is not verified the framework check the username; when the algorithm checks whether the username is from a trusted source such as "weather reporter", "news journalist" etc., it adds a "+1" point to the overall credibility score.
- However, if the username is not from a trusted source, no points is deducted as the user could fall in the category of Micro Influencer or Influencer. Then the algorithm checks the

user profile description for slang and swear words; once such words detected, “-1” point is added to the score. Otherwise, “+1” is added.

- The algorithm checks the Follower/Following ratio for the users; Micro Influencer and Influencer users get “+2” points added to the overall score, the Normal users get “+1” score, and the users with lower ratios get “-1” point since they fall into the Spammer or Suspicious user categories.

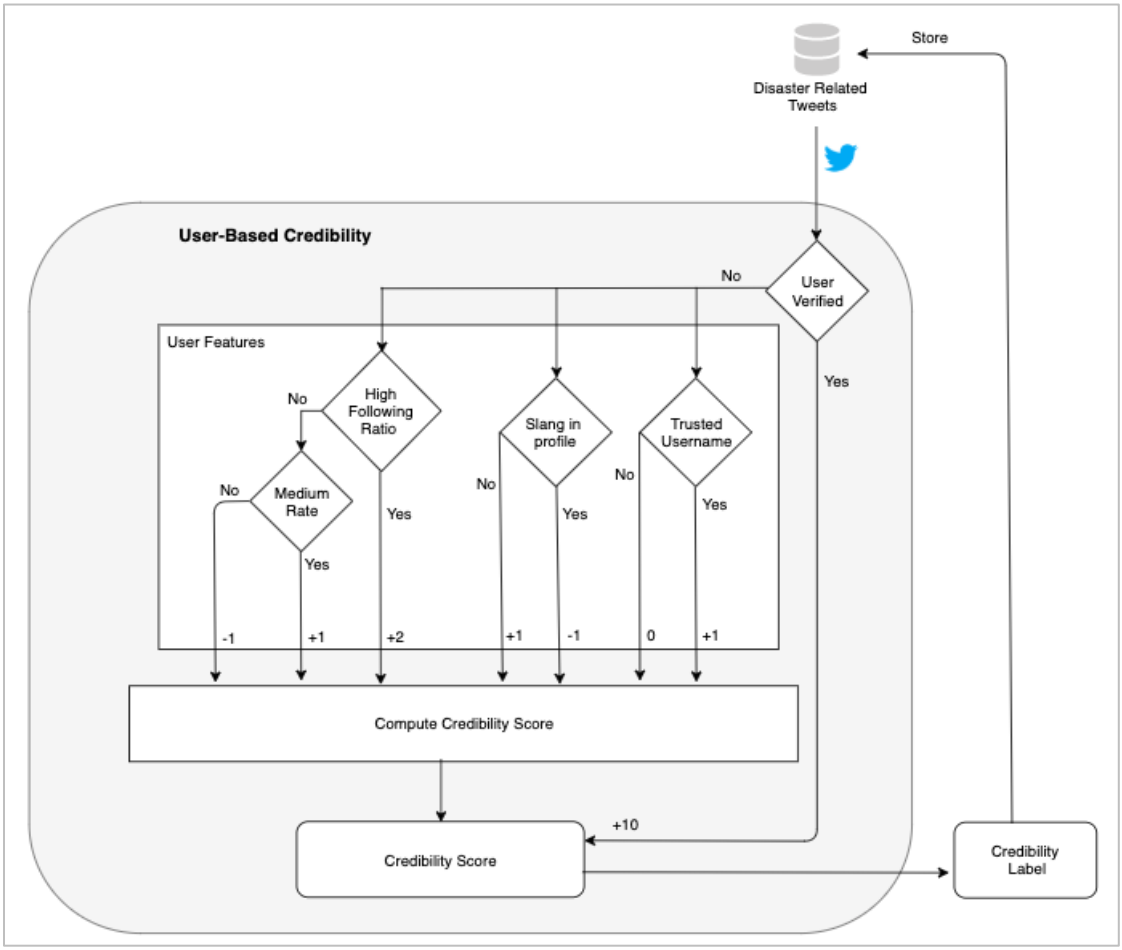

Figure 10. Calculating credibility score from user- based features.

3.5.1.2 Content-Based Features

- ***Slang in tweet***

If a tweet has slang or swear words, the credibility score is deducted. Tweets that have slang or swear words tends to be not credible (Afify et al., 2019).

- ***Tweet has Question or Exclamation marks in tweet***

If a tweet contains Question marks "?" or Exclamation marks"!", the credibility score is deducted. Tweets that contain Question or Exclamation marks tend to be not credible (Afify et al., 2019).

- ***Valid tweet length***

Credible tweets tend to be wordy and descriptive; therefore, the credibility score is increased for lengthy tweets (Jardaneh et al., 2019).

- ***Trusted and valid URL in tweet***

Tweets that contain bad URLs tend to be less credible (Shariff et al., 2014). The URLs in tweets are validated in two steps: (a) Validate the domain of the URL by searching for keywords such as "news", "weatherchannel", "hurricane-michael", "bbc-news" etc. in the URL. (b) Validate the URL request. The system checks whether the URL request is valid by issuing the request and checking the response code. Table 6 shows the interpretations of the request codes and the response status. Figure 11 shows an example of how the algorithm validates the URLs in tweets.

Table 6

URL request status codes

Status Code	Meaning	Status Code Example
2XX Success	It indicates the action requested by the client was received, understood and accepted.	200 OK
3xx Redirection	It indicates the client must take additional action to complete the request.	301 Moved Permanently

4xx Client Errors	It indicates the error seems to have been caused by the client.	400 Bad Request
5xx Server Errors	It indicates the server failed to fulfill a request.	502 Bad Gateway

Figure 11. An example of the output of the algorithm URL validation .

- ***Tweet Engagement Ratio***

The tweet engagement ratio measures how a tweet was received and has been interacted with by other users on Twitter (Meinert et al., 2019). Table 7 shows the descriptions of the engagement ratio for the tweets. It is calculated as follows (Mee et al., 2018):

(19) $$Engagement\ Ratio = \frac{Number\ of\ tweet\ likes + number\ of\ retweets}{Total\ number\ of\ user\ posts}$$

Table 7

Tweet Engagement ratio descriptions

Engagement Ratio	Description	Engagement Level

0% - 0.02%	The ratio is considered to be low. An influencer could expect between 0 - 0.2 reactions for every 1000 followers.	Low Engagement
0.02% - 0.09%	The ratio is considered to be good. An influencer could expect between 0.2 - 0.9 reactions for every 1000 followers.	Mild Engagement
0.09% - 0.33%	The ratio is considered to be high. An influencer could expect 0.9 - 3.3 reactions for every 1000 followers on Twitter.	High Engagement
0.33% - 1%	The ratio is considered to be very high. An influencer could expect 3.3 - 10 for every 1000 Twitter followers.	Very High Engagement

Figure 12 show how credibility score is assigned to a tweet based on a Content-features. At this point, the overall score is obtained based on the user-based features. It is the initial value before considering the content-based features. Then the scoring is pursued as follows:

- The URLs contained in the tweet are extracted and checked for their trustworthiness and validity; if the URLs are valid and trusted, the credibility score gets a “+2” points. Otherwise, “-1” point is added to the overall score.
- The system checks whether the tweet included slang or swear words exist. If such words exist, “-1” point is added to the score. If no slang or sear words exist, “+1” point is added to the credibility score.
- The system checks if the tweet has question and exclamation marks, if it has, “-1” point from is added to the score. Otherwise, “+1” point is added.

- The length of the tweet; tweets is then checked. Tweets with less than twenty characters "-1" point is added to the credibility score. Otherwise, "+1" point is added.
- The system checks the tweet engagement ratio aggregation. Highly engaged tweets get "+3" points added to the credibility score, the mildly engaged tweets get "+1" point, and the tweets with lower engagement ratios get "-1".

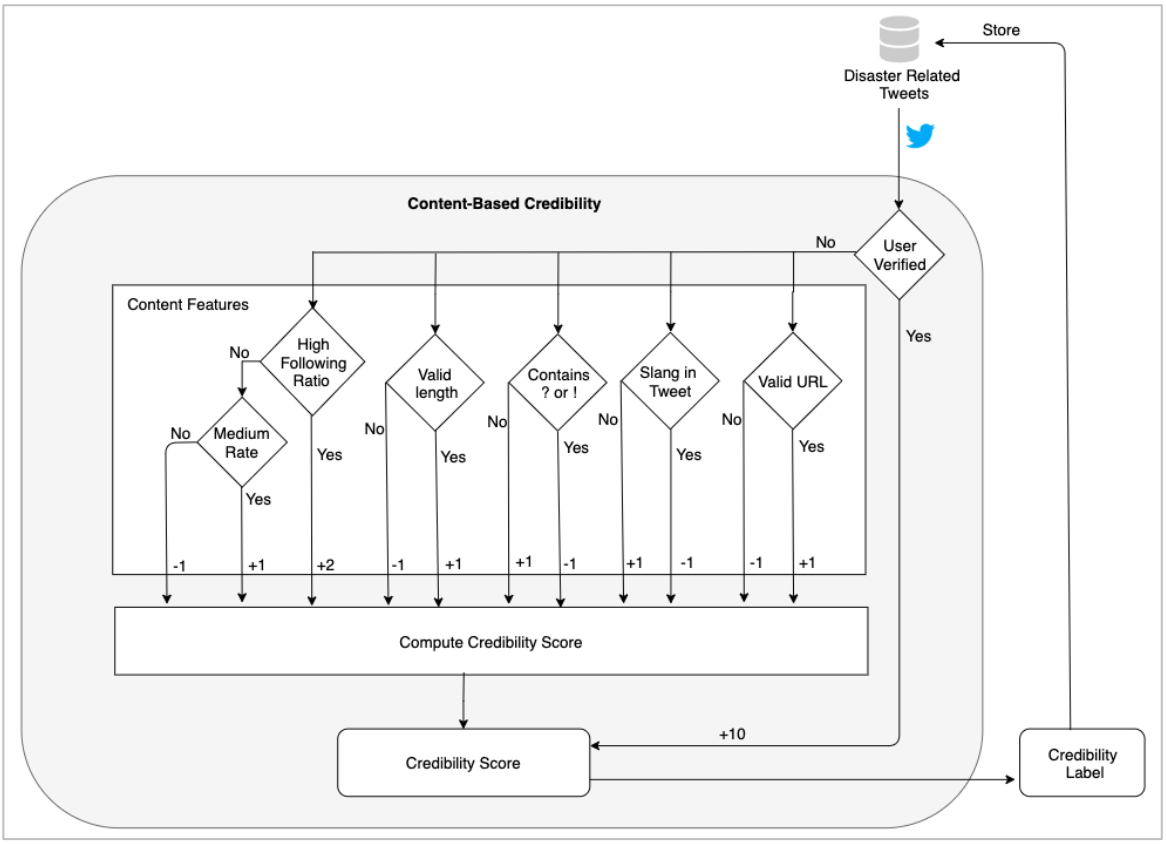

Figure 12. Calculating the credibility score based on Content-based features

3.5.2 Tweet credibility classification

After classifying tweets into disaster-related and not disaster related tweets. The credibility of disaster-related tweets was analyzed, as the tweets that are not related to the disaster usually do not carry helpful information about the disaster. Learning algorithms were used to classify the credibility of the tweets using the user-based features and content-based features. Table 8 shows the features used for the credibility classification. These features were created from analyzing the

extracted tweet entities discussed in section 3.2. Different classification models such as: Naive Bayes, Support Vector Machine (SVM), k-Nearest Neighbors (KNN), Logistic Regression, Random Forest, and Decision Tree, were used to classify the credibility of the tweets, and their performance was compared. Figure 13 shows the process of classifying the credibility of the tweets using different classification models.

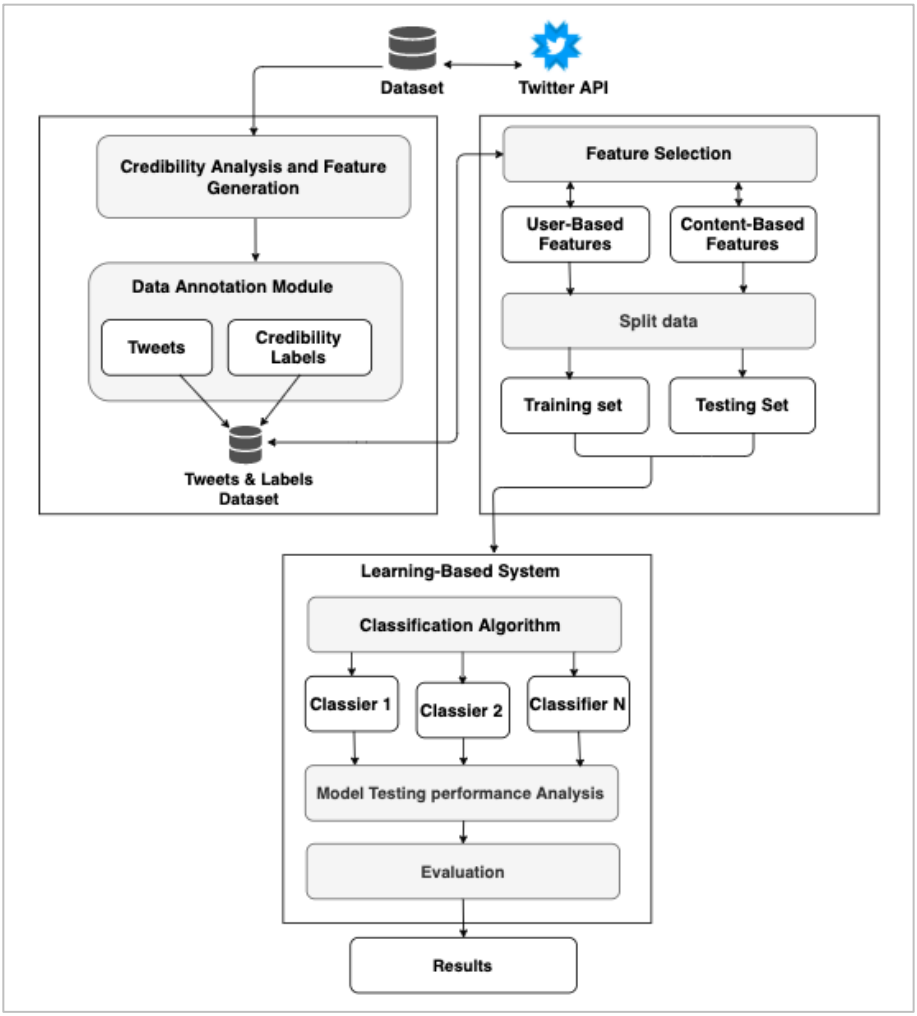

Figure 13. The process of classifying the credibility of tweets

Table 8

The features extracted from the main attributes of the tweets

User-based Features	Content-based Features

Verified user account?	Tweet has slang?
Trusted user source?	Tweet contains Question or Exclamation marks?
User profile description has slang?	Valid tweet length?
User profile description has trusted information?	Tweet engagement ratio
User Follower/Following Ratio	URL in tweet trusted?
	URL in tweet valid?

3.5.2.1 Manual credibility assessment

In order to validate the accuracy and correctness of the automated labeling of the tweets, four participants were recruited to manually evaluate the first 500 tweets of hurricane Florence dataset. This is useful to establish the ground truth for labeling. Each participant was given a copy of the dataset and was given the following guidelines to consider while labeling the data:

User:

- Is the username a trusted source? (e.g. Weather Channel, ABC news)
- Is the user account verified by Twitter? (this can be located under the "Is_verified" column in the dataset)
- Does the user profile description contain slang or swear words? (profiles that contain slang/swear words tend to be not credible)
- Does the user have many followers and friends? And, is the number of followers more the number of friends? (users with followers than friends are considered more credible)

- Does the user have many posts? (Users with high following ratio and actively posts contents are more credible)

Tweet:

- Is the tweet lengthy or wordy? (wordy tweets tend to be credible)
- Does the tweet contain slang or swear words? (Tweets that contain slang/swear words tend to be not credible)
- Does the tweet contain a question mark or exclamation mark? (Tweets that contain "?" or "!" tend to be not credible)
- Does the tweet have many likes? (likeable tweets tend to be credible)
- Does the tweet have many retweets? (retweeted tweets tend to be credible)

After the dataset was labeled by the participants, the labeling was compared in order to:

- Measure the accuracies of the automated labeling method.
- Use the manually labeled dataset as ground truth, to train supervised machine learning classifiers, and evaluate the prediction performance.

3.6 Panic Trigger Identification Framework (PTIF)

After the disaster-related tweets were identified and classified and their credibility levels were determined**,** PTIF was implemented to identify panic triggers and indicators that can cause unwanted consequences if they were acted upon and not mitigated by the emergency responders.

Indicators data may be categorized into predictive (uncertain data indicator) and actionable (certain data indicator) (Stroud et al., 2013). Actionable indicators are usually scripted directly in the tweet content and require less analysis before taking evacuation actions; however, predictive (unscripted indicators) require interpretation of the data before taking actions. For example, "expected water rising in a certain area..." may imply that there will be "Flooding". Understanding

these characteristics of indicators helps inform decisions about how best to use them. Table 9 shows example of actionable and predictive triggers and indicators.

Table 9

Examples of actionable and predictive panic triggers and indicators

Predictive (uncertain data indicator)	Actionable (certain data indicator)
- Hurricane A is approaching the southern area… - Weather update: the speed of wind is increasing drastically… - Local hospitals are receiving a large number of cases in critical situations… - a person found stuck in a certain area... - weather update: expected water rising in a certain area...	- people must evacuate immediately… - no gas... - no water… - water shortage… - no power… - flooded areas… - Bridge breakage… - find shelters immediately… - food shortage...

3.6.1 Panic triggers collection and dictionary generation

There have not been many research studies that on panic triggers during natural disasters. Therefore, it was challenging to find defined triggers in the literature. In order to collect panic triggers, more than 150 panic triggers have been manually collected, and a panic-trigger-dictionary file was constructed. These triggers have been collected from weather channels, and news reports during natural disasters (Rubin, 2019; MHS, 2006). Usually these news mediums highlight what people panic about and the psychology of people during natural disasters (Heide, 2004; Gantt et

al., 2012). Figure 14 shows a sample of the collected panic triggers generated by Word Cloud Python library.

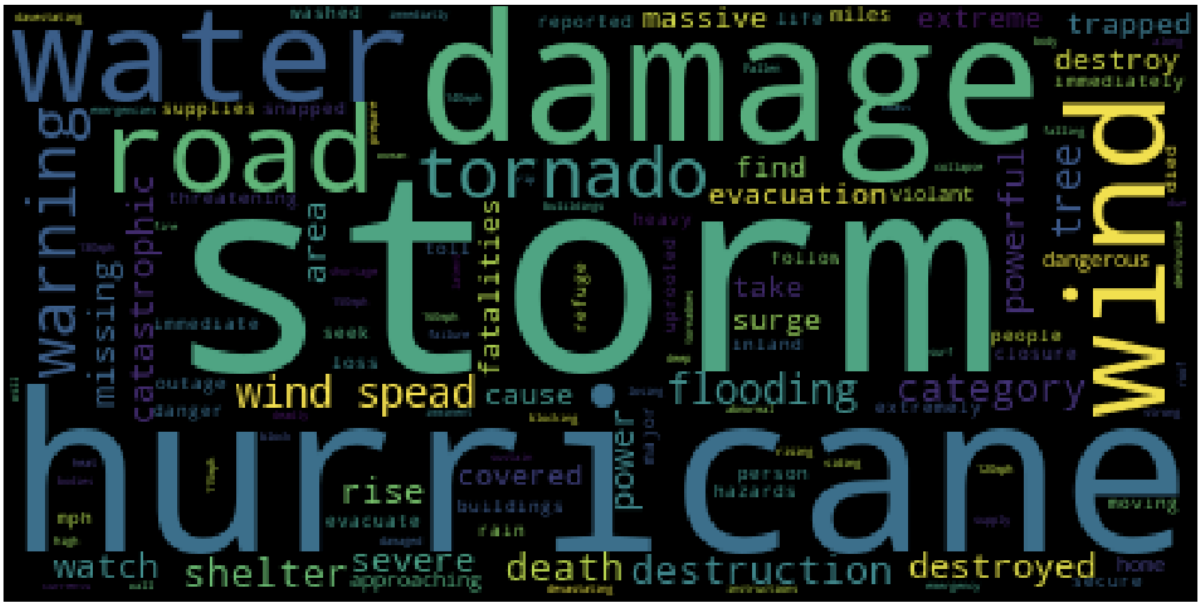

Figure 14. Panic triggers terms collected.

3.6.2 Tweet Analysis for panic triggers

This step needs text analysis; therefore, first, using Natural Language Processing, data preprocessing is conduct in order to clean tweets from unwanted characters explained in section 3.2.1. Once all the tweet texts are processed the framework analyzes each one by checking if the text contains any of the predefined triggers stored in the dictionary, Figure 15 shows the process of checking the tweets for panic triggers and automatically labeling. If the system encounters a trigger within the tweet, the tweet and the trigger it contains are stored. Then the system analyzes the credibility of the tweets that contain the triggers, and conducts the following:

- If the credibility of the tweet X is “credible”, the framework assigns a label “Mitigation”, which means that tweet X is credible and carries useful information, and emergency responders need to take actions to mitigate any actions taken by the public.

- If the credibility of the tweet X is “not credible”, the system assigns a label “Mitigation_and_Correction”, which means that emergency responders need to confirm the correctness of the information contained in the tweet X, and if correct, emergency responders need to take actions to expect and mitigate any actions taken by the public. If the information is incorrect, then emergency responders need to correct the information.

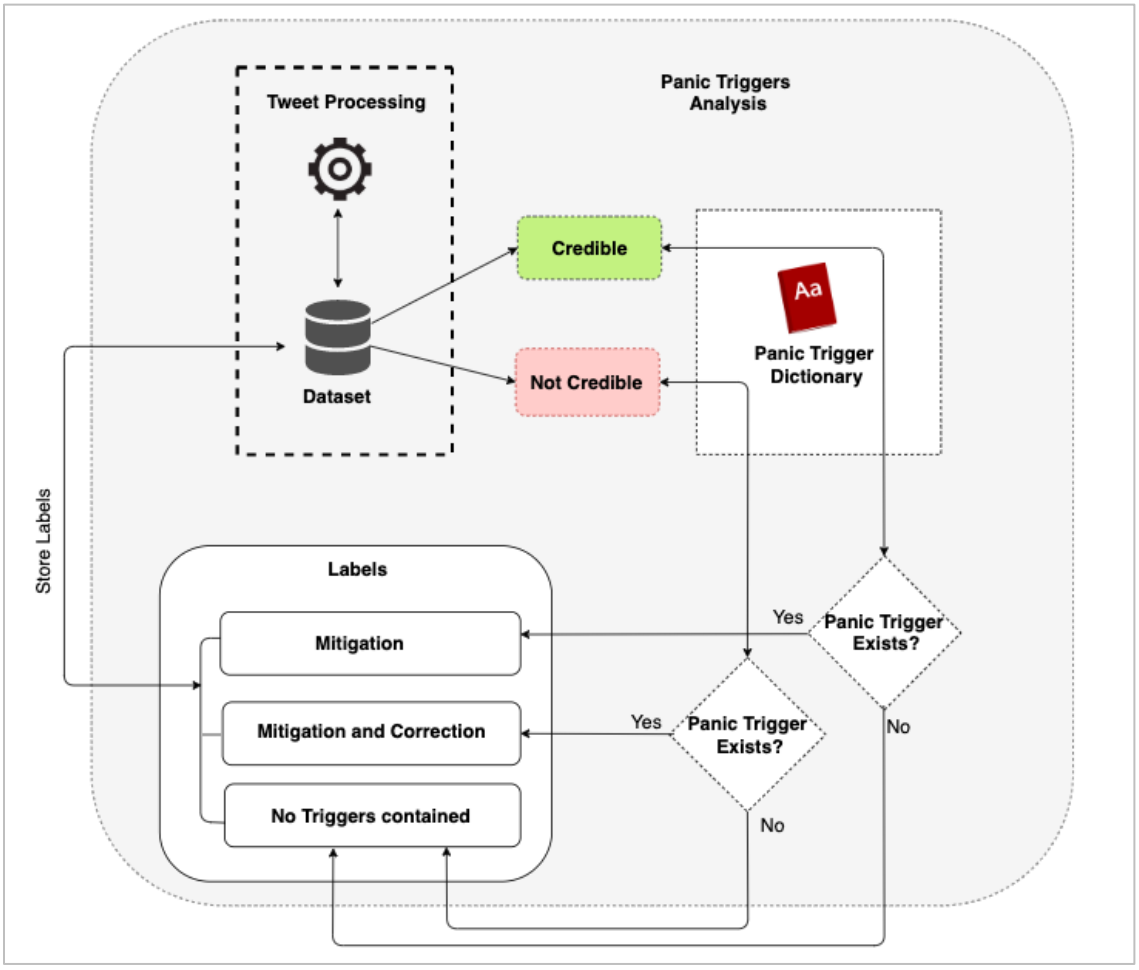

Figure 15. Analyzing the panic triggers in tweets and assigning labels.

3.6.3 Panic tweet classification

After the tweets with were labeled with: “mitigation”, “Mitigation_and_Correction” and “No_Trggiers_Contained”, supervised machine learning classification models were used to classify the tweets.

Machine learning models suitable for classification with discrete features (e.g., word counts, and word frequencies for text classification) such as, K-Nearest Neighbors (KNN), Logistic Regression, Random Forest, and Decision Tree were used. However, the tweet raw data is a sequence of symbols and cannot be fed directly to the classification algorithms as most of these algorithms expect numerical feature vectors with a fixed size rather than the raw text documents with variable length. Therefore, TfidfVectorizer which converts a collection of raw documents to a matrix of TF-IDF features (term frequency–inverse document frequency) was used. TfidfVectorizer reflects how important a word is to a document in a collection. Also, CountVectorizer which implements both tokenization and occurrence counting in a single class was used. These two types of vectorizers produce features that can be used by the classification models, Figure 16 shows the process of classifying the tweets regarding panic triggers using learning-based methods.

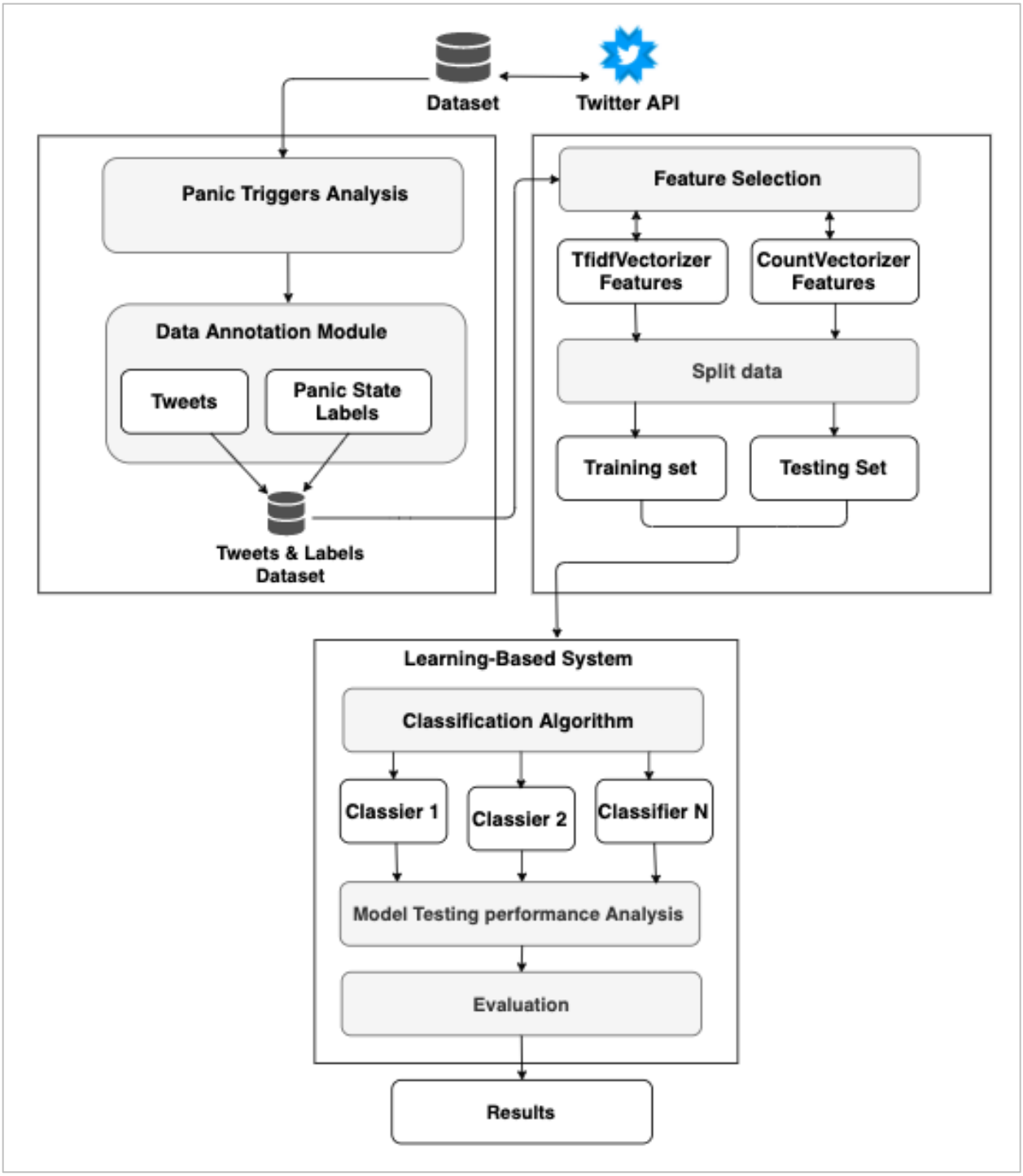

Figure 16. The process of panic tweet feature generation and learning-based classification.

CHAPTER 4

Results

4.1 Historical Data Collection

Using the Tweepy library, historical tweets have been collected using premium Twitter API, preprocessed and stored in datasets. The datasets contain a total of *10,898* tweets about Hurricane Michael and *16,086* tweets about Hurricane Florence. The keyword "hurricane Michael" and "hurricane Florence", were used for searching and importing tweets. The tweets were collected according to pre-landfall, landfall, and post-landfall for each disaster. For the purpose of identifying disaster related events, evaluating the tweet credibility, and identifying panic triggers, specific attributes and entities were extracted for each tweet object. Table 10. Shows the attributes extracted from the tweet objects during the data collection process.

Table 10

Attributes extracted and stored in the hurricane datasets

User Attributes	**Tweet Attributes**
username	tweet
user_profile_description	URL_in_Tweet
user_screen_name	tweet_created_date
number_of_followers	tweet_source
number_of_friends	number_of_retweets
user_account_created_date	number_of_likes
user_likes_count	length_of_tweet
user_posts_count	hashtags_contained_in_tweet
user_account_verified	

Before storing to the dataset, the tweets have been preprocessed and cleaned from emojis, punctuation, numbers, user mentions, URLs, stop words and so on using Natural Language Processing and Regular Expressions. The end result is a total of two datasets processed, cleaned and ready for the analysis, Table 11 shows an example of one of the dataset records.

Table 11

An exmple of a tweet record stored in the dataset.

Attribute	**Content**
username	WSVN 7 News
user_screen_name	wsvn
user_profile_description	South Florida's #1 News Station! Your 24/7 source for breaking news, @7Weather & @7SportsXtra powered by our digital team. Breaking news? newsdesk@wsvn.com
number_of_followers	381597
number_of_friends	1077
user_account_created_date	9/17/2008 4:23:47 PM
user_likes_count	4748
user_posts_count	130662
user_account_verified	TRUE
tweet	HIGHTECH AID Hurricane season has started, and Miami-Dade County has created teams to use drones with

	the ability to livestream video during search-and-rescue missions. Fire rescue crews gave 7News a demonstration.
URL_in_Tweet	https://wsvn.com/news/local/mdfr-teams-up-with-office-of-emergency-management-to-show-how-drones-will-assist-in-hurricanes/
tweet_created_date	6/4/2019 1:55:00 AM
tweet_source	SocialNewsDesk
number_of_retweets	143
number_of_likes	88
length_of_tweet	241
hashtags_contained_in_tweet	None

4.2 Tweets as Disaster Related and Not Disaster Related

In this process, each tweet is compared with the terms stored in the predefined dictionary of disaster-related keywords. If the tweet includes one or more keywords in the dictionary, it is given the label “Related”. The tweet is given the label “Not_Related” otherwise. Table 12 and Figure 17 show that the disaster related tweets significantly outnumber the non-related tweets. The large number of disaster related tweets indicate that Twitter users become very concern when a disaster occurs, and that Twitter was used as a communication medium to share information about disaster events.

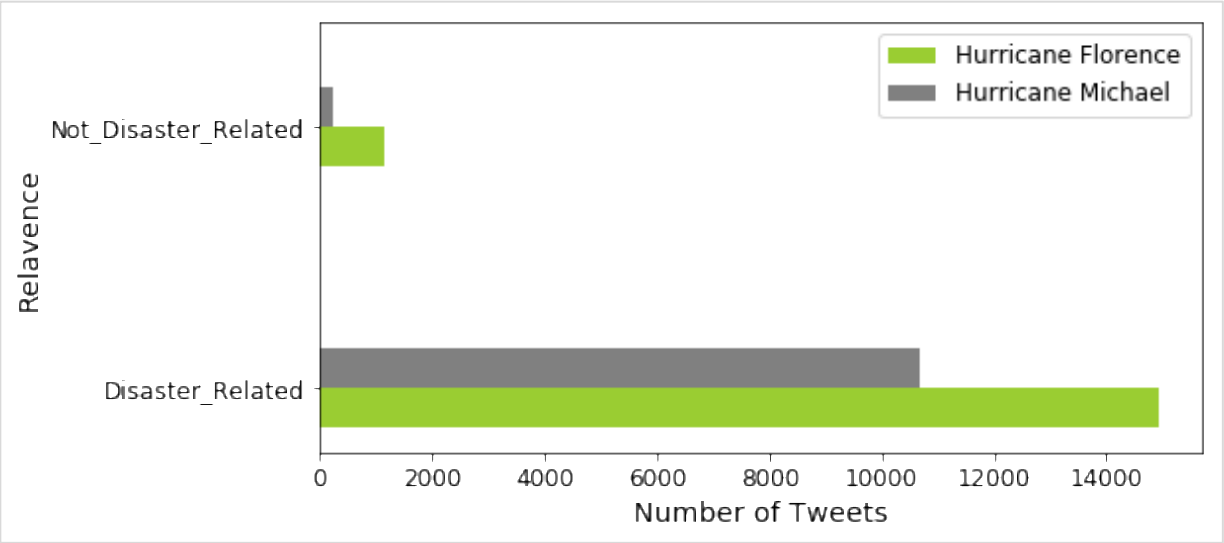

Figure 17. Overall counts of disaster-related and not-related tweets in both datasets.

Table 12

The total of automated labeled disaster tweets

Dataset	Total Tweets	Number (percentage) of Disaster Related Tweets	Number (percentage) of Not Disaster Related Tweets
Hurricane Florence Dataset	16086	14949 (92.9%)	1137 (7.1%)
Hurricane Michael Dataset	10898	10675 (98%)	223%)

4.2.1 Validating data annotation tool

In order to validate the quality of the labeling framework, a team consists of four participants manually and labelled the first *2,000* tweets in each dataset. The team was asked to take notes of the disaster related keywords they come across. These keywords were included in the disaster-related term dictionary. The following aspects were considered for deciding whether a tweet is related to the disaster or not:

- Does the tweet contain information about the Hurricane?

- What is the topic of the tweet?

- What are the hashtags contained in the tweet?

- What is the list of keywords the tweet contains?

The results from manual labeling that were compared with the results of the automatic labeling. The automated labeling achieved 95% accuracy. The dictionary was updated and included the disaster terms that existed in the remaining 5% disaster related tweets that was not identified by the automated labeling. Then automated labeling was run again on the dataset to retest the correctness of the labeling.

4.3 Disaster Data Classification Using Machine Learning

After applying the word vectorizers to both labeled disaster datasets and acquiring the TfidfVectorizer features and CountVectorizer features, these features were fed into machine learning algorithms for classification. In order to evaluate the performance of the classifiers, the accuracy, precision, recall, and f score of the test results were calculated. A good classifier classifies a large amount of data in a short amount of time with high precision and recall scores.

In the experiment, the data was split into 70% training set and 30% test set. The training set contains the known output was used for the training the classifiers. Table 13 to and 14 show the overall classification accuracies of different algorithms using both TfidfVectorizer features and CountVectorizer features.

Table 13

The classification accuracies for Hurricane Florence dataset

Model	Accuracy	
	TfidfVec. Features	CountVec. Features
Logistic Regression	98%	99%
Multinomial Naive Bayes	97%	97%
SVM	98%	99%
KNN	76%	96%
Random Forest	99%	99%
Decision Tree	99%	99%

Table 14

The classification accuracies for hurricane Michael Dataset.

Model	Accuracy	
	Tfidf-Vec. Features	Count-Vec. Features
Logistic Regression	99%	99%
Multinomial Naive Bayes	96%	98%
SVM	99%	99%
KNN	90%	97%
Random Forest	99%	99%
Decision Tree	99%	99%

Most algorithms have high classification accuracies for both TfidfVectorizer features and CountVectorizer features. However, KNN model had the least accuracy for both datasets, especially when using the TfidfVectorizer features. Overall, the accuracies of the models with CountVectorizer features are higher than the accuracies of the models with TfidfVectorizer features.

Table 15, 16, 17, and 18 show the precision, recall, and f-score values for different machine learning algorithms using the two vectorizers for the two datasets. It can be seen that all the models show high precision, recall and f-score values for the tweets in the "Related" class. The precision recall and f-score values for the "Not Related" classes are lower. The reason is that there are a much larger number of disaster-related tweets than Not Related tweets. It was also noticed that the models performed better using CountVectorizer features than using the TfidfVectorizer features. Compared with other models, KNN model has the lowest value for precision, recall and f score, especially when using TfidfVectorizer features.

Table 15

The classification performance for hurricane Michael Dataset Using TfidfVectorizer Features

Model	TfidfVect.					
	Related Tweet Prediction			Not Related Tweet Prediction		
	Pr	Re	F1	Pr	Re	F1
Logistic Regression	0.99	0.99	0.99	0.92	0.87	0.90

Model	TfidfVect.					
	Related Tweet Prediction			Not Related Tweet Prediction		
	Pr	Re	F1	Pr	Re	F1
Multinomial Naive Bayes	0.98	0.99	0.98	0.84	0.81	0.83
SVM	0.99	0.99	0.99	0.95	0.87	0.91
KNN	0.99	0.75	0.85	0.21	0.94	0.35
Random Forest	0.99	0.99	0.99	0.93	0.90	0.92
Decision Tree	0.99	0.99	0.99	0.93	0.99	0.96

Table 16

The classification performance for hurricane Florence Dataset Using TfidfVectorizer Features

Model	CountVect.					
	Related Tweet Prediction			Not Related Tweet Prediction		
	Pr	Re	F1	Pr	Re	F1
Logistic Regression	1.00	0.99	0.99	0.88	0.99	0.93
Multinomial Naive Bayes	0.97	0.99	0.98	0.91	0.68	0.78
SVM	0.99	0.99	0.99	0.89	0.99	0.94
KNN	0.99	0.96	0.97	0.64	0.98	0.77

Random Forest	0.99	0.99	0.99	0.91	0.94	0.92
Decision Tree	0.99	0.99	0.99	0.92	0.96	0.94

Table 17

The classification performance for hurricane Florence Dataset Using CountVectorizer Features

Model	TfidfVect.					
	Related Tweet Prediction			Not Related Tweet Prediction		
	Pr	Re	F1	Pr	Re	F1
Logistic Regression	0.99	0.99	0.99	0.79	0.90	0.84
Multinomial Naive Bayes	0.99	0.97	0.98	0.40	0.83	0.54
SVM	0.99	1.00	0.99	1.00	0.87	0.93
KNN	1.00	0.90	0.95	0.19	0.98	0.32
Random Forest	0.99	0.99	0.99	0.91	0.93	0.92
Decision Tree	0.99	1.00	1.00	1.00	0.95	0.97

Table 18

The classification performance for hurricane Michael Dataset Using CountVectorizer Features

Model	CountVect.					
	Related Tweet Prediction			Not Related Tweet Prediction		
	Pr	Re	F1	Pr	Re	F1
Logistic Regression	1.00	0.99	0.99	0.88	0.98	0.93
Multinomial Naive Bayes	0.99	0.99	0.99	0.89	0.58	0.71
SVM	0.99	0.99	0.99	0.93	0.95	0.94
KNN	1.00	0.97	0.98	0.51	0.98	0.67
Random Forest	0.99	1.00	0.99	0.98	0.87	0.92
Decision Tree	0.99	1.00	0.99	0.98	0.95	0.97

It is important to interpret a classifier with its structure of Receiver Operating Characteristic (ROC) curve and Area Under Curve (AUC) scores. The ROC curves show the prediction success of the models. The framework implemented ROC curves were plotted and AUC scores were calculated to summarize the performance of a classifier over all possible thresholds. It is generated by plotting the True Positive Rate TPR (y-axis) against the False Positive Rate FPR (x-axis) as the thresholds were varied for assigning observations to a given class. The plot of TPR (sensitivity) versus FPR (1-specificity) across varying cut-offs generates a curve in the unit square called ROC curve. ROC curve corresponding to progressively greater discriminant capacity of diagnostic tests

are located progressively closer to the upper left-hand corner in ROC area (Hajian-Tilaki, 2013). From Figure 18 and Figure 21, we can see that KNN classifier has the least discriminant capacity than other models, especially when using TfidfVectorizer features. The models have a greater discriminant capacity of diagnostic tests with CountVectorizer features than with TfidfVectorizer features.

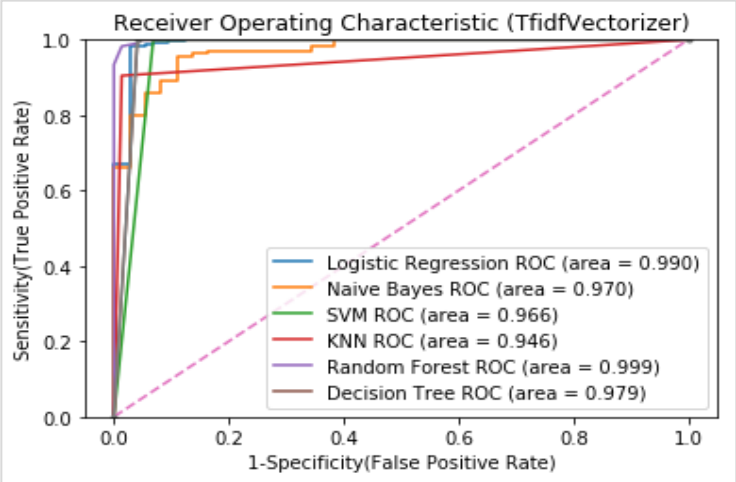

Figure 18. Overall classifiers ROC performance for Hurricane Florence dataset using CountVectorizer.

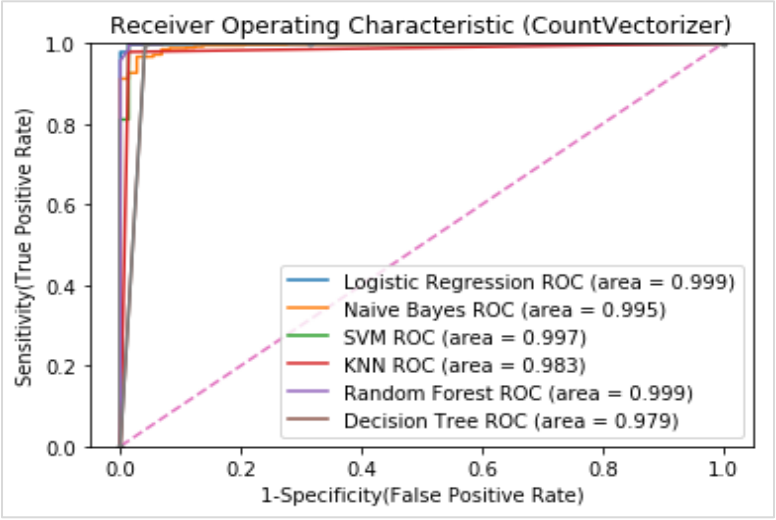

Figure 19. Overall classifiers ROC performance for Hurricane Florence dataset using TfidfVectorizer.

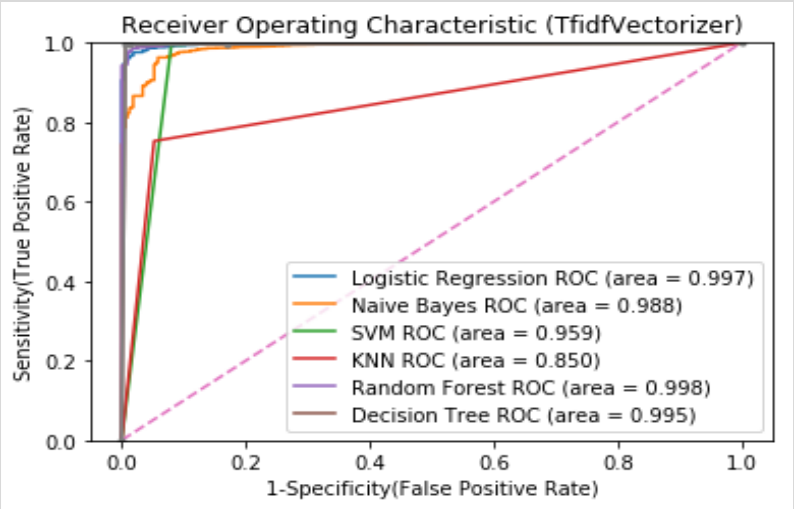

Figure 20. Overall classifiers ROC performance for Hurricane Michael dataset using TfidfVectorizer.

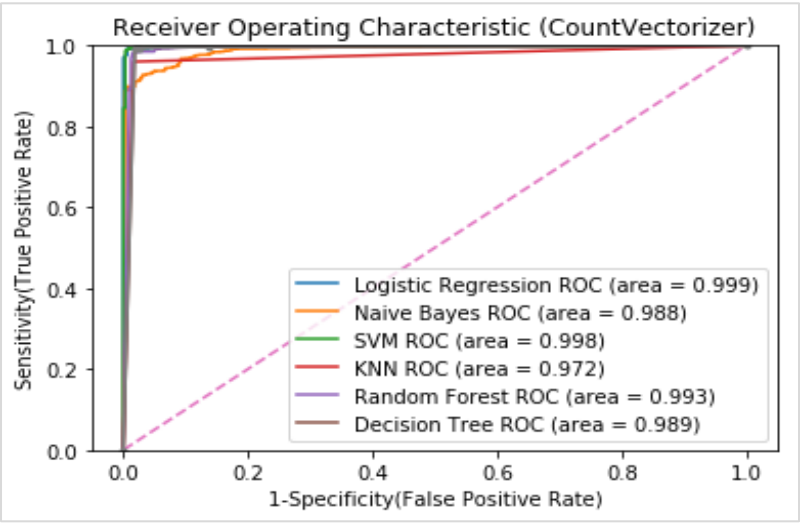

Figure 21. Overall classifiers ROC performance for Hurricane Michael dataset using CountVectorizer.

4.4 Tweet Credibility Analysis

After categorizing the tweets into disaster-related and not disaster related tweets, the credibility of disaster-related tweets was analyzed. User-based features and content-based features were used to determine the credibility of the tweets as shown in Table 19.

Table 19

User-based features and content-based features

User-based Features	**Data Type**	**Content-based Features**	**Data Type**
Verified user account?	Boolean	Tweet has slang?	Boolean
Trusted user source?	Boolean	Tweet contains Question or Exclamation marks?	Boolean
User profile description has slang?	Boolean	Valid tweet length?	Boolean
User profile description has trusted information?	Boolean	Tweet engagement ratio	Float
User follower/following ratio	Float	URL in tweet trusted?	Boolean
		URL in tweet valid?	Boolean

Based on these features, the credibility score was calculated, and the credibility label was determined. The credibility score has a value from 0 to 10. A tweet with a credibility score from 0 to 4 is considered not credible, and a tweet with a score from 5 to 10 is considered credible.

Figure 22 shows the number of credible and not credible tweets in the two datasets. As we can see from the Figure 22, the number of credible tweets in hurricane Florence dataset is

higher than the one in hurricane Michael datasets. Overall, the non-credible tweets appear to be higher than the credible ones in both datasets.

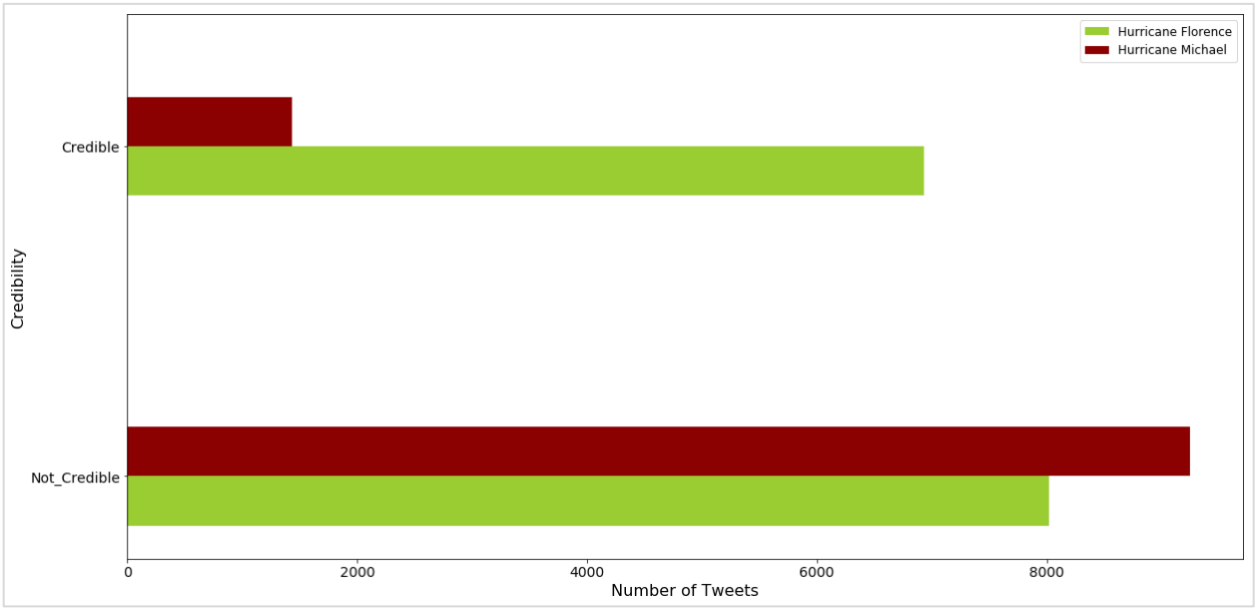

Figure 22. The number of credible and non-credible tweets in both datasets

4.4.1 Manual credibility assessment

In order to validate the accuracy and correctness of the framework for labeling credibility, three participants were recruited to evaluate the first 500 tweets of hurricane Florence dataset. This was useful to establish the ground truth for labeling.

The results of the manual labeling is similar to the results of the automated labeling framework as shown in Figure 23 to 25.

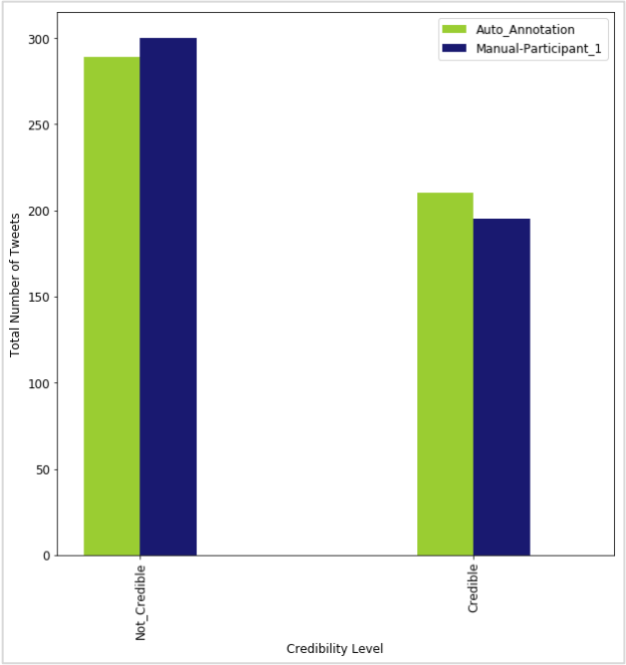

Figure 23. A comparison between the results of the automated labeling and manual labeling by participant_1.

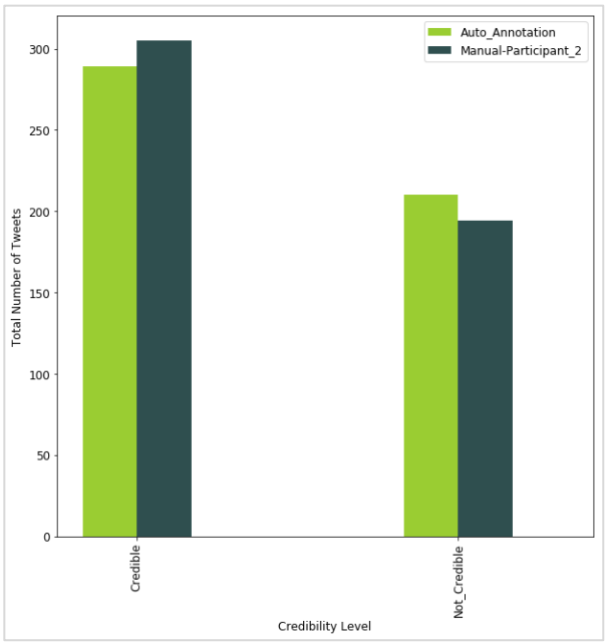

Figure 24. A comparison between the results of the automated labeling and manual labeling by participant_2.

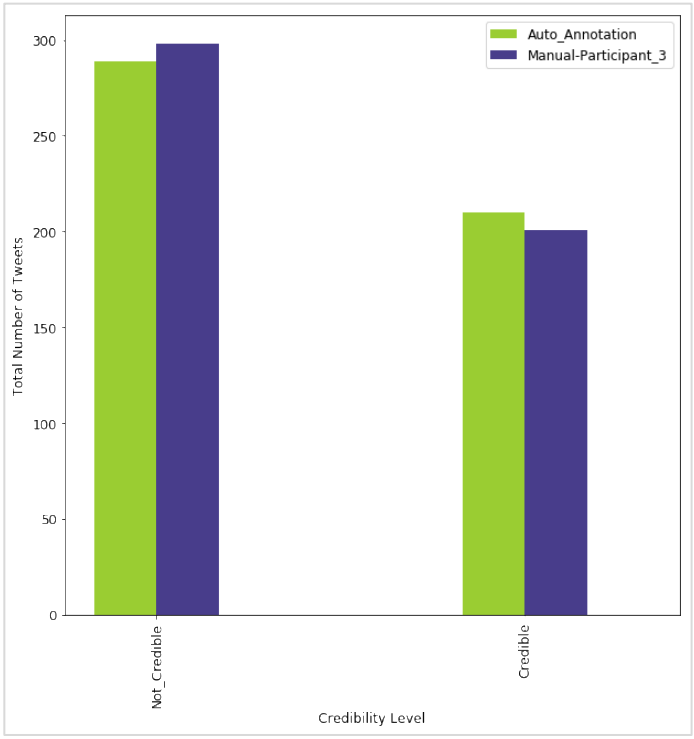

Figure 25. A comparison between the results of the automated labeling and manual labeling by participant_3.

4.5 Credibility Classification Using Machine Learning

Once credibility labels for each tweet were obtained, supervised machine learning was conducted to classify the credibility of the tweets.

4.5.1 Experiments with different machine learning algorithms

For both hurricane Florence and Michael datasets labeled using the automated labeling, the data was split into 70% training set and 30% test set. The training set was used to train the classification models, and the models were used to predict the classification of the test data. Table 20 and Table 21 show the performance metrics of the algorithms used in the experiment. For each the classification model the following features were used:

```
features = hurricane_dataset['account_Verified','trusted_username',
'tweet_contains_Q_E_chars', 'valid_tweet_length', 'tweet_has_slang',
```

```
'descr_has_slang', 'trusted_user_desc', 'user_following_ratio',
 'tweet_engagement_ratio',      'valid_url',      'trusted_url_source',
'Credibility']
```

All these features contain Boolean values (True, False), except for 'user_following_ratio', and 'tweet_engagement_ratio' which are Float values produced by the automated labeling.

Table 20

The credibility classification performance metrics for Hurricane Michael dataset

Classifier	Labels	Precision	Recall	F-score	Accuracy
SVM	Credible	0.93	0.85	0.89	0.97
	Not Credible	0.98	0.99	0.98	
KNN	Credible	0.93	0.93	0.93	0.98
	Not Credible	0.99	0.99	0.99	
Decision Tree	Credible	0.99	0.99	0.99	0.99
	Not Credible	1.00	1.00	1.00	
Random Forest	Credible	0.99	0.99	0.99	0.99
	Not Credible	1.00	1.00	1.00	
	Credible	0.92	0.35	0.51	0.92

Logistic Regression	Not Credible	0.92	1.00	0.96	

Table 21

Credibility Classification performance for Hurricane Florence dataset

Classifier	Labels	Precision	Recall	F-score	Accuracy
SVM	Credible	0.96	0.92	0.94	0.98
	Not Credible	0.98	0.99	0.99	
KNN	Credible	0.95	0.98	0.95	0.98
	Not Credible	0.99	0.99	0.99	
Decision Tree	Credible	1.00	0.99	1.00	0.99
	Not Credible	1.00	1.00	1.00	
Random Forest	Credible	1.00	1.00	0.99	0.99
	Not Credible	1.00	1.00	1.00	
Logistic Regression	Credible	0.92	0.54	0.68	0.91
	Not Credible	0.92	0.99	0.95	

Form the Table 20 and Table 21, we can see that the classifiers used in the experiment have produced very high classification accuracies based on the given features. The results show that decision-based algorithms like Decision tree and Random Forest have the best performance. However, Logistic Regression has shown the worst performance with an accuracy of 92% for classifying hurricane Michael dataset, and 91% for Florence dataset. The performance of the algorithms in classifying hurricane Florence dataset is higher than classifying hurricane Michael dataset. The reason may be that the dataset contains more tweets which means there are more instances in the training set and test set.

Figure 26 to Figure 35 show the confusion matrix for each classification model. We can see specifically how many instances were classified correctly and how many were classified incorrectly. Based on these instances' classification the Precision, Recall, F-score, and the overall accuracy were calculated. As we can see, all the models can correctly classify most of the tweets.

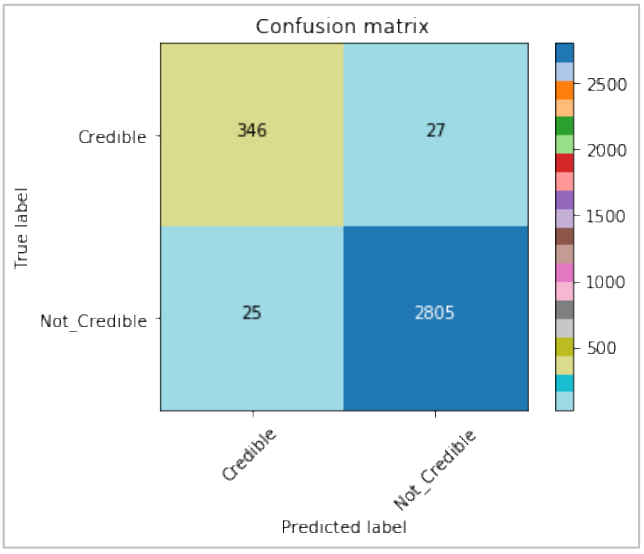

Figure 26. Confusion matrix for KNN model for credibility classification for hurricane Michael dataset

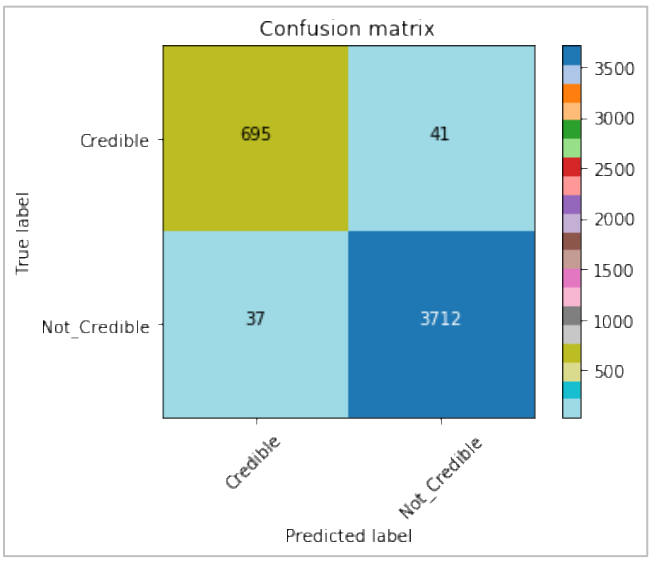

Figure 27. Confusion matrix for KNN model for credibility classification for hurricane Florence dataset

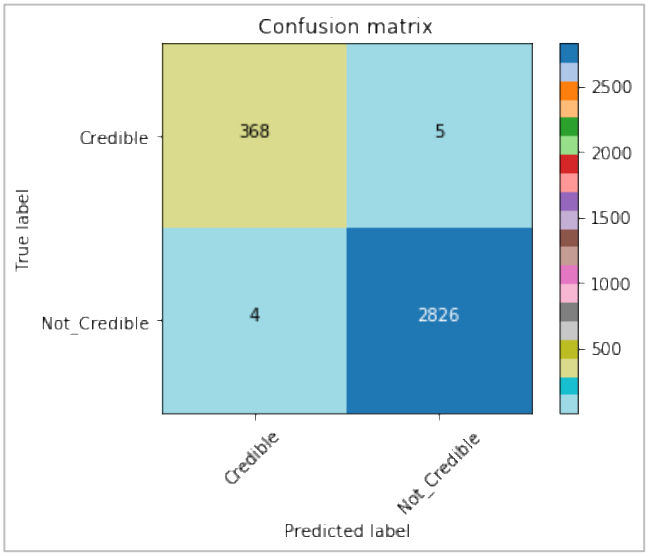

Figure 28. Confusion matrix for Decision Tree model for credibility classification for hurricane Michael dataset

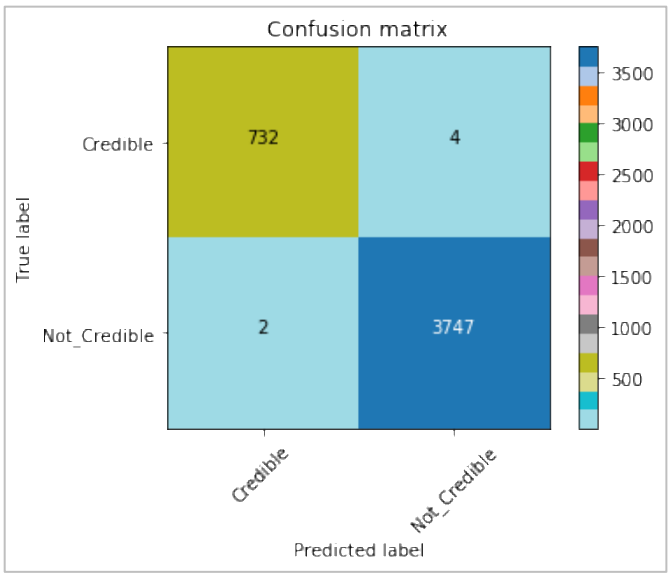

Figure 29. Confusion matrix for Decision Tree model for credibility classification for hurricane Florence dataset

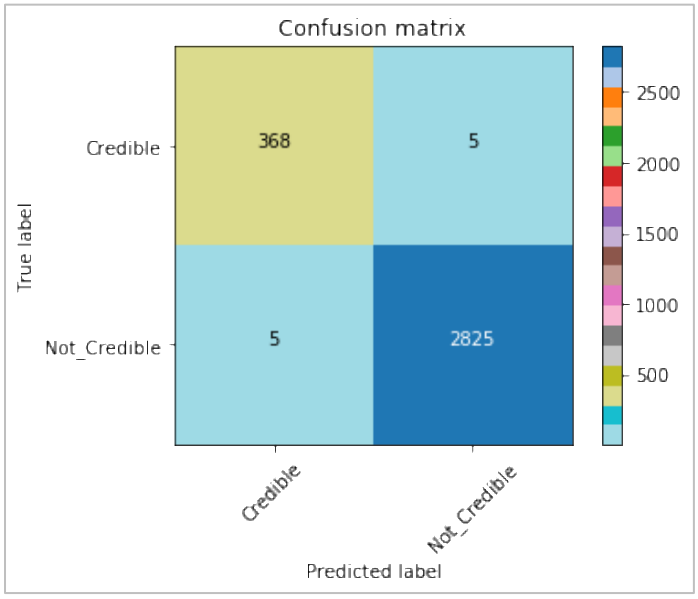

Figure 30. Confusion matrix for Random Forest model for credibility classification for hurricane Michael dataset

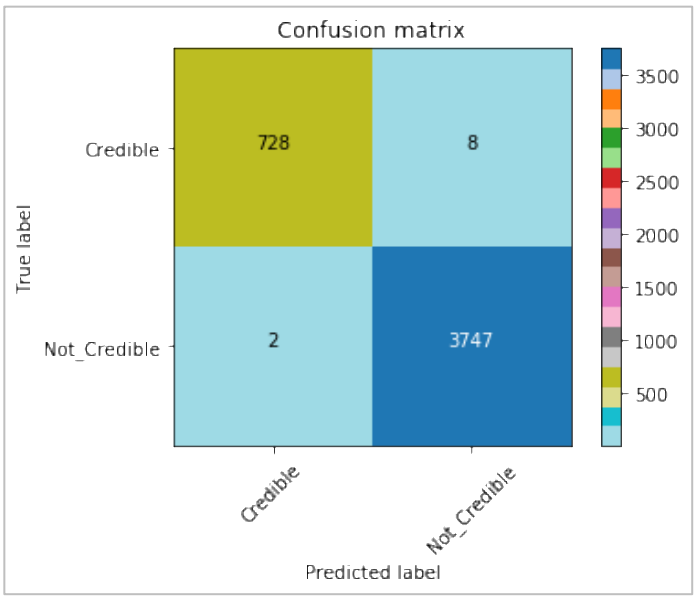

Figure 31. Confusion matrix for Random Forest model for credibility classification for hurricane Florence dataset

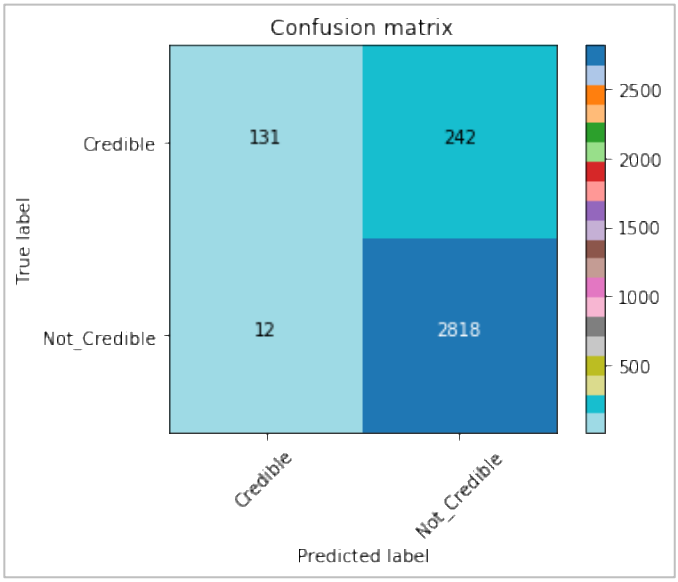

Figure 32. Confusion matrix for Logistic Regression model for credibility classification for hurricane Michael dataset

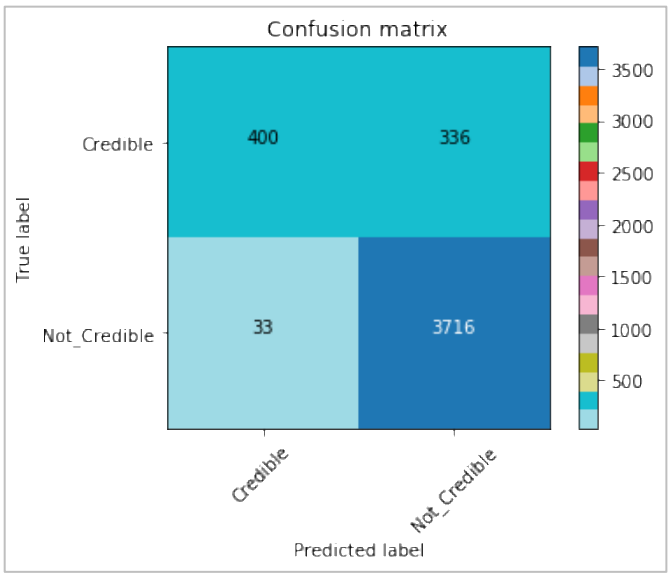

Figure 33. Confusion matrix for Logistic Regression model for credibility classification for hurricane Florence dataset

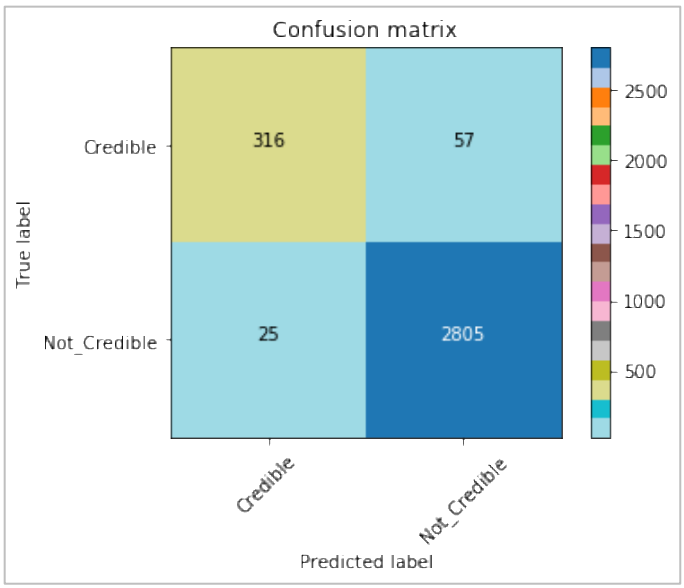

Figure 34. Confusion matrix for SVM model for credibility classification for hurricane Michael dataset

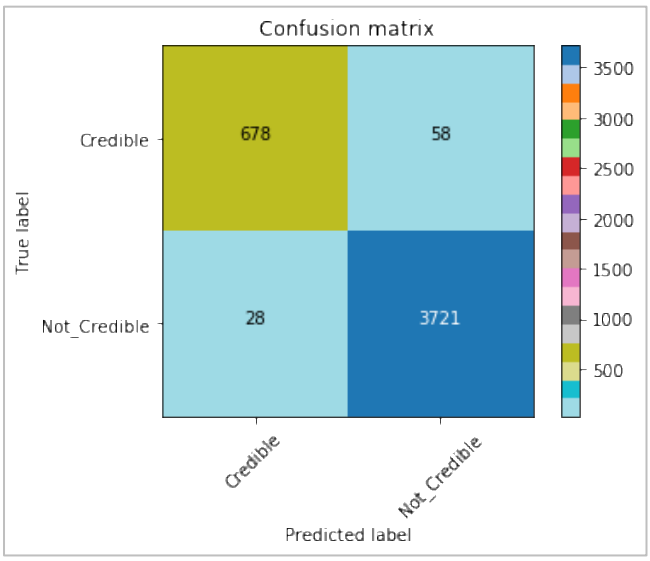

Figure 35. Confusion matrix for SVM model for credibility classification for hurricane Florence dataset

Next, False Acceptance Rate (FAR), False Rejection Rate (FRR), and Equal Error Rate (EER) were calculated as shown in Table 22 and Table 23. The Receiver Operating Characteristic (ROC) plots were drawn to compare the algorithms. The ROC curve plots the True Positive Rate (TPR) against the False Negative Rate (FNR). High sensitivity (TPR) means that the model has a good capacity to detect the positives instances. High specificity (1 - FNR) is shows that the model can detect most of the negative instances.

We can summarize from table 22 and table 23 that decision tree and random forest models have the lowest ERR. Also, they show high sensitivity and specificity rates in comparison to other classifiers, Figure 38 to Figure 47. Also, Logistic regression shows the lowest accuracy with higher ERR, and low sensitivity and specificity rates for both datasets (Figure 36 and Figure 37).

Table 22

Credibility Classification FAR, FRR, and EER rates for Hurricane Florence dataset

Classifier	Labels	FAR	FRR	EER
SVM	Credible	0.007	0.078	0.024
	Not Credible	0.078	0.007	0.024
KNN	Credible	0.0098	0.055	0.018
	Not Credible	0.055	0.009	0.018
Decision Tree	Credible	0.0005	0.005	0.005
	Not Credible	0.005	0.0005	0.005
Random Forest	Credible	0.0005	0.010	0.002
	Not Credible	0.010	0.0005	0.002
Logistic Regression	Credible	0.008	0.45	0.14
	Not Credible	0.145	0.008	0.14

Table 23

Credibility Classification FAR, FRR, and ERR rates for Hurricane Michael dataset

Classifier	Labels	FAR	FRR	EER
SVM	Credible	0.15	0.008	0.031
	Not Credible	0.008	0.15	0.031
KNN	Credible	0.072	0.008	0.018
	Not Credible	0.008	0.072	0.018
Decision Tree	Credible	0.013	0.0014	0.013
	Not Credible	0.0014	0.014	0.013
Random Forest	Credible	0.013	0.0017	0.0058
	Not Credible	0.0017	0.013	0.005
Logistic Regression	Credible	0.64	0.004	0.30
	Not Credible	0.30	0.64	0.30

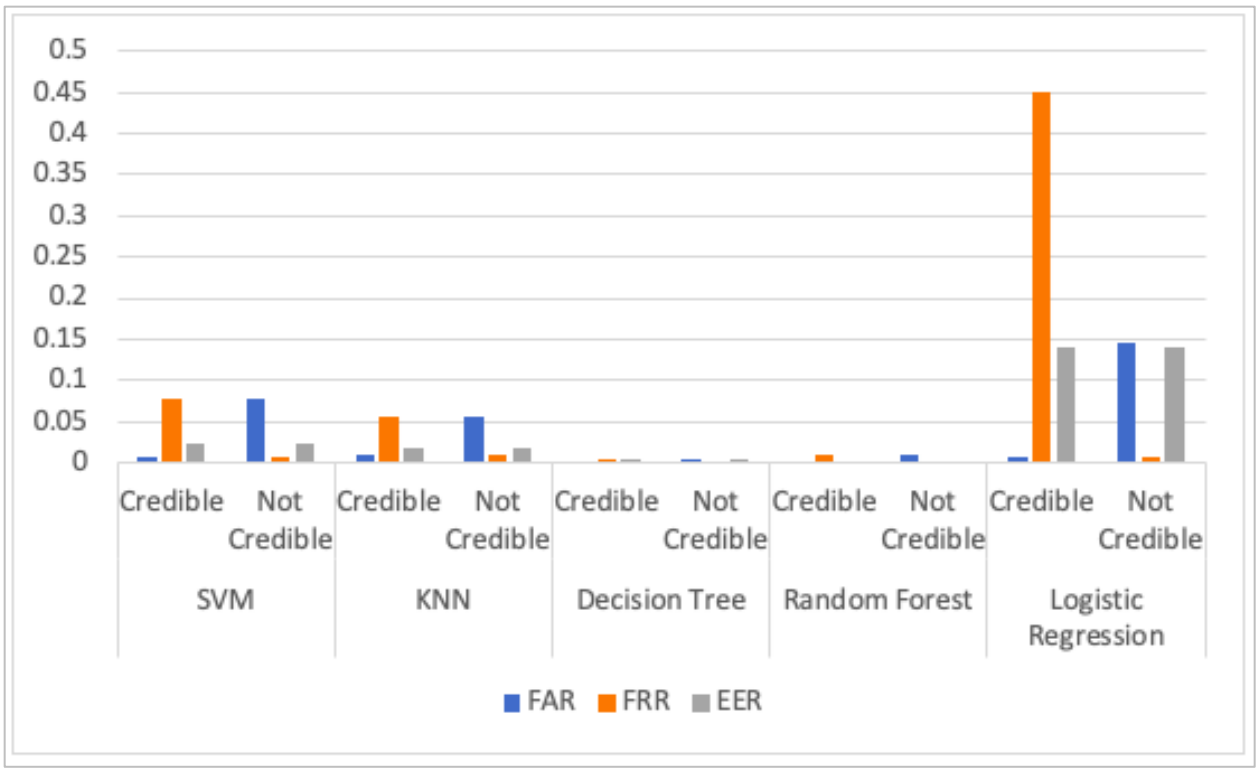

Figure 36. Credibility Classification FAR, FRR, and EER rates for Hurricane Florence dataset

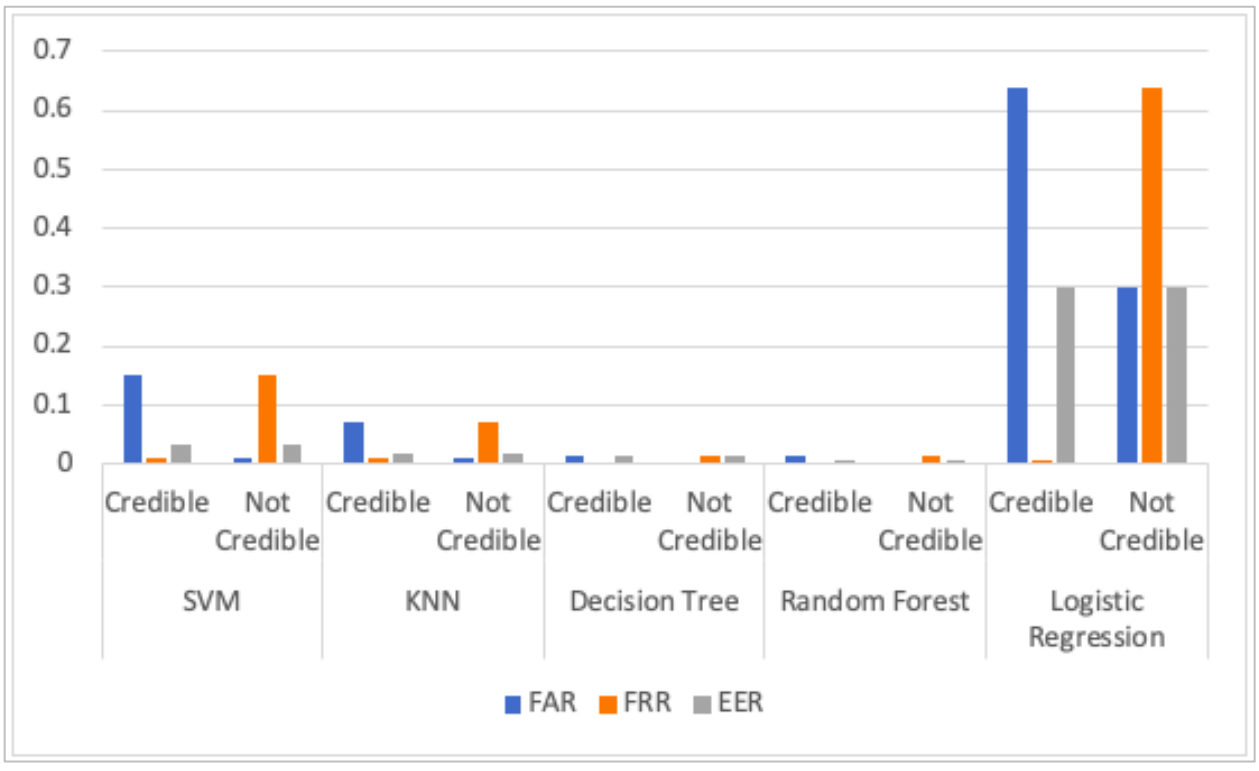

Figure 37. Credibility Classification FAR, FRR, and EER rates for Hurricane Michael dataset

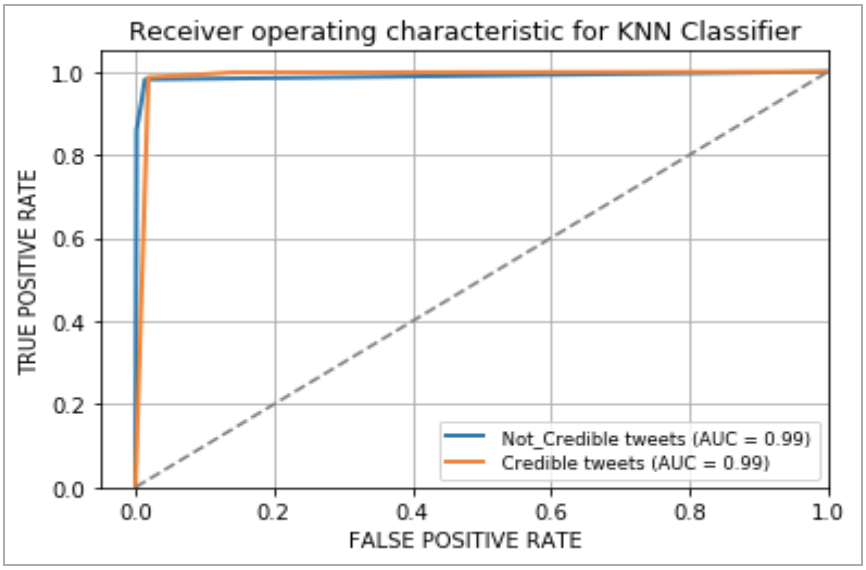

Figure 38. KNN classifier ROC performance for Hurricane Florence dataset

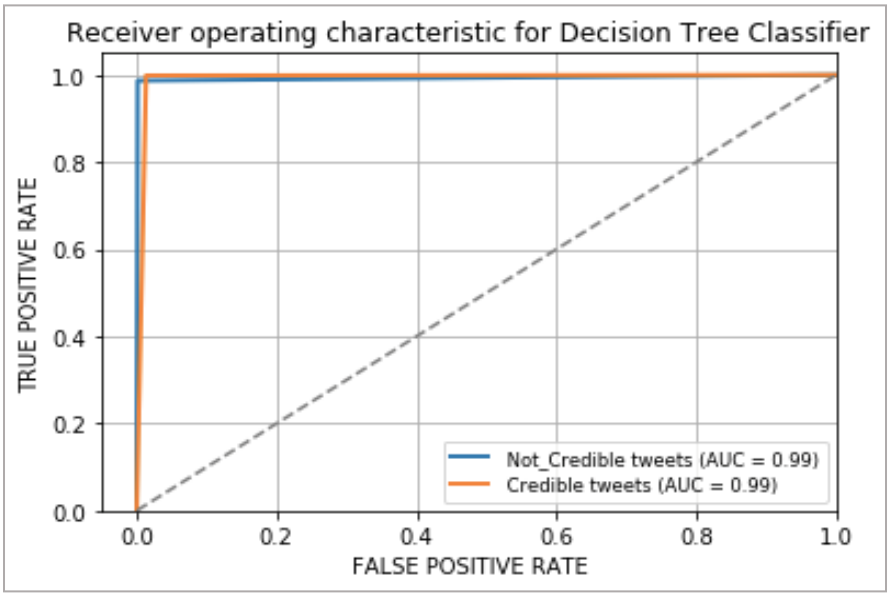

Figure 39. Decision Tree classifier ROC performance for Hurricane Florence dataset

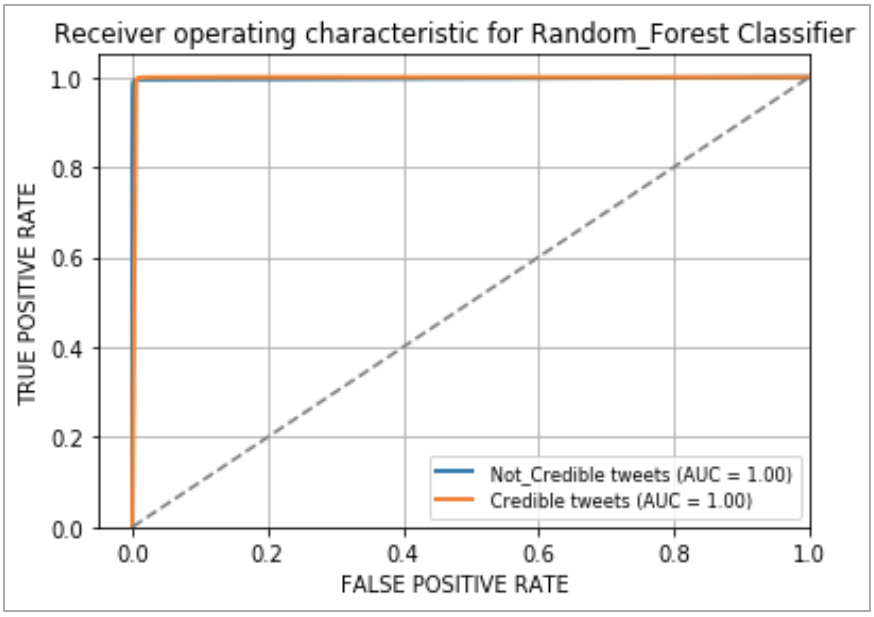

Figure 40. Random Forest classifier ROC performance for Hurricane Florence dataset

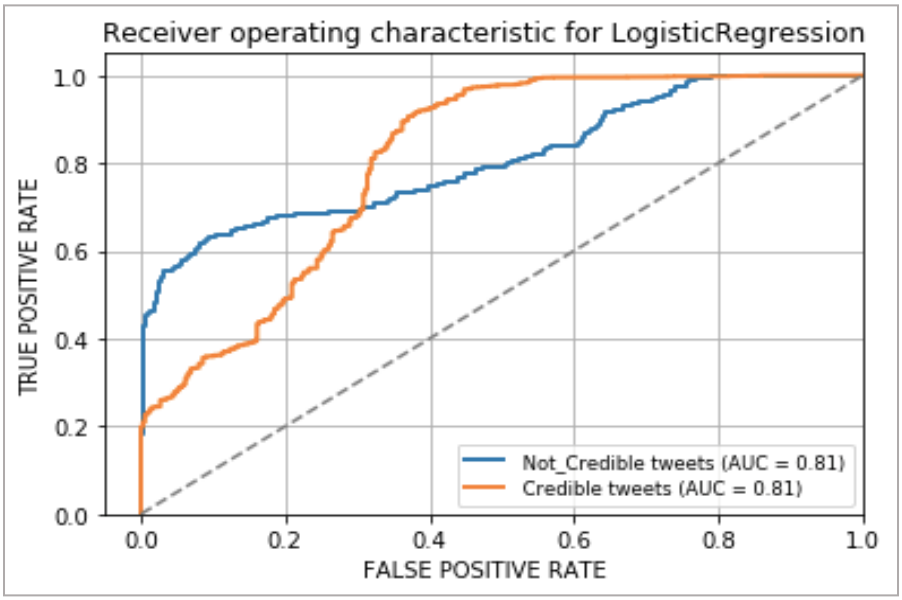

Figure 41. Logistic Regression classifier ROC performance for Hurricane Florence dataset

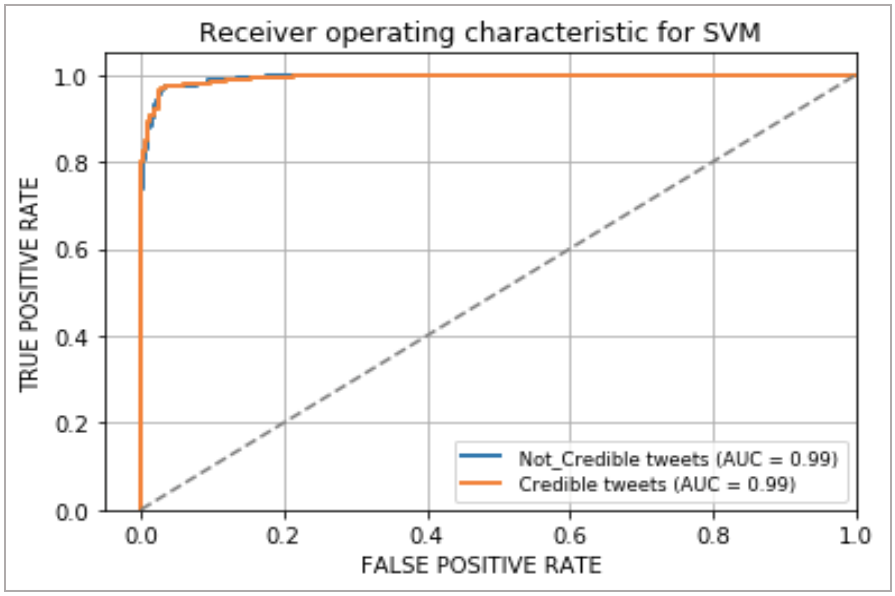

Figure 42. SVM classifier ROC performance for Hurricane Florence dataset

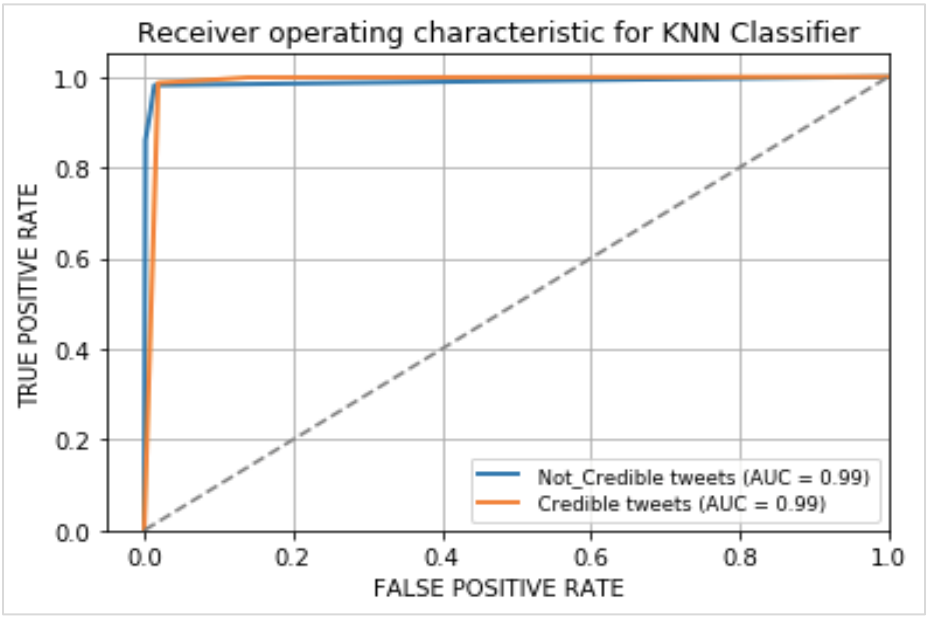

Figure 43. KNN classifier ROC performance for Hurricane Michael dataset

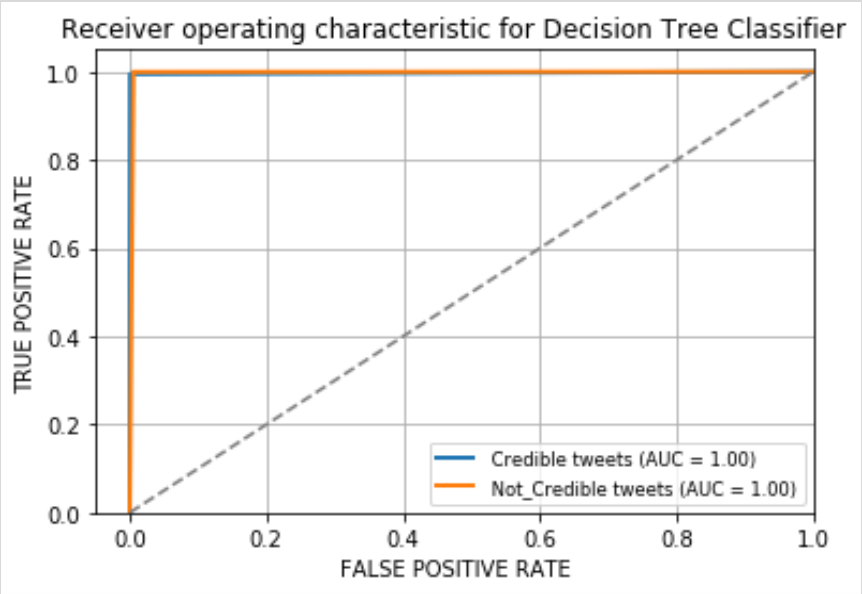

Figure 44. Decision Tree classifier ROC performance for Hurricane Michael dataset

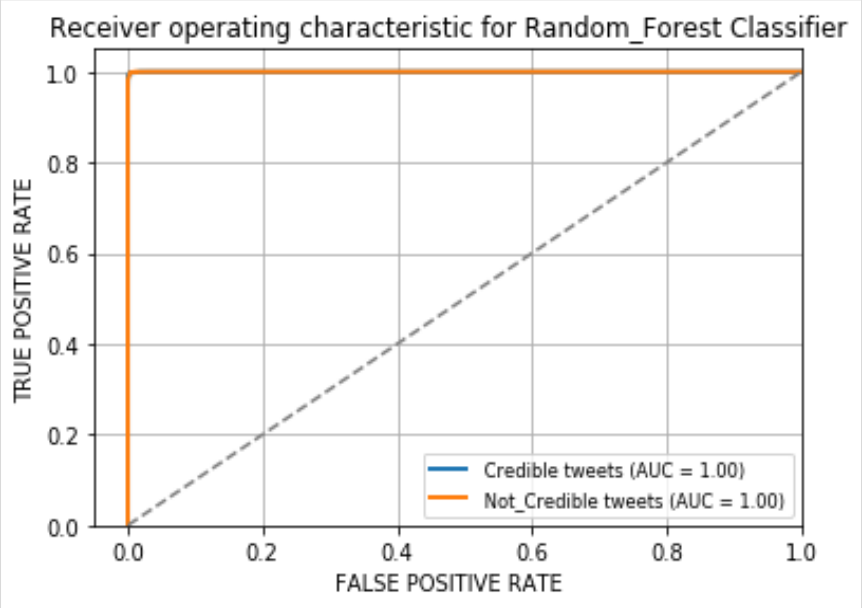

Figure 45. Random Forest classifier ROC performance for Hurricane Michael dataset

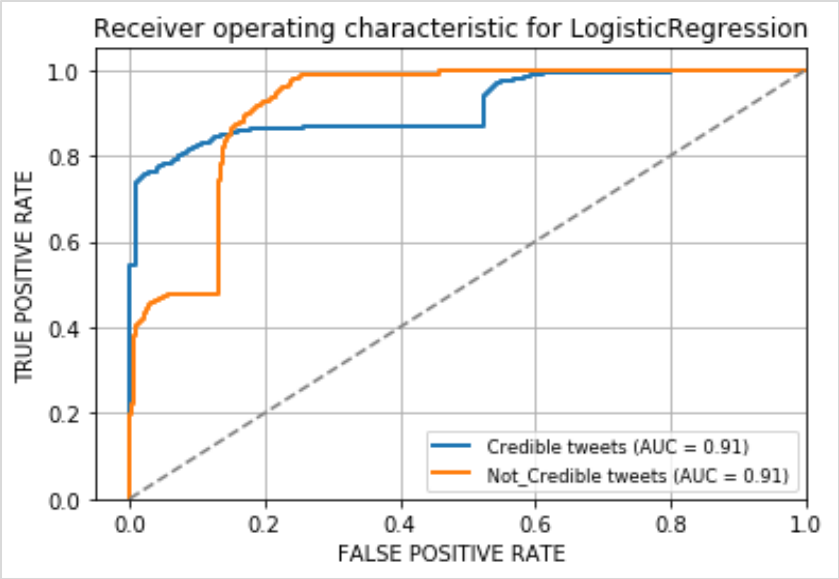

Figure 46. Logistic Regression classifier ROC performance for Hurricane Michael dataset

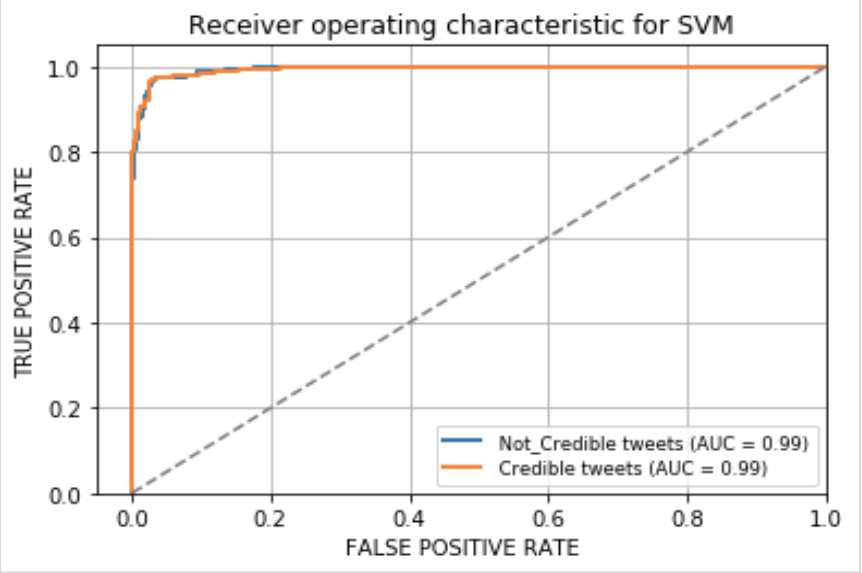

Figure 47. SVM classifier ROC performance for Hurricane Michael dataset

4.5.2 Using manually labeled dataset as a test set

In this experiment, manually labeled dataset was used as a test set. Features that was not included in manual labeling like 'Valid_URL' and 'Trused_URL_source' were removed. Then the performance of the classification models was evaluated to measure that correctness of the labels that was produced by the automated labeling framework. Only the first 500 tweets of the hurricane

dataset labeled using the automated labeling was used as training set, and the 500 manually labeled tweets were used as a test set. After that, machine learning classification was implemented to analyze and compare the outcomes of the classification model predictions. The training set contained the known output as ground truth, and the classification algorithms use this data for learning. Hence, the performance of the models was measured to highlight which model performed the best and produced optimal accuracy. Table 24 shows the overall classification accuracies of the algorithms used in the study. For the classification model, the following features for the training dataset were used:

```
features = hurricane_dataset ['account_verified','trusted_username',
'tweet_contains_Q_E_chars', 'valid_tweet_length', 'tweet_has_slang',
'descr_has_slang', 'trusted_user_desc', 'user_following_ratio',
 'tweet_engagement_ratio', 'Credibility_Level' ]
```

All these features contain Boolean values (True, False) except for 'user_following_ratio', and 'tweet_engagement_ratio' which contain Float values. Table 24 shows the performance metrics obtained for each classifier for classifying hurricane Florence.

Table 24

Credibility Classification performance for Hurricane Florence dataset

Classifier	Labels	Precision	Recall	F-score	Accuracy
SVM	Credible	0.90	0.80	0.89	0.89
	Not Credible	0.88	0.95	0.91	

KNN	Credible	0.90	0.90	0.90	0.92
	Not Credible	0.93	0.93	0.93	
Decision Tree	Credible	0.92	0.93	0.92	0.95
	Not Credible	0.96	0.96	0.96	
Random Forest	Credible	0.95	0.98	0.97	0.97
	Not Credible	0.99	0.97	0.98	
Logistic Regression	Credible	0.93	0.69	0.80	0.87
	Not Credible	0.83	0.97	0.89	

Table 24 shows that data labeled by the automated labeling framework can be sufficient. The majority of the classifiers produced high accuracies. The decision-based algorithms like Random Forest and Decision tree preformed best with the classification accuracies of 97% and 95% respectively. However, Logistic Regression has shown the least given accuracy performance with an accuracy of 87% for classifying the manually labeled data.

Figure 48 to Figure 52 show the confusion matrix for each classification model. We can see specifically how many instances were classified correctly and how many were classified incorrectly. Based on the confusion matrix, the Precision, Recall, F-score, and the overall accuracy were calculated.

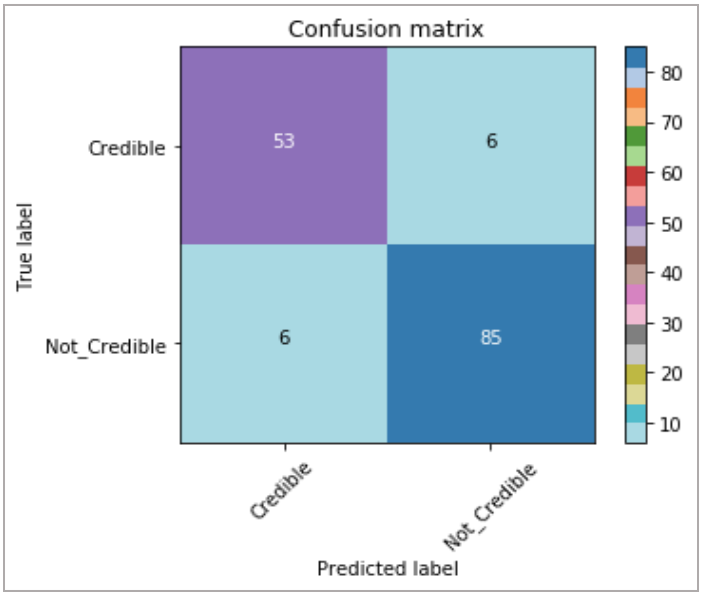

Figure 48. Confusion matrix for KNN model for classifying manually labeled dataset

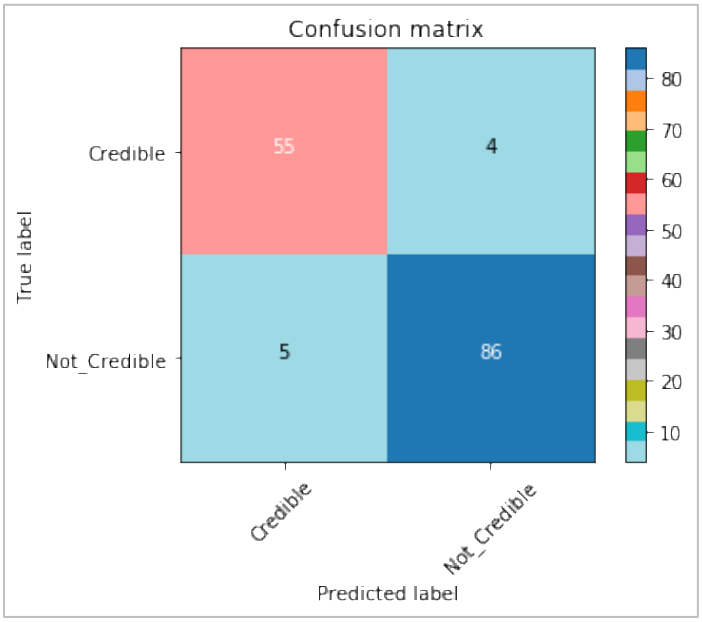

Figure 49. Confusion matrix for Decision Tree model for classifying manually labeled dataset

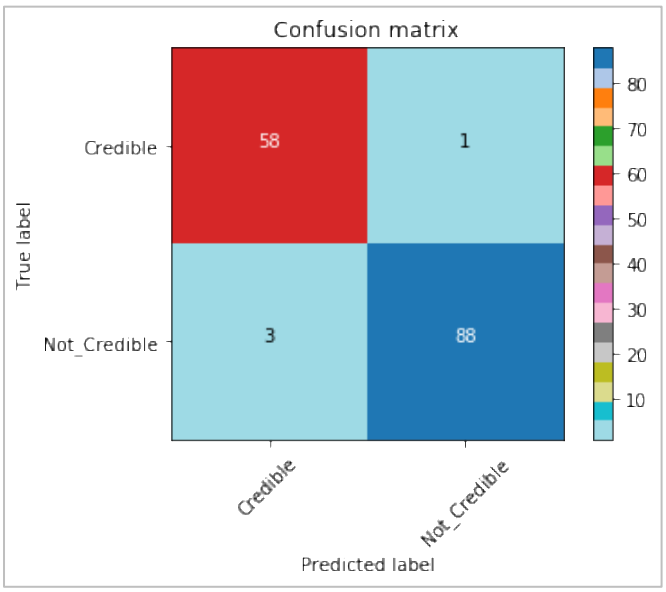

Figure 50. Confusion matrix for Random Forest model for classifying manually labeled dataset

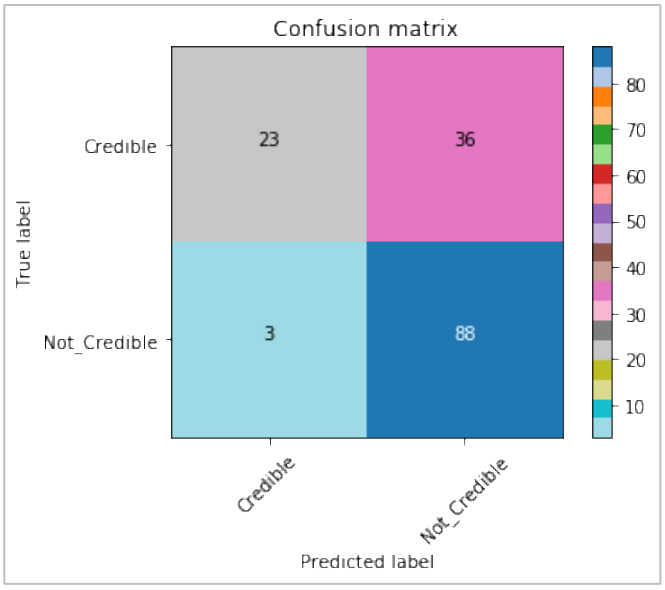

Figure 51. Confusion matrix for Logistic Regression model for classifying manually labeled dataset

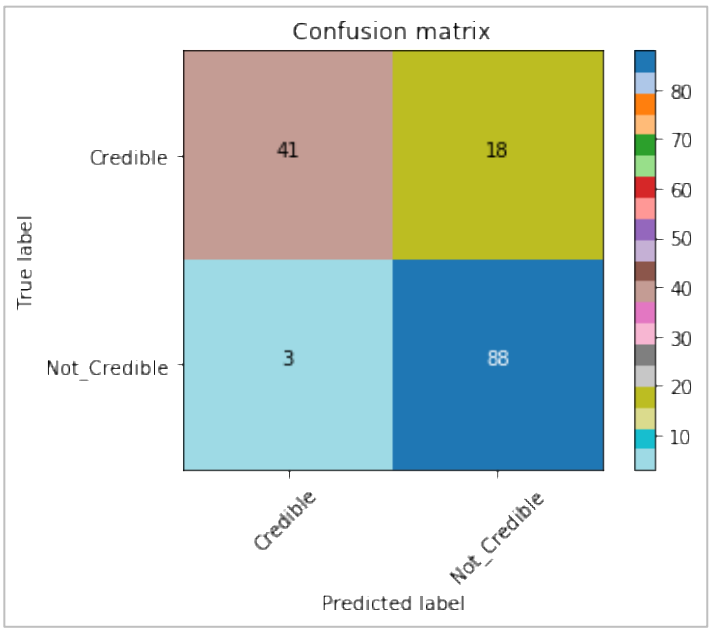

Figure 52. Confusion matrix for SVM model for classifying manually labeled dataset

Next, each model's performance in terms of False Acceptance Rate (FAR), False Rejection Rate (FRR), and Equal Error Rate (EER) are measured and shown in Table 25 and Figure 53. Figure 54 to Figure 58 show the ROC plots for the classifiers.

Table 24 and Table 25 show that decision tree and random forest models have the highest accuracies and the lowest ERR. Also, Figure 54 to Figure 58 show that they have high sensitivity and specificity rates in comparison to other classifiers, Figure 54 to Figure 59. Also, Logistic regression has the lowest accuracy with higher ERR, and a low sensitivity and specificity rates.

Table 25

The FAR, FRR, EER rates for classifying manually labeled dataset

Classifier	Labels	FAR	FRR	EER
SVM	Credible	0.054	0.20	0.10
	Not Credible	0.20	0.054	0.10
KNN	Credible	0.065	0.10	0.078
	Not Credible	0.10	0.065	0.078
Decision Tree	Credible	0.054	0.067	0.066
	Not Credible	0.067	0.054	0.066
Random Forest	Credible	0.032	0.016	0.032
	Not Credible	0.016	0.032	0.032
Logistic Regression	Credible	0.032	0.30	0.076
	Not Credible	0.30	0.032	0.076

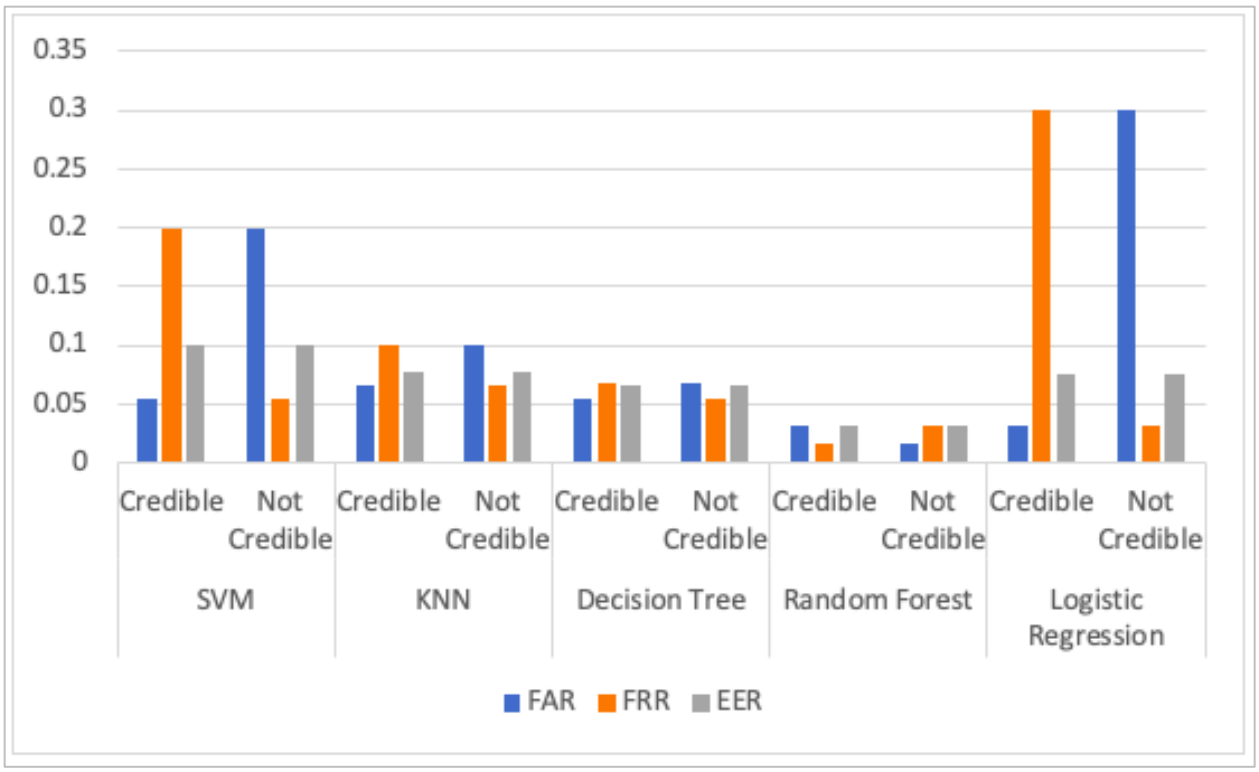

Figure 53. FAR, FRR, EER performance of the credibility classification models for classifying manually labeled dataset.

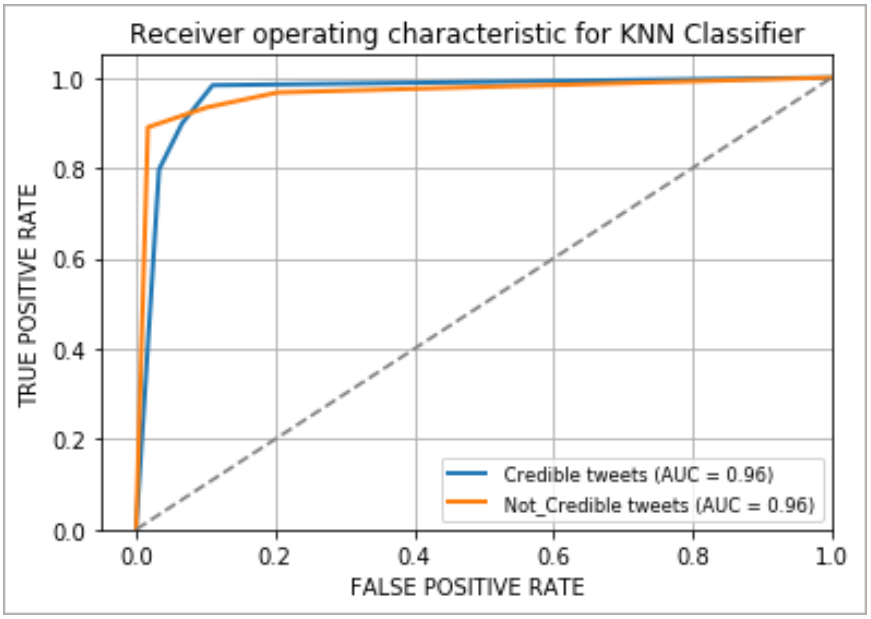

Figure 54. KNN model ROC performance for classifying manually labeled dataset

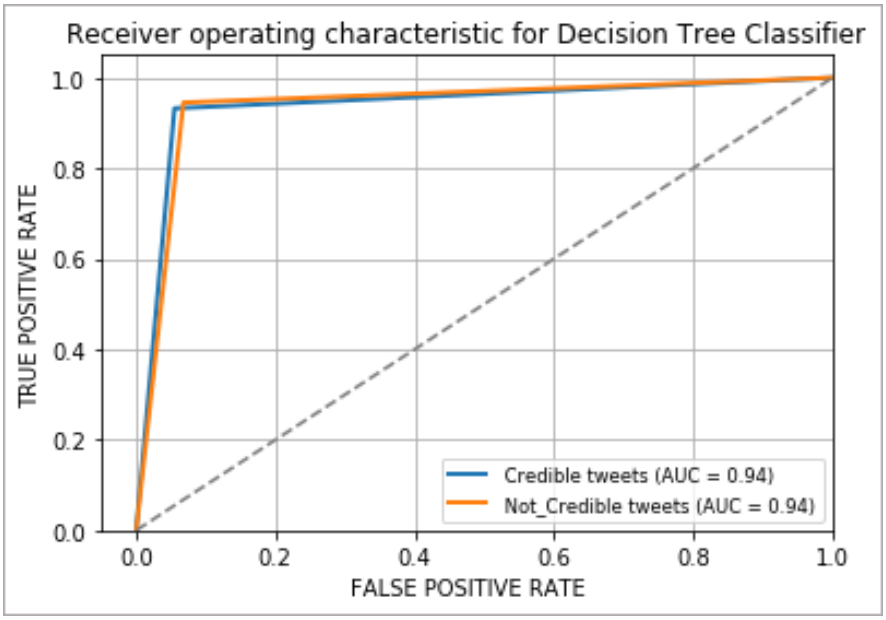

Figure 55. Decision Tree model ROC performance for classifying manually labeled dataset

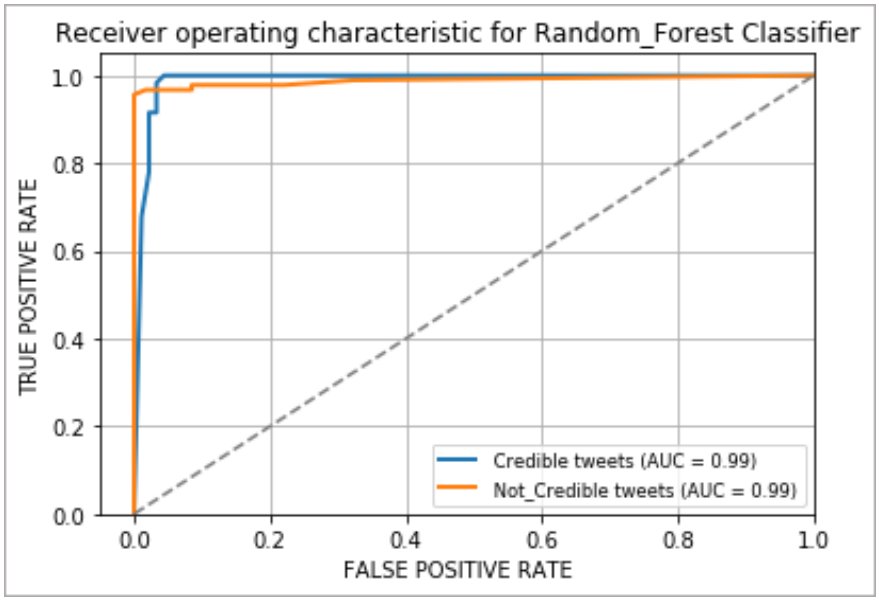

Figure 56. Random Forest model ROC performance for classifying manually labeled dataset

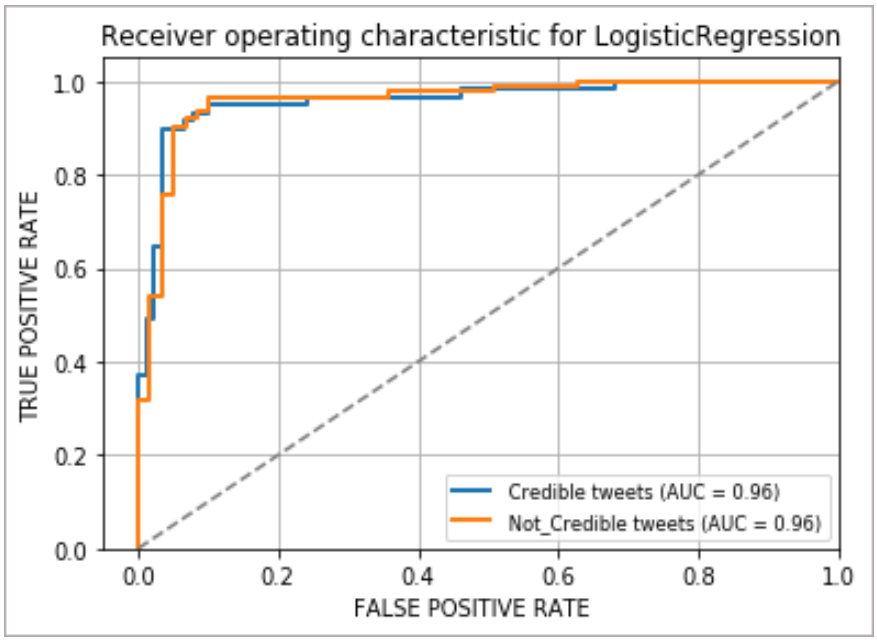

Figure 57. Logistic Regression model ROC performance for classifying manually labeled dataset

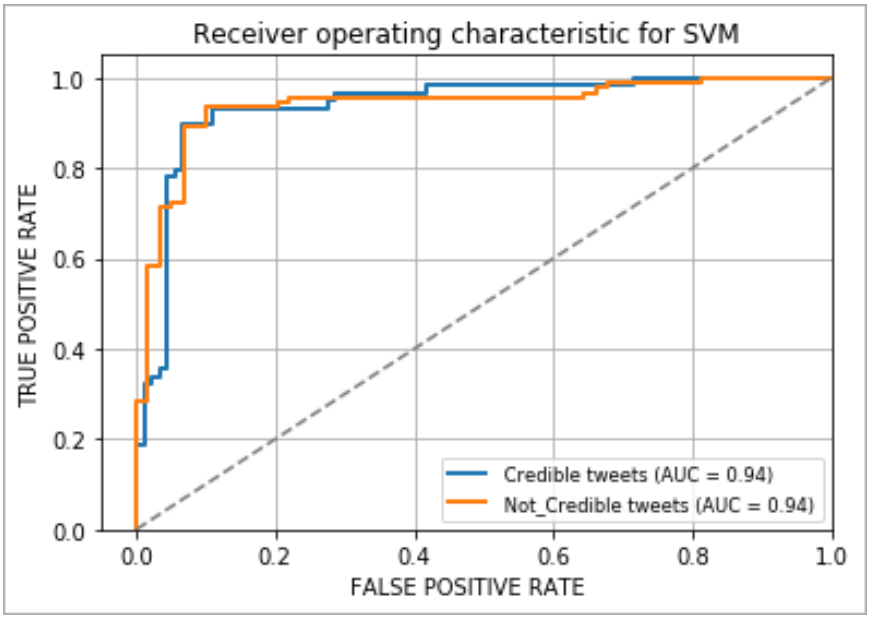

Figure 58. SVM model ROC performance for classifying manually labeled dataset

4.6 Panic Trigger Identification and Classification

In order to identify panic triggers, there was the need to construct a dictionary that contains panic triggers and indicators, which will be looked up when investigating each tweet. The generated dictionary consisted of over 150 panic triggers. This dictionary can be expanded by

adding more triggers for future research. The dictionary only contains triggers regarding hurricane disaster.

Next, after cleaning each tweet, the system searches for trigger within each tweet. Once a match is found, the tweet and all the triggers contained in it are stored. Then the tweet is classified credible or not credible, and a label for response to the triggers is assigned. Figure 59 shows an example of the end results of the panic trigger identification process output.

Raw Tweet	Preprocessed Tweet	Credibility Level	Panic Triggers found	Trigger Response Label
RT @WBRZ: WATCH: Firefighters had to rescue a woman trapped in a flooded car. https://t.co/5bsnw3ziCM	watch firefighters had to rescue a woman trapped in a flooded car	Not Credible	['women trapped']	Mitigation and Correction
RT @Qaugyols,: This Ile-Ife rain dey ,confuse person Sha. Small wind, NEPA go take light. Heavy rain, tornado, hailstorm, whirlwind, NEPA:%#; https://t.co/CUYtj5h34R	This ile ife rain dey confuse person Sha. Small wind, nepa go take light. Heavy rain, tornado, hailstorm, whirlwind, nepa	Not Credible	['heavy rain']	Mitigation and Correction
A Severe Storm Watch has been issued until 10 Mag, main threat i s damaging wind to 70 mph with a lower chance of hail up to 2" in diameter is a very low tornado threat. Watch @ActionlONews First at 4 for the latest track of the storms. https://tco/w2AywbKj4T	a severe storm watch has been issued until 10 pm the main threat is damaging wind to 70 mp h with a lower chance of hail up to 2" in diameter there is a very low tornado threat.	Credible	['severe storm watch', '70 mp', 'storm watch']	Mitigation
"Drone teams showed those in attendance how the technology would be used to help firefighters in search and rescue efforts in the aftermath of a hurricane and more." https://t.co/cPu4dXTmSK	drone teams showed those in attendance how the technology would be used to help firelighters in search and rescue efforts in the aftermath of a hurricane and more	Not Credible		No Triggers Contained

Figure 59. An example of the end result of Panic Trigger Identification.

Figure 60 and figure 61, show the overall percentage of tweets with different trigger response labels on hurricane Florence and hurricane Michael datasets. machine learning classification is conducted using the following features:

```
Classification_features = dataframe [ 'processed_tweet', 'Credibility',
'panic_triggers_found', 'trigger_response_label]
```

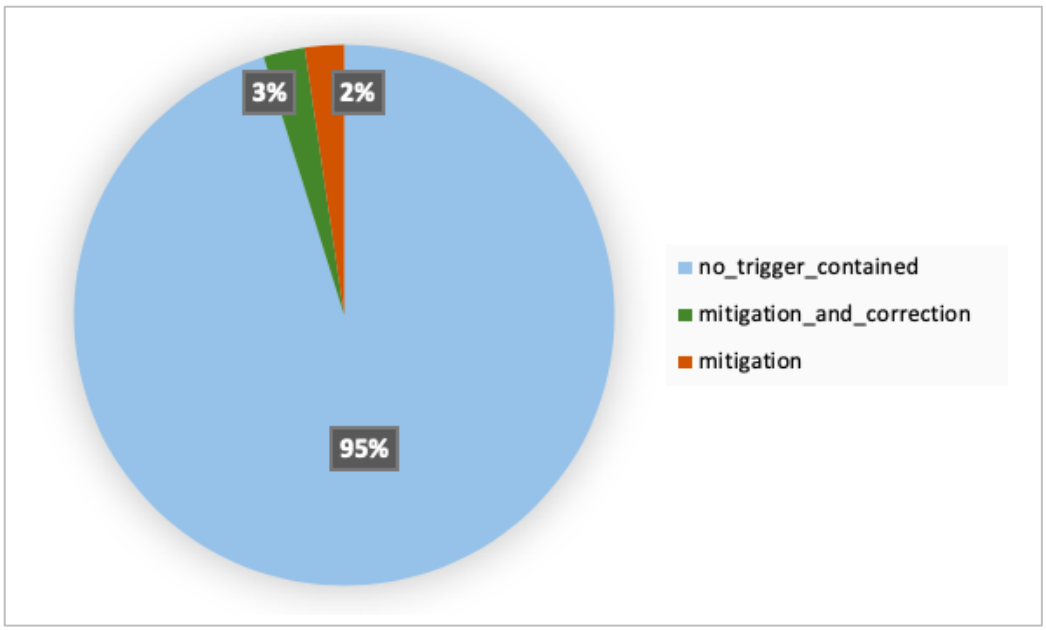

Figure 60. Percentage of tweets with different panic trigger response labels on hurricane Florence

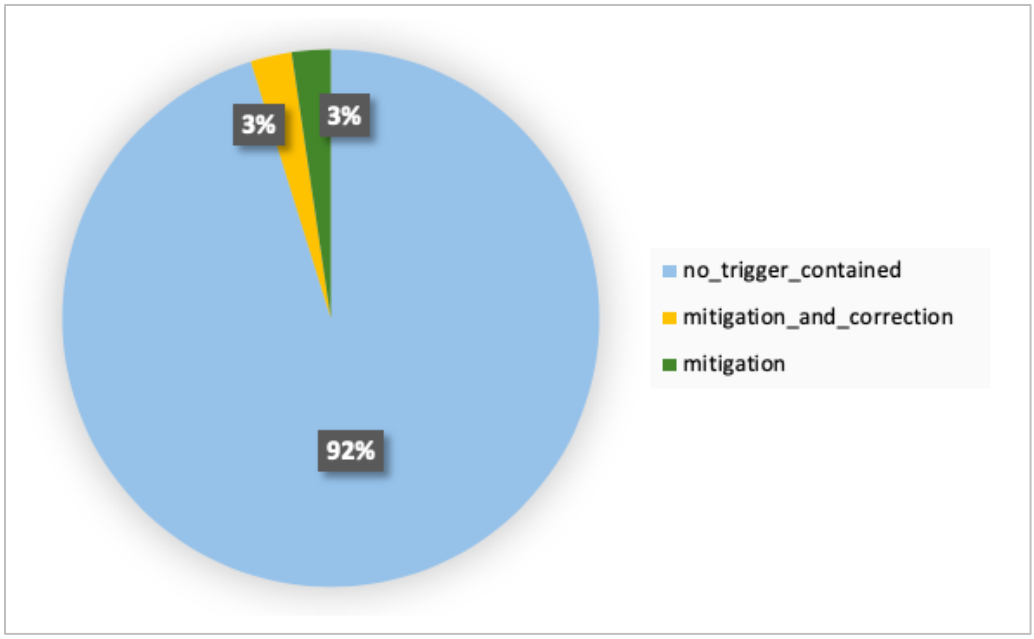

Figure 61. Percentage of tweets with different panic trigger response labels on hurricane Michael

4.6.1 Classifying tweets using TfidfVectorizer

4.6.1.1 Hurricane Florence dataset

After applying the word vectorizers to the dataset and acquiring the TfidfVectorizer features, these features were fed into machine learning algorithms to classify the tweets into three classes: “Mitigation”, “Mitigation_and_Correction”, and “No_triggers_Contiained”. Then the performance of the algorithms was compared. The accuracy, precision, recall, f score and ROC, FAR, FRR, and EER of the test results were calculated for each algorithm. A good classifier identifies a large amount of data in a short amount of time with high precision and recall scores and low EER.

The dataset was split into 70% training set and 30% test set. The training set contains the known output and the classification algorithms learn on this data in order to classify test data. Table 26 shows the precision, recall, and f-score and the accuracy values for all the models using TfidVectorizer vectorization features for hurricane Florence dataset. It can be seen that all the models showed high precision, recall, f-score and the accuracy values for the tweets in the “No_Triggers_Contained” class. However, the precision, recall, and f-score values for the “Mitigation” and “Mitigation_and_Correction” classes are lower. This was because there was a much higher number of tweets that do not contain panic triggers in the training set. Figure 62 to Figure 65, show the confusion matrix for each classification model.

Table 26

Classification performance metrics for hurricane Florence dataset using TfidfVectorizer features

Classifier	Labels	Precision	Recall	F-score	Accuracy
KNN	Mitigation and correction	0.71	0.50	0.49	0.96
	mitigation	0.72	0.45	0.52	
	No trigger contained	0.98	1.00	0.99	
Decision Tree	Mitigation and correction	0.62	0.67	0.64	0.98
	mitigation	0.65	0.62	0.62	
	No trigger contained	1.00	1.00	1.00	
Random Forest	Mitigation and correction	0.66	0.62	0.63	0.97
	mitigation	0.73	0.50	0.60	
	No trigger contained	0.99	1.00	0.99	
Logistic Regression	Mitigation and correction	0.73	0.45	0.40	0.96
	mitigation	0.73	0.45	0.49	
	No trigger contained	0.97	1.00	0.98	

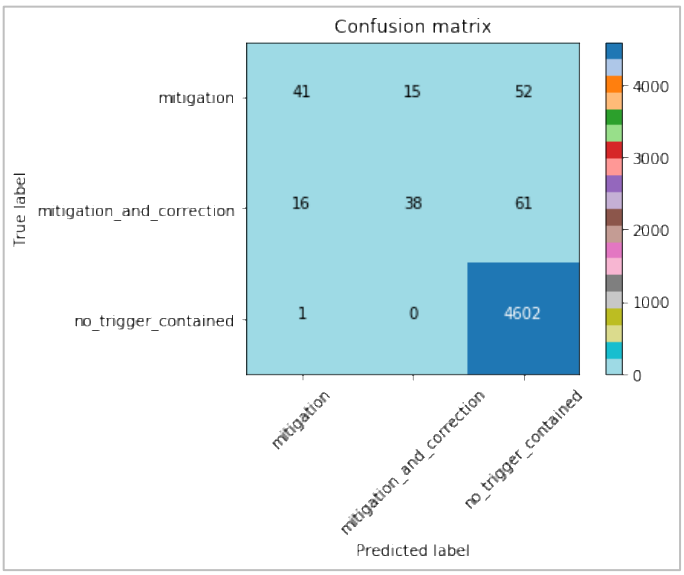

Figure 62. Confusion matrix for KNN model for predicting panic trigger labels for hurricane Florence dataset using TfidfVectorizer

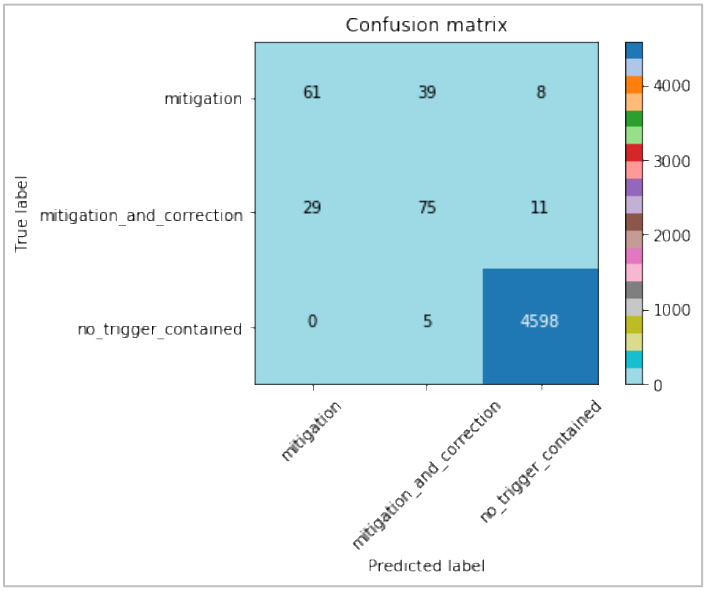

Figure 63. Confusion matrix for Decision Tree model for predicting panic trigger labels for hurricane Florence dataset using TfidfVectorizer

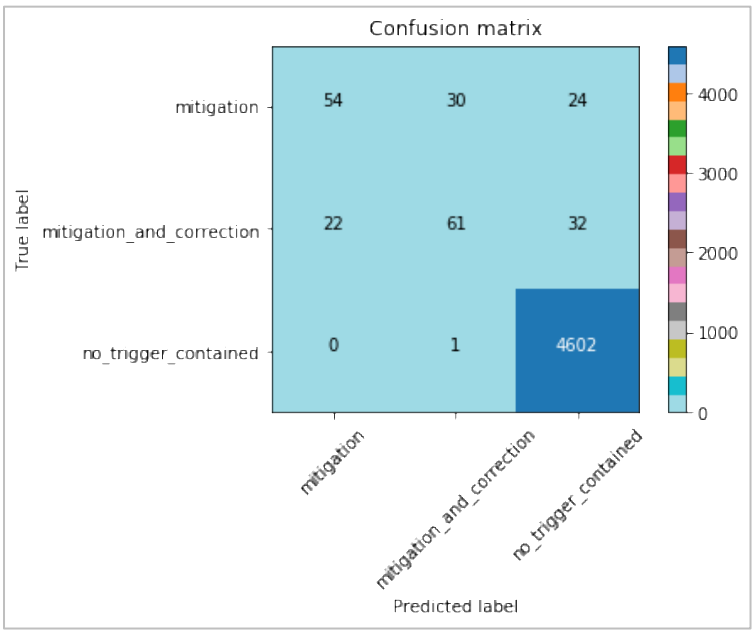

Figure 64. Confusion matrix for Random Forest model for predicting panic trigger labels for hurricane Florence dataset using TfidfVectorizer

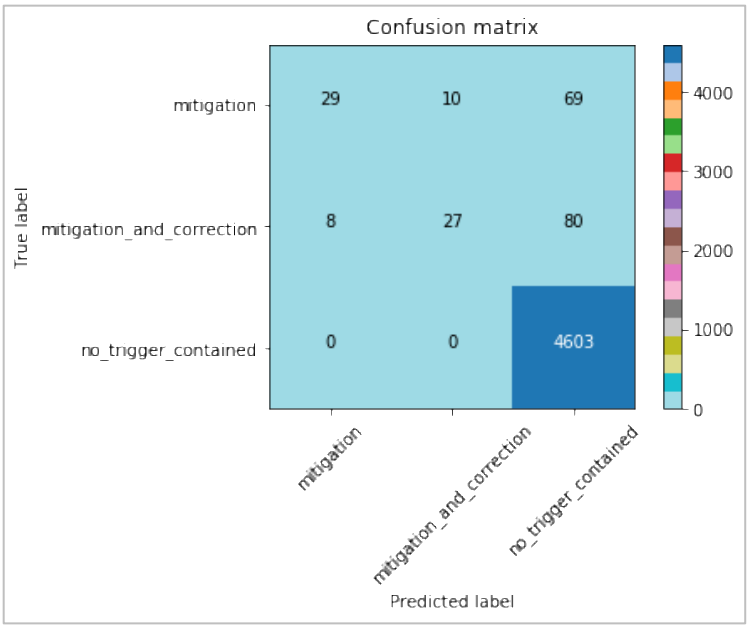

Figure 65. Confusion matrix for Logistic Regression model for predicting panic trigger labels on hurricane Florence dataset using TfidfVectorizer

Next, each model's performance in terms of False Acceptance Rate (FAR), False Rejection Rate (FRR), and Equal Error Rate (EER)is measured and shown in Table 27 and Figure 66. Figure 68 to Figure 70 show the ROC plots for each classifier.

We can summarize from Table 26, Table 27, and Figure 66 that decision tree and random forest models show the highest accuracies and the lowest ERR. Figure 68 to Figure 70 show that they had high sensitivity and specificity rates in comparison to other classifiers. Logistic regression shows the lowest accuracy with higher ERR, and a low sensitivity and specificity rates. The performance of KNN is the least with higher EER rates.

Table 27

The FAR, FRR, and ERR rates for predicting panic trigger labels for Hurricane Florence dataset using TfidfVectorizer

Classifier	Labels	FAR	FRR	ERR
KNN	Mitigation and correction	0.003	0.62	0.28
	mitigation	0.50	0.0	0.2
	No trigger contained	0.003	0.66	0.33
Decision Tree	Mitigation and correction	0.0	0.43	0.26
	mitigation	0.08	0.001	0.07
	No trigger contained	0.0	0.34	0.17
Random Forest	Mitigation and correction	0.0	0.5	0.16

	mitigation	0.25	0.46	0.09
	No trigger contained	0.0	0.49	0.09
Logistic Regression	Mitigation and correction	0.0	0.43	0.06
	mitigation	0.40	0.0	0.04
	No trigger contained	0.0	0.41	0.07

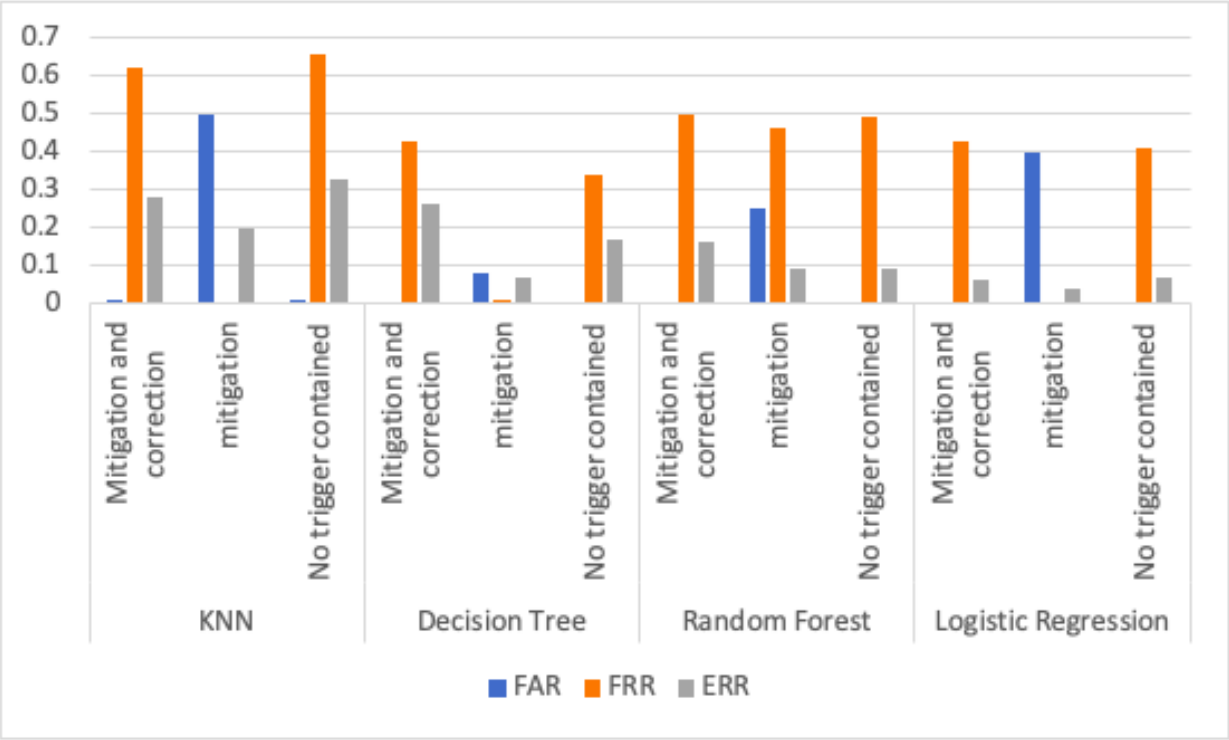

Figure 66. The FAR, FRR, and ERR rates for predicting panic trigger labels for Hurricane Florence dataset using TfidfVectorizer

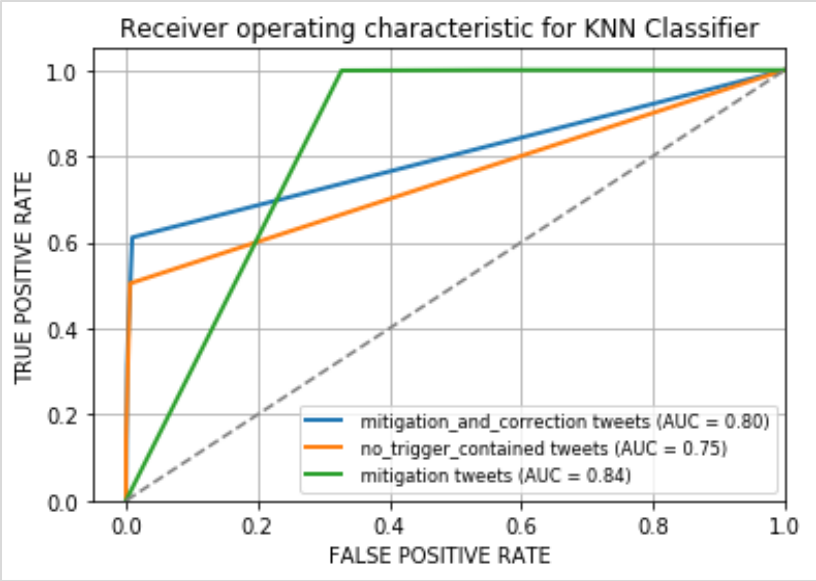

Figure 67. ROC plot for KNN model for predicting panic trigger labels for hurricane Florence dataset using TfidfVectorizer

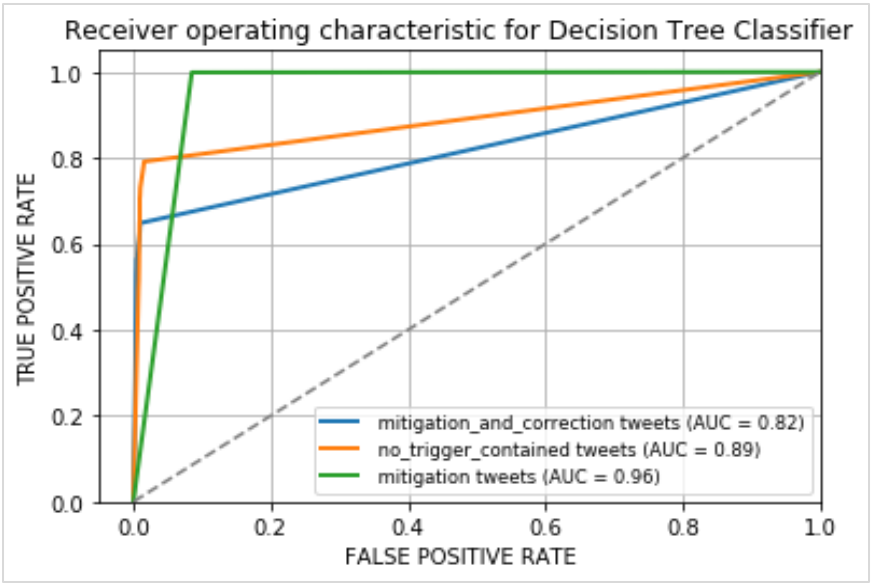

Figure 68. ROC plot for Decision Tree model for predicting panic trigger labels for hurricane Florence dataset using TfidfVectorizer

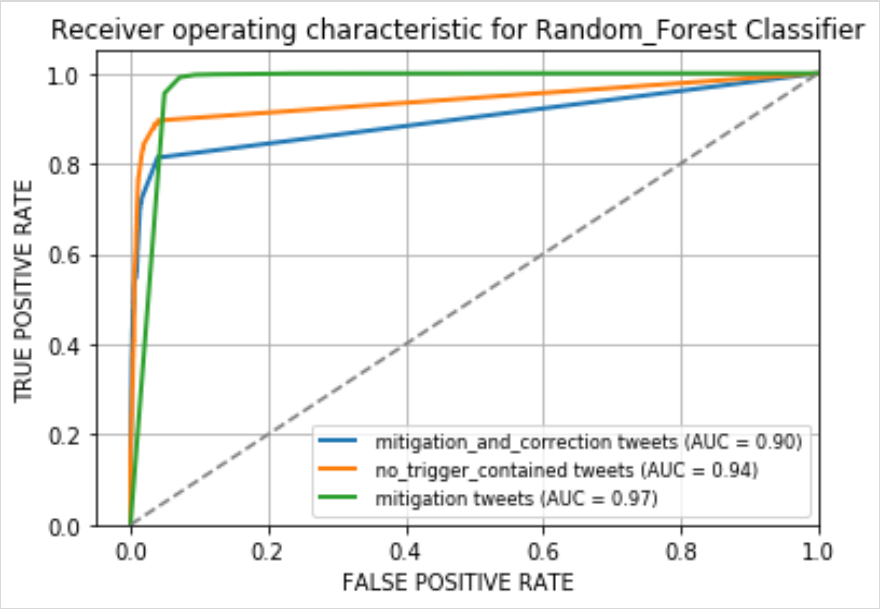

Figure 69. ROC plot for Random Forest model for predicting panic trigger labels for hurricane Florence dataset using TfidfVectorizer

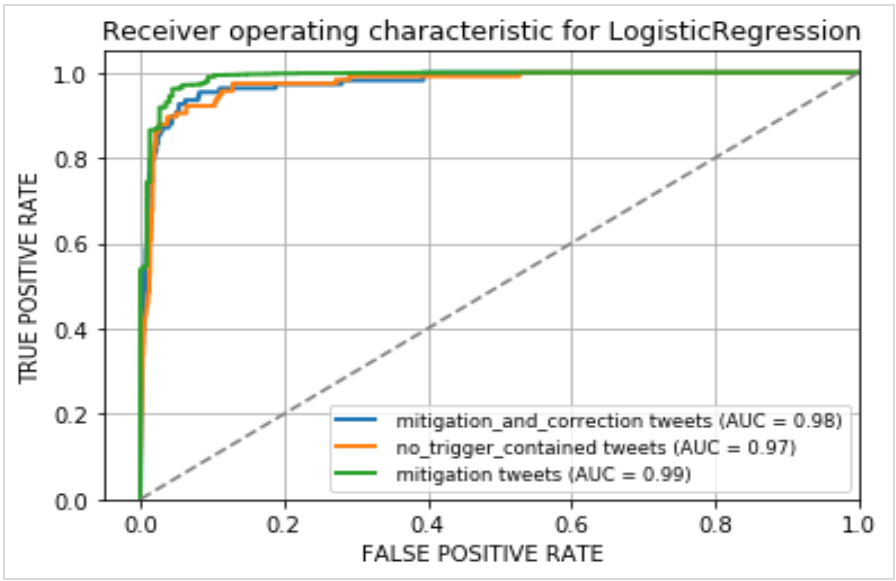

Figure 70. ROC plot for Logistic Regression model for predicting panic trigger labels for hurricane Florence dataset using TfidfVectorizer

4.6.1.2 Hurricane Michael dataset

The same experiment was conducted on hurricane Michael dataset. Table 28 shows the precision, recall, and f-score, and accuracy values for all the models using TfidfVectorizer vectorization features for hurricane Michael dataset. It can be seen that all the models showed high precision, recall and f-score values for the tweets in the "No_Triggers_contained" class. However, the precision recall and f-score values for the "Mitigation" and "Mitigation_and_Correction" classes are lower. This was because there was a much higher number of tweets that do not contain panic triggers in the training set. Figure 71 to Figure 74, show the confusion matrix for each classification model.

Table 28

Classification performance metrics for hurricane Michael dataset using TfidfVectorizer features

Classifier	Labels	Precision	Recall	F-score	Accuracy
KNN	Mitigation and correction	0.57	0.45	0.45	0.94
	mitigation	0.64	0.50	0.55	
	No trigger contained	0.97	1.00	0.98	
Decision Tree	Mitigation and correction	0.71	0.62	0.65	0.96
	mitigation	0.64	0.77	0.72	
	No trigger contained	1.00	1.00	1.00	
Random Forest	Mitigation and correction	0.73	0.60	0.63	0.96

	mitigation	0.72	0.75	0.74	
	No trigger contained	0.99	1.00	1.00	
Logistic Regression	Mitigation and correction	0.64	0.35	0.55	0.93
	mitigation	0.65	0.50	0.40	
	No trigger contained	0.95	1.00	0.98	

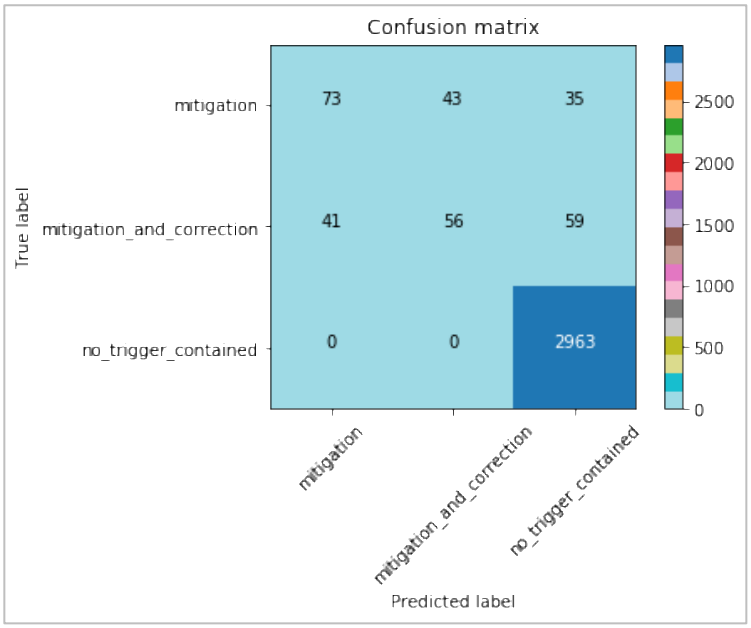

Figure 71. Confusion matrix for KNN model for predicting panic trigger labels for hurricane Michael dataset using TfidfVectorizer

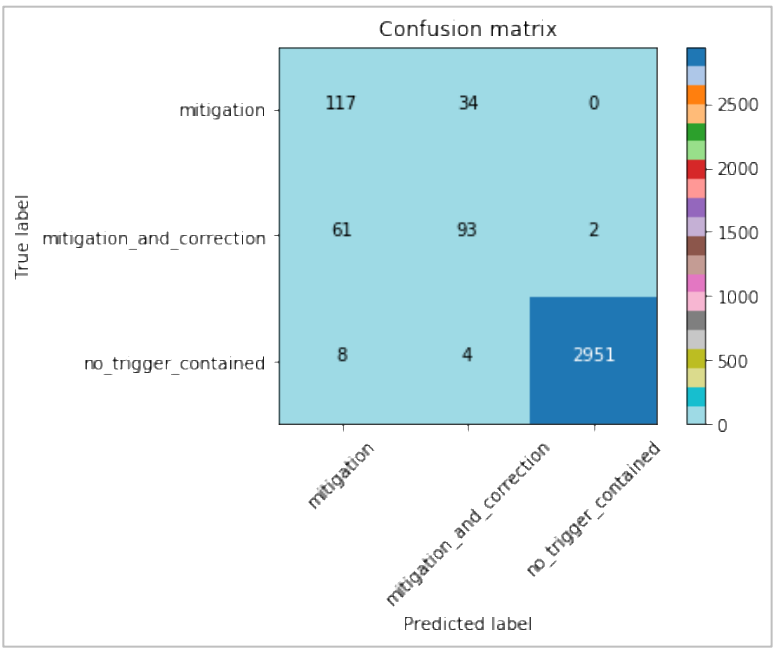

Figure 72. Confusion matrix for Decision Tree model for predicting panic trigger labels for hurricane Michael dataset using TfidfVectorizer

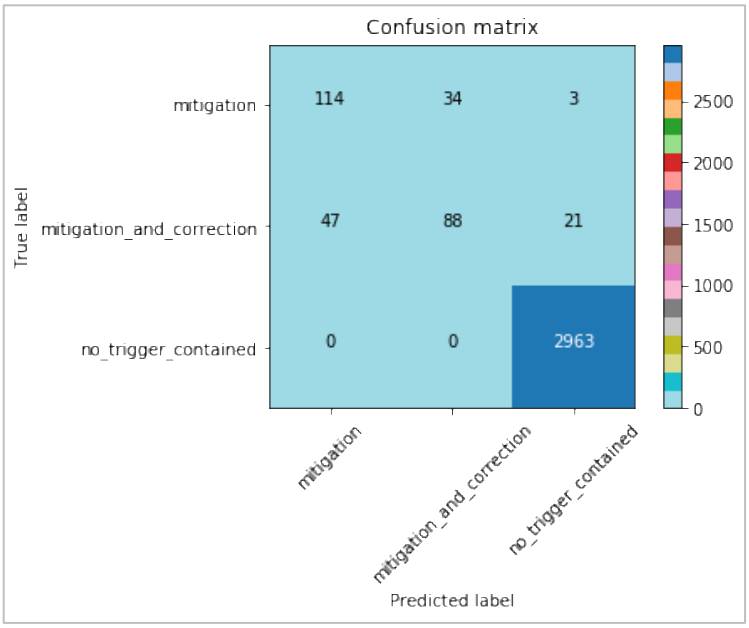

Figure 73. Confusion matrix for Random Forests model for predicting panic trigger labels for hurricane Michael dataset using TfidfVectorizer

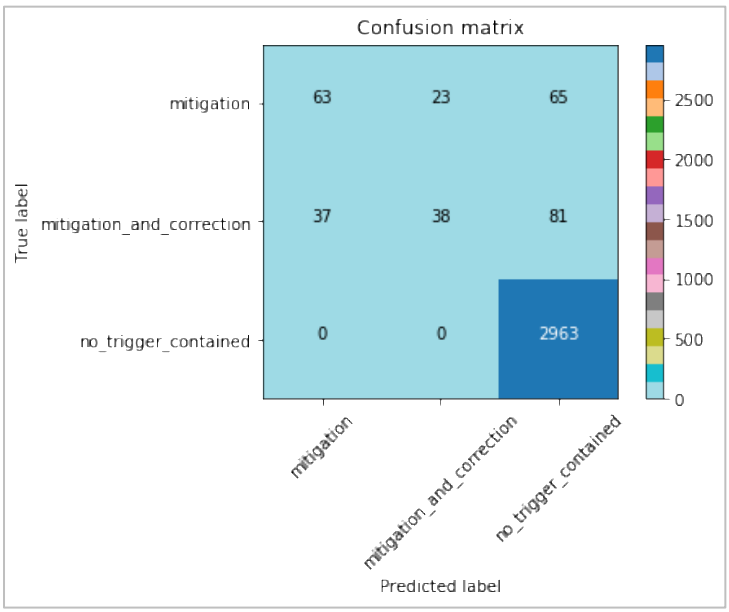

Figure 74. Confusion matrix for Logistic Regression model for predicting panic trigger labels for hurricane Michael dataset using TfidfVectorizer

Next, each model's performance in terms of False Acceptance Rate (FAR), False Rejection Rate (FRR), and Equal Error Rate (EER) is measured and shown in Table 29 and Figure 75. Table 28, Table 29, and Figure 75 show that decision tree and random forest models show the highest accuracies and the lowest ERR. Figure 76 to Figure 79 show that they had high sensitivity and specificity rates in comparison to other classifiers, and Logistic regression had the lowest accuracy with higher ERR, and a low sensitivity and specificity rates. The KNN performance was the least with higher EER rates.

Table 29

The FAR, FRR, and ERR rates for predicting panic trigger labels for Hurricane Michael dataset using TfidfVectorizer

Classifier	Labels	FAR	FRR	ERR

KNN	Mitigation and correction	0.015	0.40	0.05
	mitigation	0.21	0.0	0.05
	No trigger contained	0.01	0.5	0.16
Decision Tree	Mitigation and correction	0.017	0.19	0.02
	mitigation	0.0	0.001	0.001
	No trigger contained	0.01	0.3	0.05
Random Forest	Mitigation and correction	0.01	0.25	0.03
	mitigation	0.04	0.001	0.008
	No trigger contained	0.01	0.40	0.04
Logistic Regression	Mitigation and correction	0.01	0.25	0.03
	mitigation	0.05	0.0	0.02
	No trigger contained	0.01	0.4	0.05

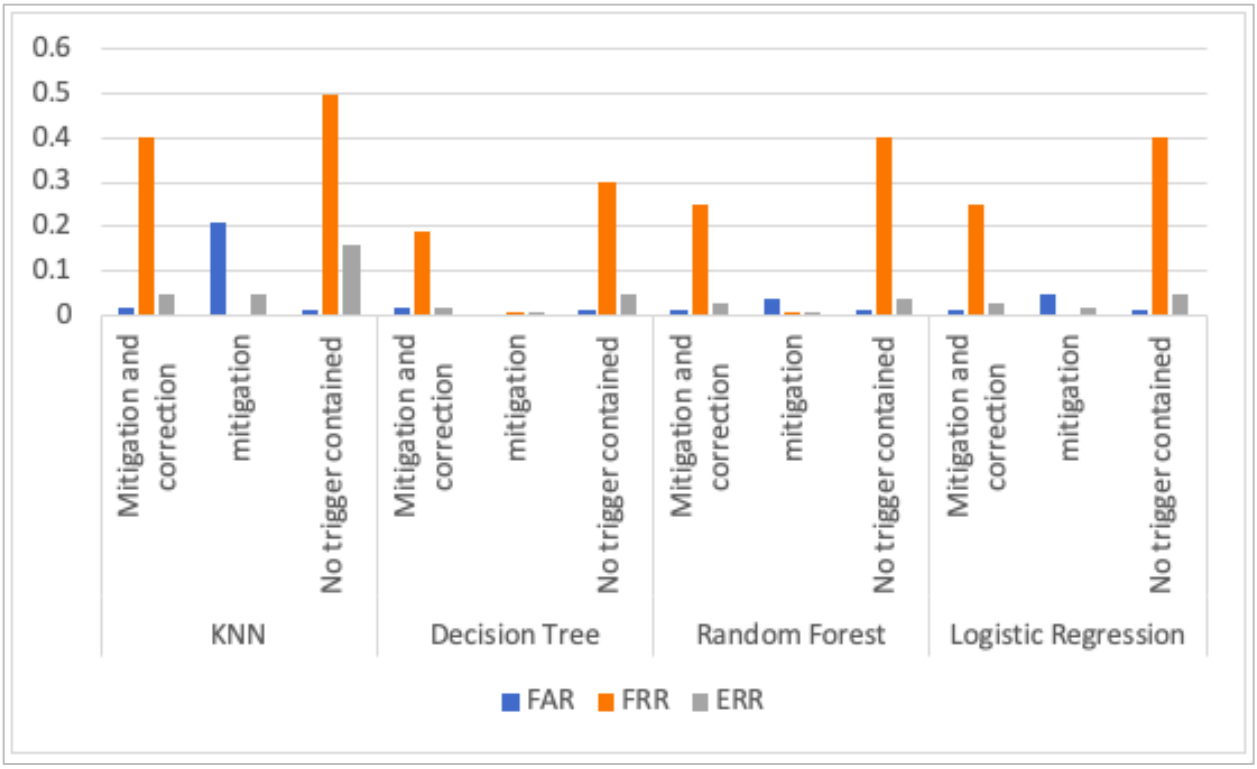

Figure 75. The FAR, FRR, and ERR rates for predicting panic trigger labels for Hurricane Michael dataset using TfidfVectorizer

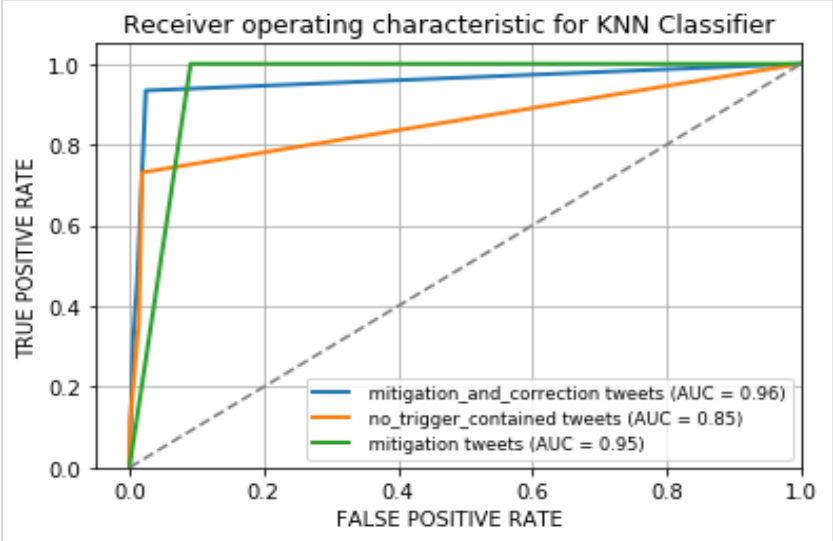

Figure 76. ROC plot for KNN model for predicting panic trigger labels for hurricane Michael dataset using TfidfVectorizer

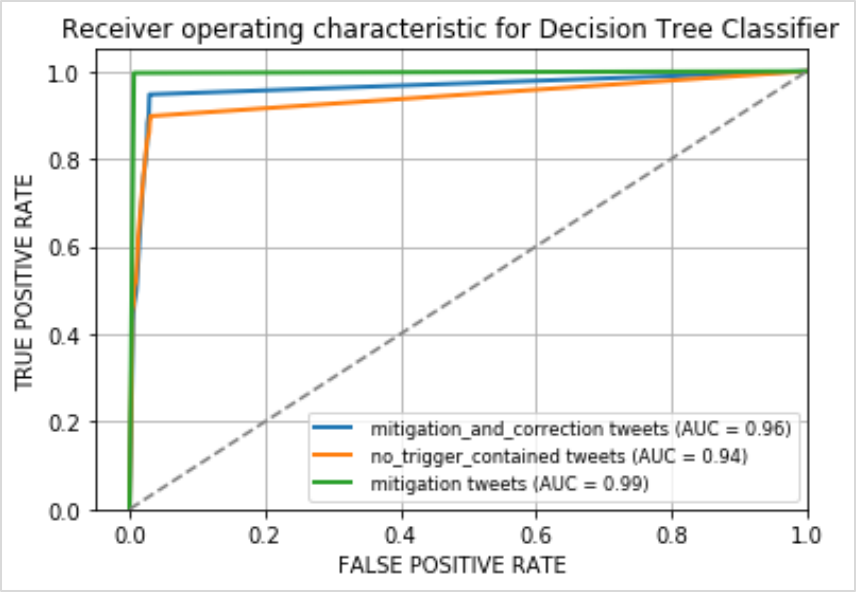

Figure 77. ROC plot for Decision Tree model for predicting panic trigger labels for hurricane Michael dataset using TfidfVectorizer

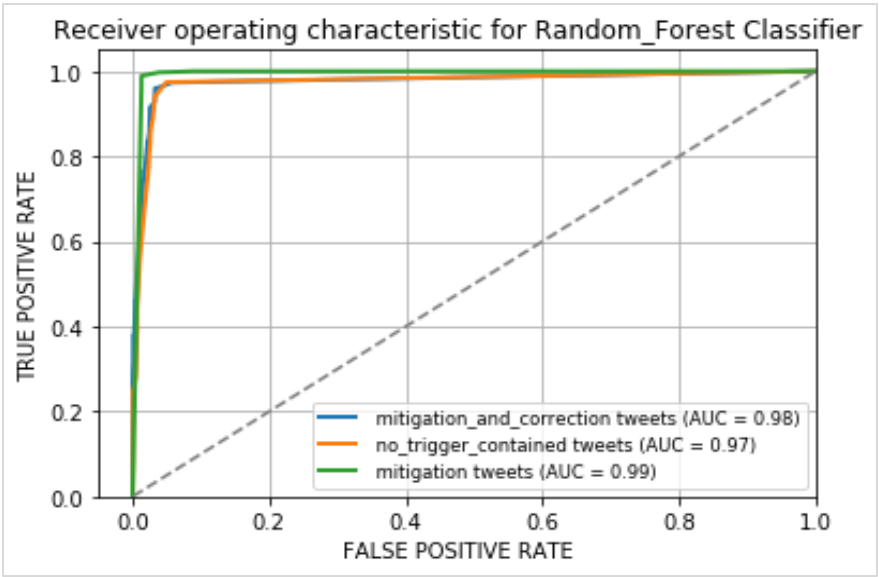

Figure 78. ROC plot for Random Forest model for predicting panic trigger labels for hurricane Michael dataset using TfidfVectorizer

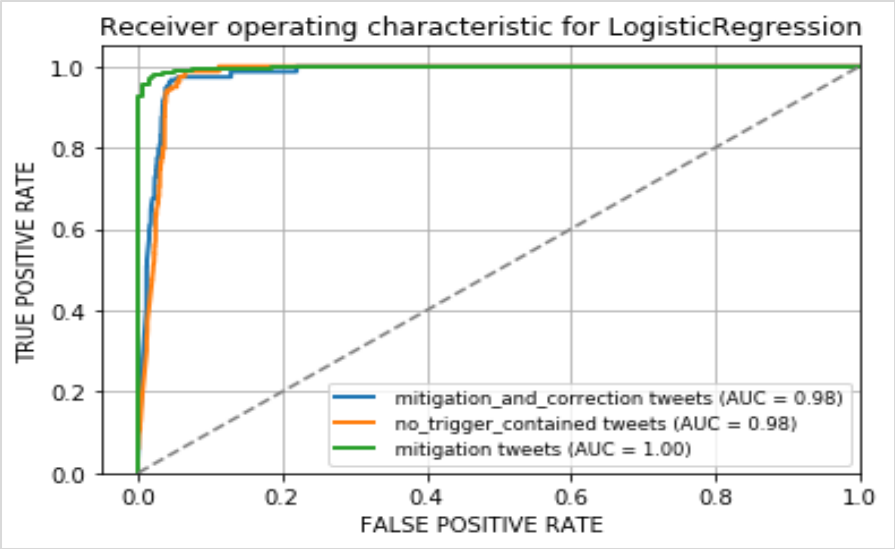

Figure 79. ROC plot for Logistic Regression model for predicting panic trigger labels for hurricane Michael dataset using TfidfVectorizer

4.6.2 Classifying panic trigger tweets using CountVectorizer

4.6.2.1 Hurricane Florence dataset

After applying the word vectorizers to the dataset and acquiring the CountVectorizer features, these features were fed into machine learning algorithms for classification, then a comparison of the performance of the algorithms was conducted. The accuracy, precision, recall, f score ROC, FAR, FRR, and EER of the test results were calculated for each algorithm. A good classifier identifies a large amount of data in a short amount of time with high precision and recall scores and low EER.

The dataset was split into 70% training set and 30% test set. The training set contains the known output and the classification algorithms learn on this data in order to classify test data. Table 30 shows the precision, recall, f-score, and accuracy values for all the models using CountVectorizer features for hurricane Florence dataset. It can be seen that all the models showed high precision, recall and f-score values for the tweets in the “No_Triggers_Contained” class.

However, the precision, recall and f-score values for the "Mitigation" and "Mitigation_and_Correction" classes are lower. This was because there was a much higher number of tweets that do not contain panic triggers in the training set. Figure 80 to Figure 83, show the confusion matrix for each classification model.

Table 30

Classification performance metrics for hurricane Florence dataset using CountVectorizer features

Classifier	Labels	Precision	Recall	F-score	Accuracy
KNN	Mitigation and correction	0.69	0.43	0.53	0.97
	Mitigation	0.67	0.40	0.50	
	No trigger contained	0.98	1.00	0.90	
Decision Tree	Mitigation and correction	0.64	0.65	0.65	0.98
	Mitigation	0.68	0.60	0.64	
	No trigger contained	1.00	1.00	1.00	
Random Forest	Mitigation and correction	0.73	0.59	0.64	0.97
	Mitigation	0.75	0.59	0.63	
	No trigger contained	0.99	1.00	0.99	
	Mitigation and correction	0.73	0.60	0.66	0.97

Logistic Regression	Mitigation	0.73	0.55	0.63	
	No trigger contained	0.99	1.00	0.99	

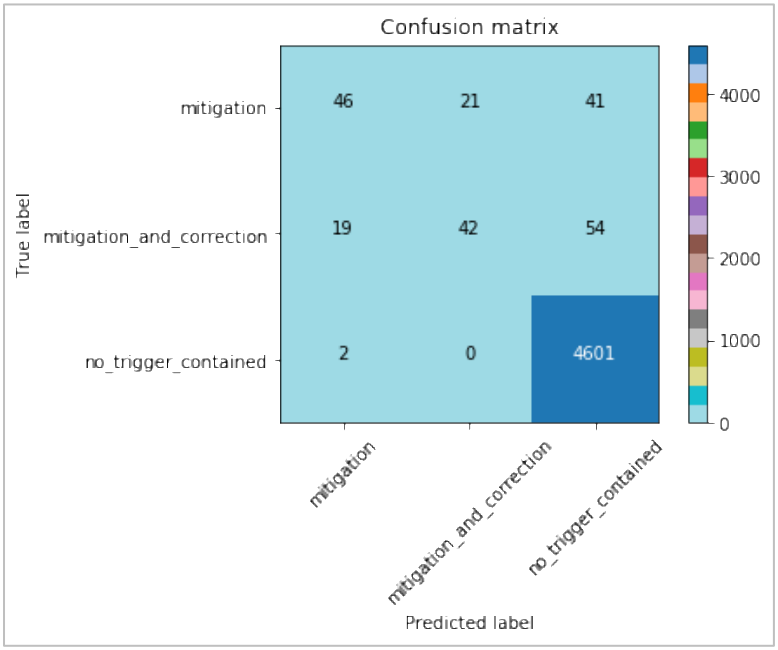

Figure 80. Confusion matrix for KNN model for predicting panic trigger labels on hurricane Florence dataset using CountVectorizer

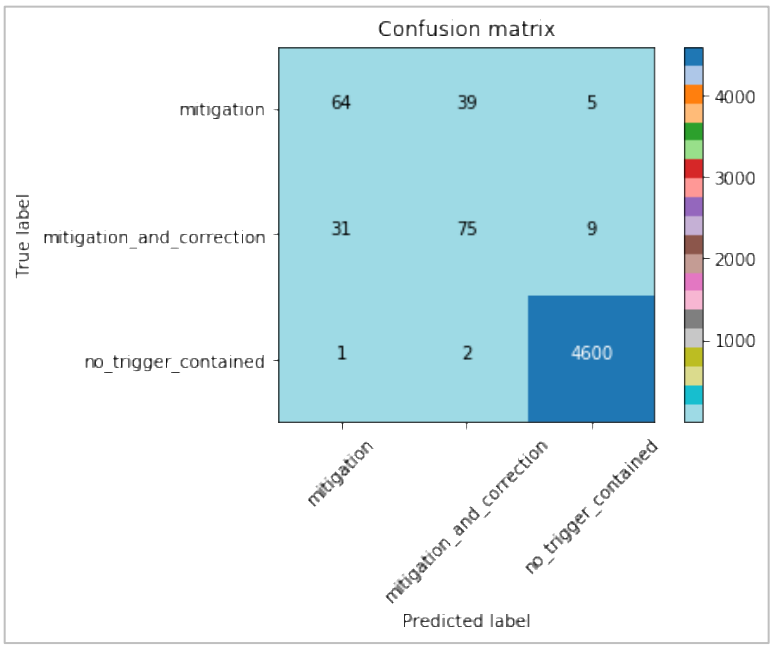

Figure 81. Confusion matrix for Decision Tree model for predicting panic trigger labels on hurricane Florence dataset using CountVectorizer

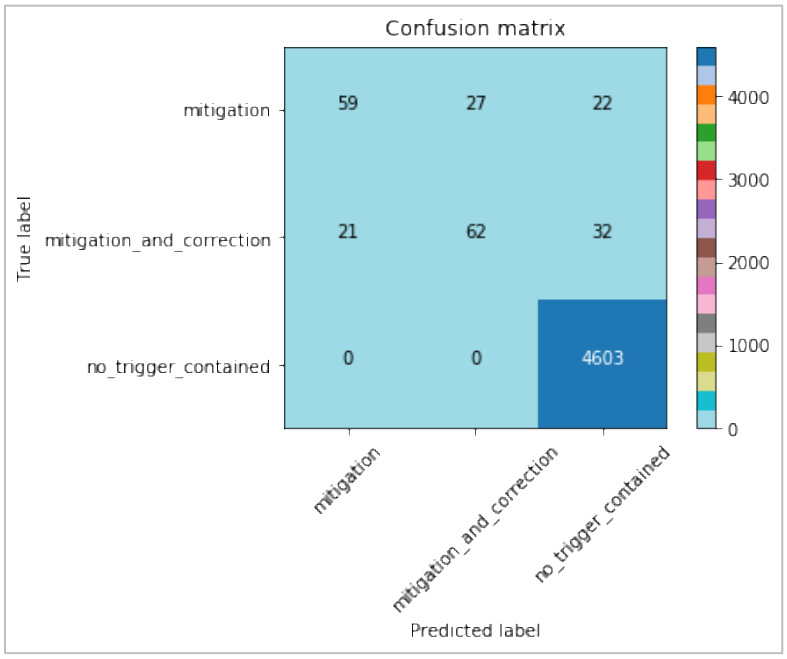

Figure 82. Confusion matrix for Random Forest model for predicting panic trigger labels on hurricane Florence dataset using CountVectorizer

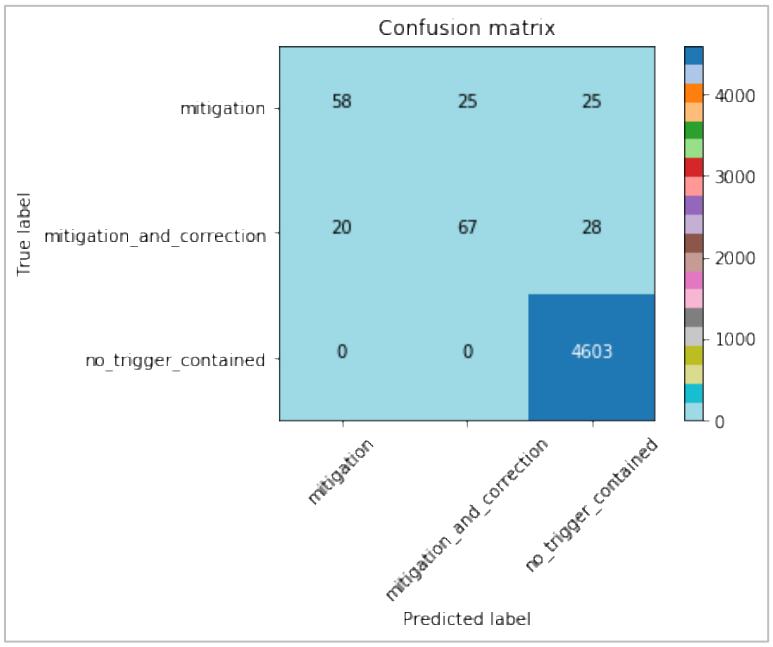

Figure 83. Confusion matrix for Logistic Regression model for predicting panic trigger labels on hurricane Florence dataset using CountVectorizer

Next, each model's performance in terms of False Acceptance Rate (FAR), False Rejection Rate (FRR), and Equal Error Rate (EER) is measured and shown in Table 30 and Figure 84. Figure 85 to Figure 88, show the ROC plots for each classifier.

Table 30, Table 31, and Figure 84 show that decision tree and random forest models show the highest accuracies and the lowest ERR. Also, Figure 85 to Figure 88 show that they had high sensitivity and specificity rates in comparison to other classifiers, Figure 86 to Figure 89. Logistic regression shows the lowest accuracy, and higher ERR with a low sensitivity and specificity rates on datasets. The performance of KNN is the least with higher EER.

Table 31

The FAR, FRR, and ERR rates for predicting panic trigger labels for Hurricane Florence dataset using CountVectorizer

Classifier	Labels	FAR	FRR	ERR
KNN	Mitigation and correction	0.004	0.57	0.26
	mitigation	0.42	0.0	0.29
	No trigger contained	0.004	0.63	0.29
Decision Tree	Mitigation and correction	0.006	0.40	0.23
	mitigation	0.06	0.0	0.05
	No trigger contained	0.008	0.34	0.16
Random Forest	Mitigation and correction	0.004	0.45	0.08
	mitigation	0.24	0.0	0.03
	No trigger contained	0.005	0.46	0.10
Logistic Regression	Mitigation and correction	0.08	0.46	0.08
	mitigation	0.23	0.0	0.04
	No trigger contained	0.005	0.41	0.10

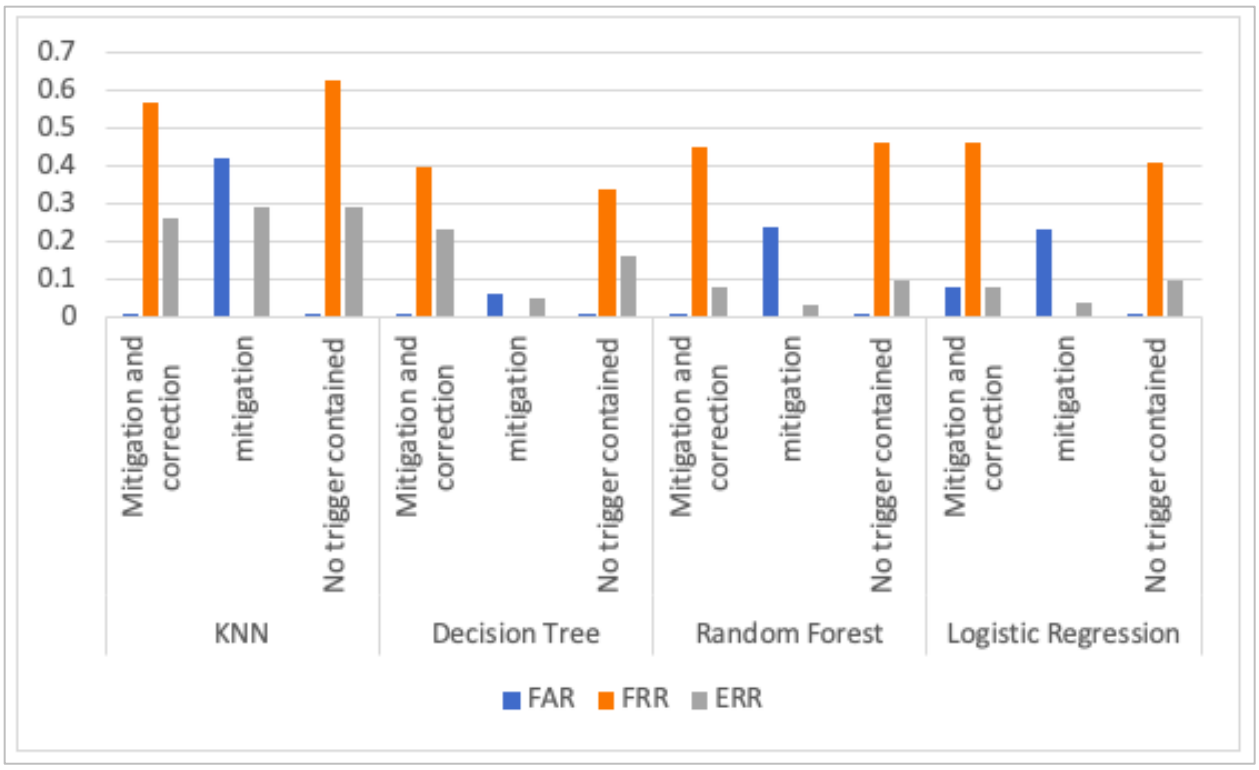

Figure 84. The FAR, FRR, and ERR rates for predicting panic trigger labels for Hurricane Florence dataset using CountVectorizer

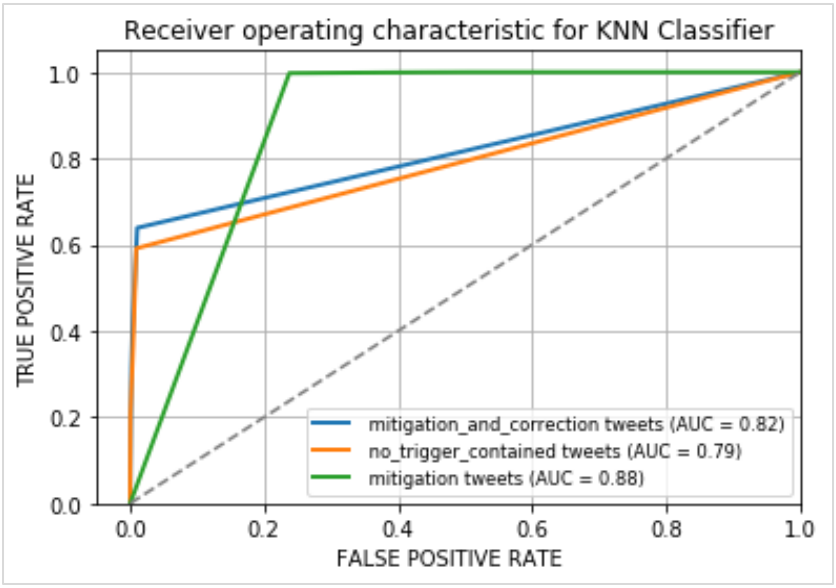

Figure 85. ROC plot for KNN model for predicting panic trigger labels for hurricane Florence dataset using CountVectorizer

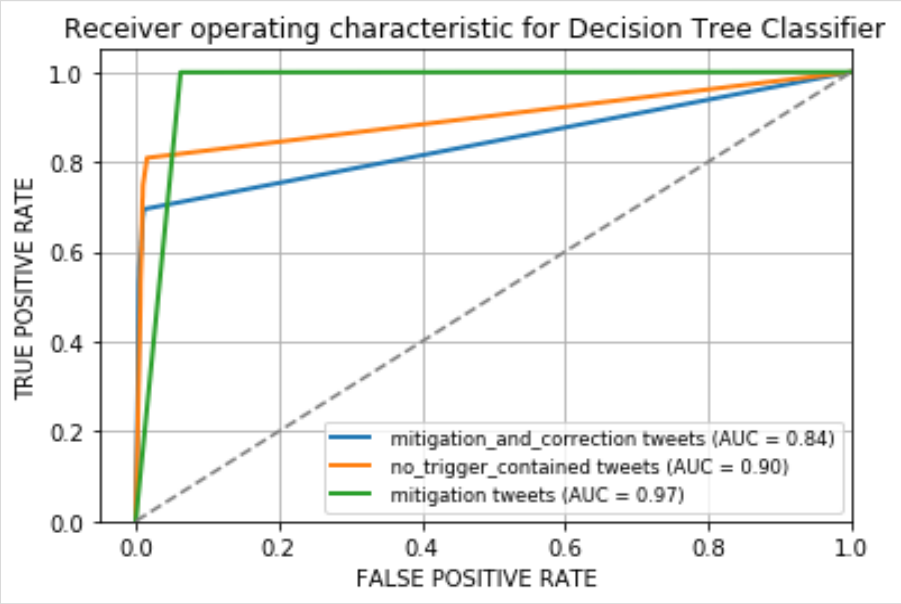

Figure 86. ROC plot for Decision Tree model for predicting panic trigger labels for hurricane Florence dataset using CountVectorizer

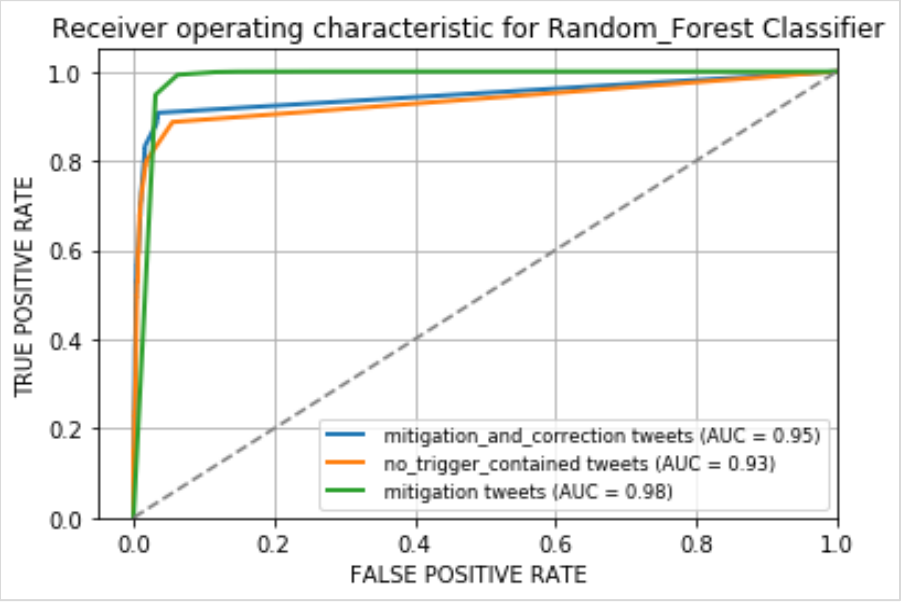

Figure 87. ROC plot for Random Forest model for predicting panic trigger labels for hurricane Florence dataset using CountVectorizer

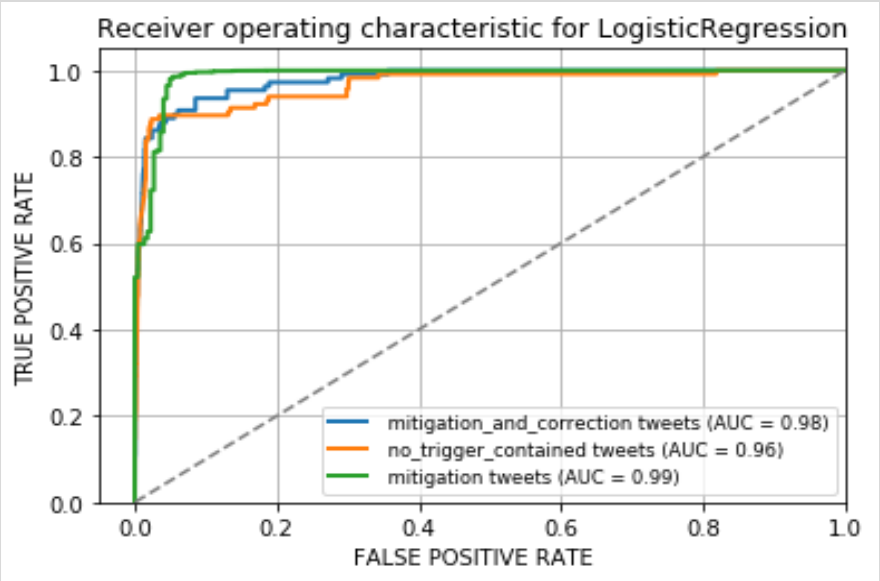

Figure 88. ROC plot for Logistic Regression model for predicting panic trigger labels for hurricane Florence dataset using CountVectorizer

4.6.2.2 Hurricane Michael dataset

After applying the word vectorizers to the dataset and acquiring the CountVectorizer features, these features were fed into machine learning algorithms for, then a comparison of the performance of the algorithms was conducted. The accuracy, precision, recall, f score and ROC, FAR, FRR, and EER of the test results were calculated. A good classifier identifies a large amount of data in a short amount of time with high precision and recall scores and low EER.

The dataset was split into 70% training set and 30% test set. The training set contains the known output and the classification algorithms learn on this data in order to classify test data. Table 32 shows the precision, recall, f score and the overall classification accuracies of the algorithms used in the experiment using CountVectorizer features. It can be seen that all the models showed high precision, recall and f-score values for the tweets in the "No_Triggers_Contained" class. However, the precision recall and f-score values for the "Mitigation" and "Mitigation_and_Correction" classes are lower. This was because there was a

much higher number of tweets that do not contain panic triggers in the training set. Figure 89 to Figure 92 show the confusion matrix for each classification model.

Table 32

Classification performance for hurricane Michael dataset using CountVectorizer features

Classifier	Labels	Precision	Recall	F-score	Accuracy
KNN	Mitigation and correction	0.60	0.59	0.59	0.95
	Mitigation	0.63	0.50	0.53	
	No trigger contained	0.98	0.98	0.98	
Decision Tree	Mitigation and correction	0.70	0.66	0.70	0.97
	Mitigation	0.70	0.80	0.71	
	No trigger contained	1.00	1.00	1.00	
Random Forest	Mitigation and correction	0.69	0.75	0.72	0.96
	Mitigation	0.69	0.75	0.72	
	No trigger contained	0.99	1.00	1.00	
Logistic Regression	Mitigation and correction	0.75	0.65	0.69	0.96
	Mitigation	0.70	0.63	0.68	
	No trigger contained	0.99	1.00	1.00	

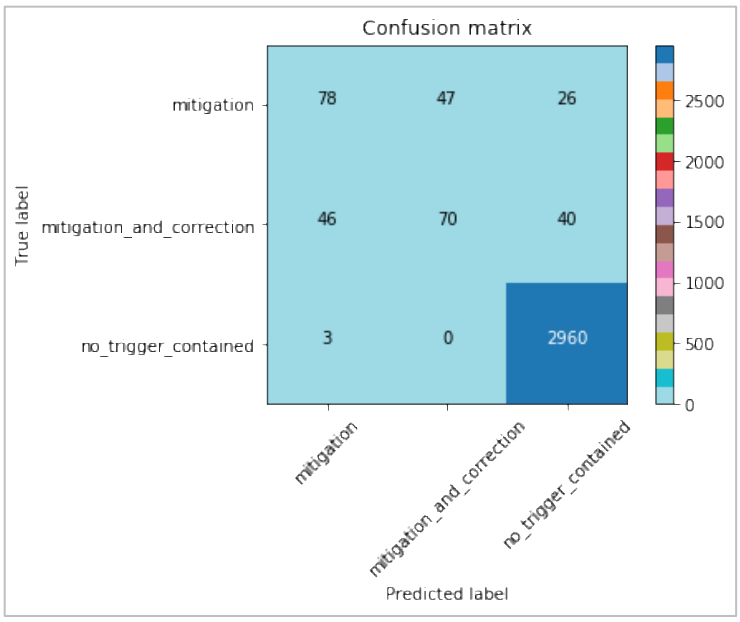

Figure 89. Confusion matrix for KNN model for predicting panic trigger labels on hurricane Michael dataset using CountVectorizer

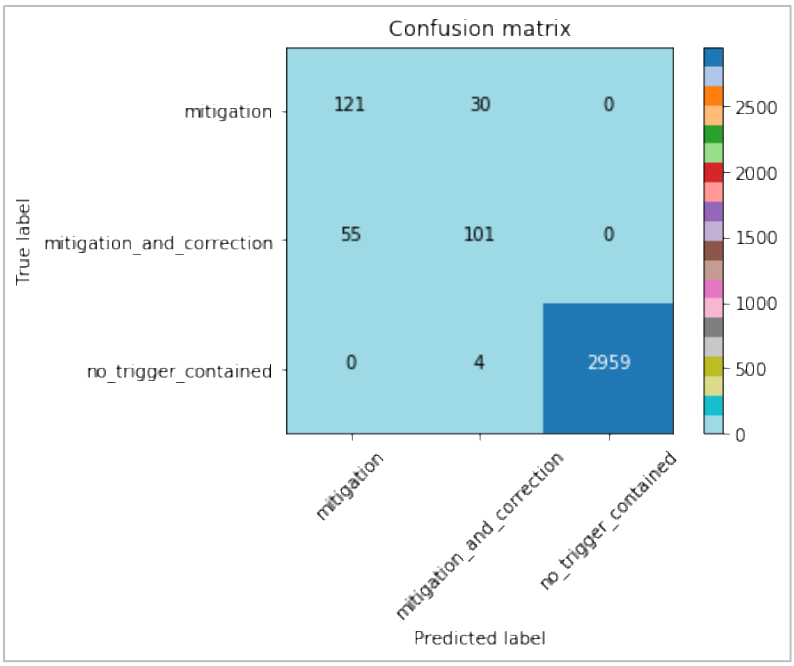

Figure 90. Confusion matrix for Decision Tree model for predicting panic trigger labels on hurricane Michael dataset using CountVectorizer

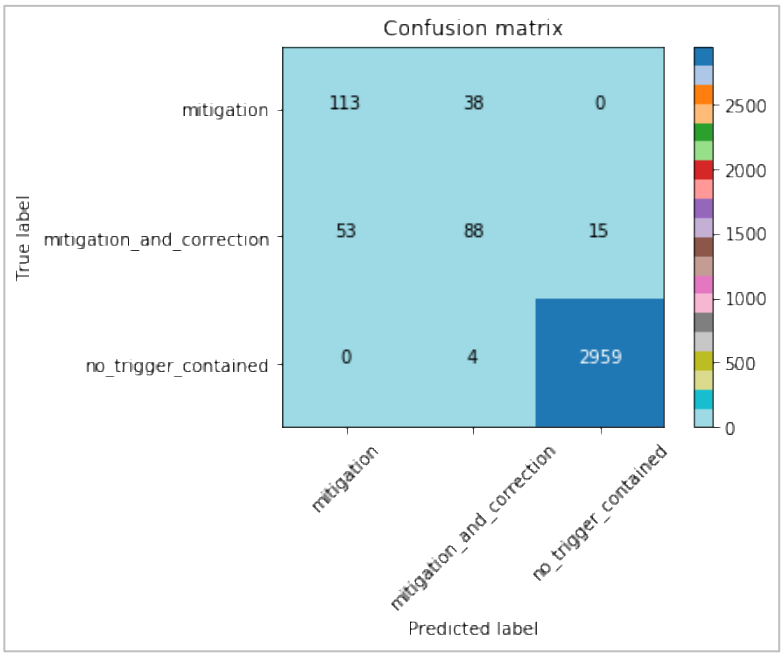

Figure 91. Confusion matrix for Random Forests model for predicting panic trigger labels on hurricane Michael dataset using CountVectorizer

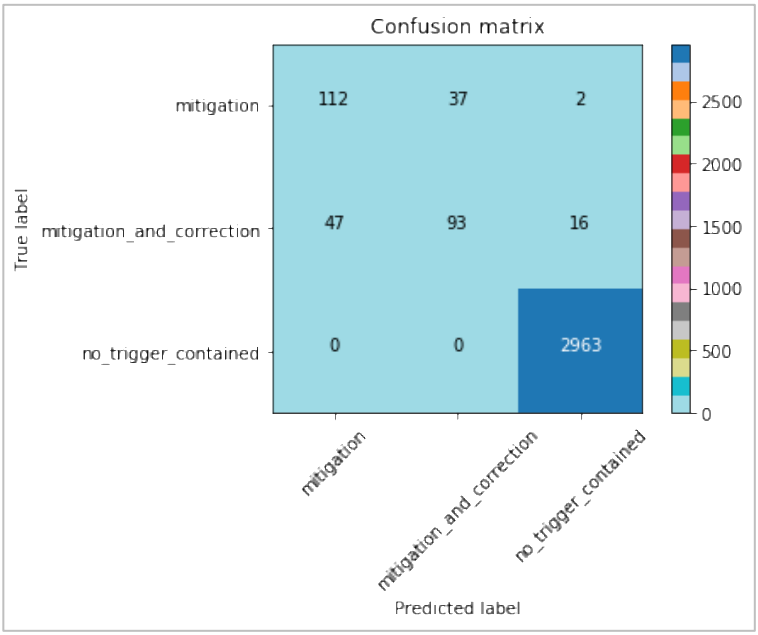

Figure 92. Confusion matrix for Logistic Regression model for predicting panic trigger labels on hurricane Michael dataset using CountVectorizer

Next, each model's performance in terms of False Acceptance Rate (FAR), False Rejection Rate (FRR), and Equal Error Rate (EER) is measured and shown in Table 33 and Figure 93. Figure 94 to Figure 97 show the ROC plots for each classification model.

We can summarize from Table 31 and Figure 93 that decision tree and random forest models show the highest accuracies and the lowest ERR. Figure 94 to Figure 97 show that they had high sensitivity and specificity rates in comparison to other classifiers. Logistic regression shows the lowest accuracy with high ERR and a low sensitivity and specificity rates. The KNN performance is the least with high EER rates.

Table 33

The FAR, FRR, and ERR rates for predicting panic trigger labels for Hurricane Michael dataset using CountVectorizer

Classifier	Labels	FAR	FRR	ERR
KNN	Mitigation and correction	0.015	0.48	0.051
	Mitigation	0.21	0.001	0.05
	No trigger contained	0.015	0.55	0.16
Decision Tree	Mitigation and correction	0.017	0.19	0.026
	Mitigation	0.0	0.0013	0.0013
	No trigger contained	0.010	0.35	0.050

Random Forest	Mitigation and correction	0.016	0.25	0.038
	Mitigation	0.048	0.001	0.008
	No trigger contained	0.013	0.43	0.048
Logistic Regression	Mitigation and correction	0.015	0.25	0.31
	Mitigation	0.058	0.0	0.02
	No trigger contained	0.25	0.031	0.057

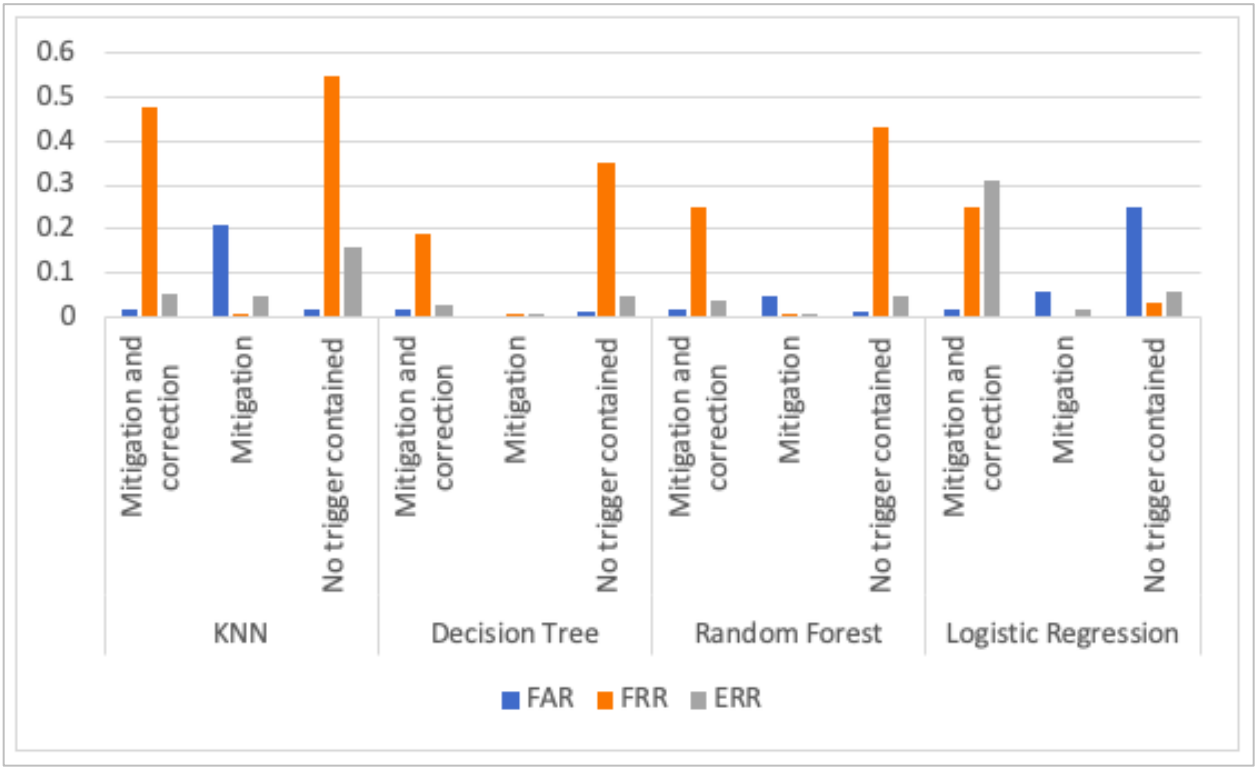

Figure 93. The FAR, FRR, and ERR rates for predicting panic trigger labels for Hurricane Michael dataset using CountVectorizer

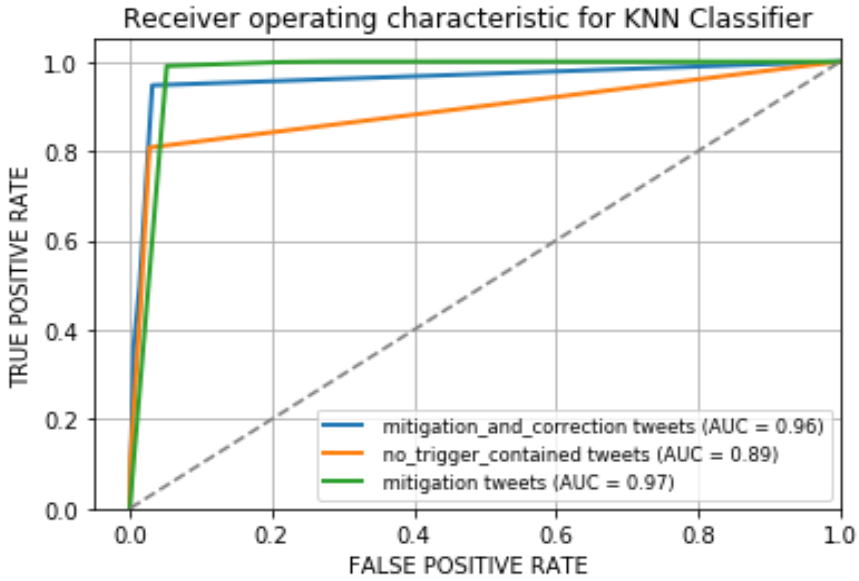

Figure 94. ROC plot for KNN model for predicting panic trigger labels for hurricane Michael dataset using CountVectorizer

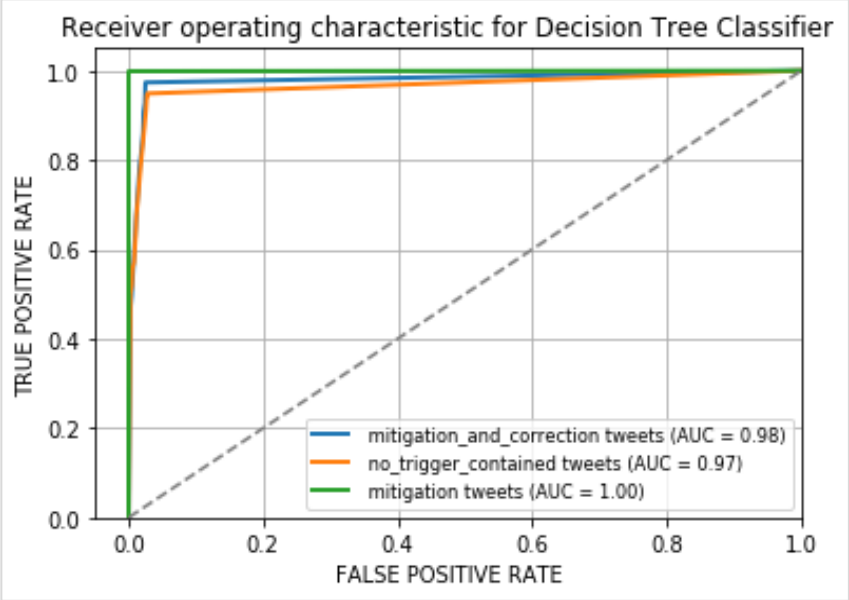

Figure 95. ROC plot for Decision Tree model for predicting panic trigger labels for hurricane Michael dataset using CountVectorizer

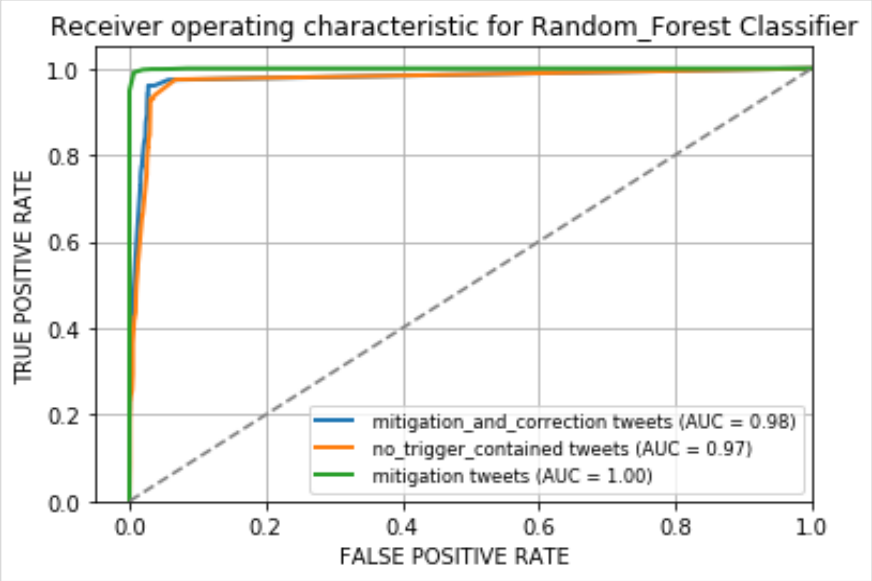

Figure 96. ROC plot for Random Forest model for predicting panic trigger labels for hurricane Michael dataset using CountVectorizer

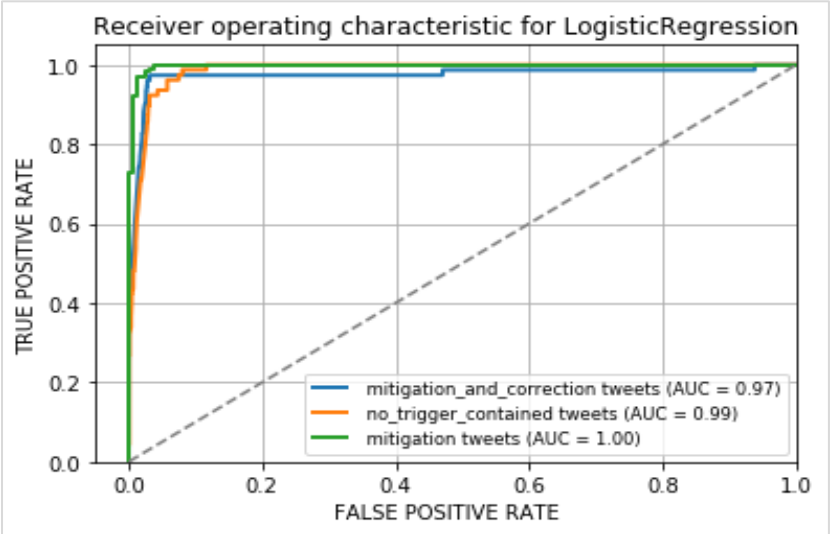

Figure 97. ROC plot for Logistic Regression model for predicting panic trigger labels for hurricane Michael dataset using CountVectorizer

To summarize the results for classifying panic trigger response labels, overall, using both TfidfVectorizer and CountVectorizer features to train the classifiers led to high accuracies and lower FAR, FRR, EER rates on both hurricane Florence and Michael datasets. The experiment

shows that classifiers using CountVectorizer features generate better accuracies and lower FAR, FRR, EER rates. Also, Decision Tree and Random Forest have the highest performance and lowest EER rate especially when using CountVectorizer features.

CHAPTER 5

Conclusion and Future Research

Twitter has become an effective platform for crowdsourcing and spreading critical information. Any maliciously intended activity, like spreading rumors on sensitive information needs to be detected and curbed from spreading immediately by emergency responders. Incorrect information can lead to chaos and panic among people around the disaster locations. Emergency responders need to have reliable knowledge about a disaster and its impact in a timely matter in order to take actions on preparation, evacuation, and recovery. However, it can be highly time and resource consuming for human to manually identify disaster-related tweets, assess the credibility of the tweets, and identify tweet that may trigger panic. In this research, a framework was developed to collect disaster-related tweets, categorize the collected data into disaster-related and not disaster-related tweets, assess the credibility and identify tweets that may trigger panic. The primary goals are to provide an approach in which Twitter data on disasters can be collected and classified in a timely manner to assist disaster managers and first responders.

The proposed Framework was used to collect tweets related to Hurricane Florence and Hurricane Michael from Twitter API. The datasets created will be made available for researchers who seek to investigate different aspects of these disaster events. This research first presented a labeling framework which automatically labeled tweets collected during hurricane disasters into disaster-related or not disaster-related. This labeling framework could be used to label tweets on future hurricane disasters according to their relevance to the disaster and would speed up the annotation process. Secondly, the framework implemented tweet classification using TfidfVectorizer and CountVectorizer features in order to determine which of these word vectorizers would provide better features for learning-based classifiers to produce the most

accurate classification. Further, for the learning-based system, a comparison was conducted between supervised machine learning classifiers. For the comparison, the performance of each classifier from the aspects of accuracy, precision, recall, and f-score were measured. Then the credibility of the disaster-related tweets was evaluated based on user-based features and content-based features. For each tweet, attributes like text messages and associated URLs, number of user followers, number of likes, and hashtags etc. were extracted in order to measure the credibility and trustworthiness of disaster-related tweets. The credibility evaluation relied on a 10-point scoring system to determine the level of tweet credibility. Supervised machine learning methods were implemented to predict the credibility of the tweets and model. A comparison between the classifiers regarding their performance was conducted. Finally, this research presented method to detect panic triggers in disaster-related tweet. Then learning-based classifiers were implemented to classify the tweets according to the response to panic triggers using two texts vectorizers: CountVectorizer, and TfidfVectorizer. Then the performance of the algorithms was compared.

The experimental results show Automated annotation can be sufficient for labeling in tweet as disaster-related and not disaster-related using predefined dictionary and as credible and not credible and Credibility using user-based and content-based features. It is possible to identify disaster-related tweets with high precision while maintaining fairly high recall especially when using CountVectorizer features. For classifying tweets into disaster-related and not disaster-related, using CountVectorizer word vectorizer has produced higher accuracies (98% on average) and low false rates especially when using Decision Tree and Random Forest models. For classifying tweet in terms of credibility, Random Forest and Decision Tree models have given the best predictions with high accuracies (96% on average) and the low false rates. Moreover, panic triggers can be automatically detected and classified with their corresponding tweet credibility.

For the classification of the tweets with panic triggers, Random Forest and Decision Tree have given the best predictions with high accuracies (95% on average) when using CountVectorizer features, and low false rates.

5.1 Future Research

This research can be extended with the following future research:

Data collection. Due to the limitation of Twitter API and the costly membership to collect data, only 26,000 tweets were collected. Future work will include collecting more disaster-related tweets, for example, one million tweets.

Image Recognition. The datasets collected do not include image data. Future work will include and analyzing credibility of images spread during natural disasters using machine learning methods.

Predicting Credibility Based on Social Network Features. User's behavior on Twitter, as well as, understand the user's sentiment will be analyzed and used as features to predict the credibility disaster-related tweets.

Panic triggers. The panic trigger terms can be expanded to cover a wide range of terms that are commonly used during hurricane disasters specifically, and other natural disasters generally.

Real-time Disaster-related Tweet Analysis. Future work will include analyzing of tweets and identifying panic triggers in real-time and generating reports to the emergency responders. This would be helpful for the emergency responders to respond to users and handle disasters timely.

References

Abbasi, M. A., & Liu, H. (2013, April). Measuring user credibility in social media. In International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction (pp. 441-448). Springer, Berlin, Heidelberg.

Afify, E. A., Eldin, A. S., Khedr, A. E., & Alsheref, F. K. (2019). User-Generated Content (UGC) Credibility on Social Media Using Sentiment Classification.

Alam, F., Ofli, F., Imran, M., & Aupetit, M. (2018). A Twitter Tale of Three Hurricanes: Harvey, Irma, and Maria. arXiv preprint arXiv:1805.05144.

Amstadter, A. B., Acierno, R., Richardson, L. K., Kilpatrick, D. G., Gros, D. F., Gaboury, M. T., ... & Buoi, L. T. (2009). Posttyphoon prevalence of posttraumatic stress disorder, major depressive disorder, panic disorder, and generalized anxiety disorder in a Vietnamese sample. Journal of Traumatic Stress: Official Publication of The International Society for Traumatic Stress Studies, 22(3), 180-188.

Ashktorab, Z., Brown, C., Nandi, M., & Culotta, A. (2014, May). Tweedr: Mining twitter to inform disaster response. In ISCRAM.

ASPR TRACIE, "Social Media in Emergency Response", U.S. Department of Health and Human Services, Office of the Assistant Secretary of Preparedness and Response, ID: 2789, 2015. https://asprtracie.hhs.gov/technical-resources/73/social-media-in-emncy-response/73

Becker, H., Naaman, M., & Gravano, L. (2010, February). Learning similarity metrics for event identification in social media. In Proceedings of the third ACM international conference on Web search and data mining (pp. 291-300). ACM.

Boukenze, B., Haqiq, A., & Mousannif, H. (2016, May). Predicting Chronic Kidney Failure Disease Using Data Mining Techniques. In International Symposium on Ubiquitous Networking (pp. 701-712). Springer, Singapore.

Breitinger, F., Stivaktakis, G., & Baier, H. (2013). FRASH: A framework to test algorithms of similarity hashing. Digital Investigation, 10, S50-S58.

Castillo, C. (2016). Big crisis data: social media in disasters and time-critical situations. Cambridge University Press.

Castillo, C., Mendoza, M., & Poblete, B. (2011, March). Information credibility on twitter. In Proceedings of the 20th international conference on World Wide Web (pp. 675-684). ACM.

Chandel, K., Kunwar, V., Sabitha, S., Choudhury, T., & Mukherjee, S. (2016). A comparative study on thyroid disease detection using K-nearest neighbor and Naive Bayes classification techniques. CSI transactions on ICT, 4(2-4), 313-319.

Chen, Y. W., & Lin, C. J. (2006). Combining SVMs with various feature selection strategies. In Feature extraction (pp. 315-324). Springer, Berlin, Heidelberg.

Collins, M., Neville, K., Hynes, W., & Madden, M. (2016). Communication in a disaster-the development of a crisis communication tool within the S-HELP project. Journal of Decision systems, 25(sup1), 160-170.

Conaire, C. O., O'Connor, N. E., & Smeaton, A. F. (2007, June). Detector adaptation by maximising agreement between independent data sources. In 2007 IEEE Conference on Computer Vision and Pattern Recognition (pp. 1-6). IEEE.

Damousis, I. G., & Argyropoulos, S. (2012). Four machine learning algorithms for biometrics fusion: A comparative study. Applied Computational Intelligence and Soft Computing, 2012, 6.

Das, A., Mallik, N., Bandyopadhyay, S., Bit, S. D., & Basak, J. (2016, March). Interactive information crowdsourcing for disaster management using SMS and Twitter: A research prototype. In 2016 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops) (pp. 1-6). IEEE.

De Albuquerque, J. P., Herfort, B., Brenning, A., & Zipf, A. (2015). A geographic approach for combining social media and authoritative data towards identifying useful information for disaster management. International Journal of Geographical Information Science, 29(4), 667-689.

Deekshatulu, B. L., & Chandra, P. (2013). Classification of heart disease using k-nearest neighbor and genetic algorithm. Procedia Technology, 10, 85-94.

Devin Soni, Towards Data Science, Introduction to k-Nearest-Neighbors, 2018, https://towardsdatascience.com/introduction-to-k-nearest-neighbors-3b534bb11d26

Elamvazuthi, I., Izhar, L., & Capi, G. (2018). Classification of Human Daily Activities Using Ensemble Methods Based on Smartphone Inertial Sensors. Sensors, 18(12), 4132.

F. Pedregosa, et al. "Sklearn.feature_extraction.text.CountVectorizer." 2013, https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html

F. Pedregosa, et al. "sklearn.feature_extraction.text.TfidfVectorizer." 2013, https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html

G. Mee, 'What is a Good Engagement Rate on Twitter?', 2018, https://www.scrunch.com/blog/what-is-a-good-engagement-rate-on-twitter

Gantt, P., & Gantt, R. (2012). Disaster psychology: dispelling the myths of panic. Professional Safety, 57(08), 42-49.

Ghosh, S., & Desarkar, M. S. (2018, April). Class specific tf-idf boosting for short-text classification: Application to short-texts generated during disasters. In Companion Proceedings of the The Web Conference 2018 (pp. 1629-1637). International World Wide Web Conferences Steering Committee.

Grace Pinegar, "How to Get Verified on Twitter in 2019", September 10, 2018, https://learn.g2.com/how-to-get-verified-on-twitter

Guan, X., & Chen, C. (2014). Using social media data to understand and assess disasters. Natural hazards, 74(2), 837-850.

Gupta, A., Kumaraguru, P., Castillo, C., & Meier, P. (2014, November). Tweetcred: Real-time credibility assessment of content on twitter. In International Conference on Social Informatics (pp. 228-243). Springer, Cham.

Gupta, M., Zhao, P., & Han, J. (2012, April). Evaluating event credibility on twitter. In Proceedings of the 2012 SIAM International Conference on Data Mining (pp. 153-164). Society for Industrial and Applied Mathematics.

Gupta, S., Basavaiah, M., & Fingerhut, J. (2011). U.S. Patent No. 8,032,529. Washington, DC: U.S. Patent and Trademark Office.

H. Polat, H. D. Mehr, and A. Cetin, "Diagnosis of chronic kidney disease based on support vector machine by feature selection methods." Journal of medical systems, 41(4), 55, April 2017.

Hajian-Tilaki, K. (2013). Receiver operating characteristic (ROC) curve analysis for medical diagnostic test evaluation. Caspian journal of internal medicine, 4(2), 627.

Heide, E. A. (2004). Common misconceptions about disasters: Panic, the disaster syndrome, and looting. The first 72 hours: A community approach to disaster preparedness, 337.

Hovland, C. I., & Weiss, W. (1951). The influence of source credibility on communication effectiveness. Public opinion quarterly, 15(4), 635-650.

Huang, Q., & Xiao, Y. (2015). Geographic situational awareness: mining tweets for disaster preparedness, emergency response, impact, and recovery. ISPRS International Journal of Geo-Information, 4(3), 1549-1568.

Imran, M., Castillo, C., Diaz, F., & Vieweg, S. (2018, April). Processing social media messages in mass emergency: Survey summary. In Companion Proceedings of the The Web Conference 2018 (pp. 507-511). International World Wide Web Conferences Steering Committee.

Imran, M., Castillo, C., Lucas, J., Meier, P., & Rogstadius, J. (2014, May). Coordinating human and machine intelligence to classify microblog communications in crises. In ISCRAM.

Imran, M., Castillo, C., Lucas, J., Meier, P., & Vieweg, S. (2014, April). AIDR: Artificial intelligence for disaster response. In Proceedings of the 23rd International Conference on World Wide Web (pp. 159-162). ACM.

Imran, M., Elbassuoni, S., Castillo, C., Diaz, F., & Meier, P. (2013, May). Practical extraction of disaster-relevant information from social media. In Proceedings of the 22nd International Conference on World Wide Web (pp. 1021-1024). ACM.

Ito, J., Song, J., Toda, H., Koike, Y., & Oyama, S. (2015, May). Assessment of tweet credibility with LDA features. In Proceedings of the 24th International Conference on World Wide Web (pp. 953-958). ACM.

J. Clement, Twitter, Survey and Published by: Twitter, Statista 2019, The Statistics Portal, "Number of monthly active Twitter users worldwide from 1st quarter 2010 to 1st quarter 2019 (in millions)". Last edited Aug 14, 2019, https://www.statista.com/statistics/282087/number-of-monthly-active-twitter-users/

James Parsons, "What is The Ideal Twitter Follower to Following Ratio?", Posted on April 1st, 2017, https://follows.com/blog/2017/04/ideal-follower-following-ratio

Jardaneh, G., Abdelhaq, H., Buzz, M., & Johnson, D. (2019, April). Classifying Arabic Tweets Based on Credibility Using Content and User Features. In 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT) (pp. 596-601). IEEE.

Joachims, T. (2002, July). Optimizing search engines using clickthrough data. In Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 133-142). ACM.

Khare, P., Burel, G., & Alani, H. (2018, June). Classifying crises-information relevancy with semantics. In European Semantic Web Conference (pp. 367-383). Springer, Cham.

Khare, P., Fernandez, M., & Alani, H. (2017). Statistical semantic classification of crisis information.

King, L. J. (2018). Social Media Use During Natural Disasters: An Analysis of Social Media Usage During Hurricanes Harvey.

Liu, J., Yu, K., Zhang, Y., & Huang, Y. (2010, December). Training conditional random fields using transfer learning for gesture recognition. In 2010 IEEE International Conference on Data Mining (pp. 314-323). IEEE.

Lu, W., Guttentag, A., Elbel, B., Kiszko, K., Abrams, C., & Kirchner, T. R. (2019). Crowdsourcing for Food Purchase Receipt Annotation via Amazon Mechanical Turk: A Feasibility Study. Journal of medical Internet research, 21(4), e12047.

Magellan Health Services (MHS), Inc., (2006), "Hurricane Preparedness and Resource Guide", Source: https://www.magellanassist.com/hurricaneguide.pdf

Meinert, J., Aker, A., & Krämer, N. (2019, January). The Impact of Twitter Features on Credibility Ratings-An Explorative Examination Combining Psychological Measurements and Feature Based Selection Methods. In Proceedings of the 52nd Hawaii International Conference on System Sciences.

Nair, M. R., Ramya, G. R., & Sivakumar, P. B. (2017). Usage and analysis of Twitter during 2015 Chennai flood towards disaster management. Procedia computer science, 115, 350-358.

Olteanu, A., Vieweg, S., & Castillo, C. (2015, February). What to expect when the unexpected happens: Social media communications across crises. In Proceedings of the 18th ACM conference on computer supported cooperative work & social computing (pp. 994-1009). ACM.

Palen, L., Anderson, K. M., Mark, G., Martin, J., Sicker, D., Palmer, M., & Grunwald, D. (2010, April). A vision for technology-mediated support for public participation & assistance in mass emergencies & disasters. In Proceedings of the 2010 ACM-BCS visions of computer science conference (p. 8). British Computer Society.

Patel, S. (2017). Chapter 2: SVM (Support Vector Machine)-Theory. Machine Learning, 101.

Petty, R. E. (2018). Attitudes and persuasion: Classic and contemporary approaches. Routledge.

R. Kathleen, "5 Tips for using Twitter during emergencies and natural disaster.", Director of Public Policy and Philanthropy, APAC, September 2018,

https://blog.twitter.com/en_sea/topics/insights/2018/5-Tips-for-using-Twitter-during-emergencies-and-natural-disaster.html

Raghuwanshi, A. S., & Pawar, S. K. (2017). Polarity Classification of Twitter Data using Sentiment Analysis. International Journal on Recent and Innovation Trends in Computing and Communication, 5(6), 434-439.

Reuter, C., & Kaufhold, M. A. (2018). Fifteen years of social media in emergencies: a retrospective review and future directions for crisis informatics. Journal of Contingencies and Crisis Management, 26(1), 41-57.

Ross, J., & Thirunarayan, K. (2016, October). Features for ranking tweets based on credibility and newsworthiness. In 2016 International Conference on Collaboration Technologies and Systems (CTS) (pp. 18-25). IEEE.

Rubin M., "Hurricane vocabulary: The difference between a typhoon, a cyclone, and a tropical storm", July 12, 2019, Quartz, source: https://qz.com/1071041/hurricane-michael-vocabulary-what-do-all-the-scientific-terms-mean/

Salem, A., Sharieh, A., Sleit, A., & Jabri, R. (2019). Enhanced Authentication System Performance Based on Keystroke Dynamics using Classification algorithms. KSII Transactions on Internet & Information Systems, 13(8).

Salman, O. A., & Hameed, S. M. (2018, November). Using Mouse Dynamics for Continuous User Authentication. In Proceedings of the Future Technologies Conference (pp. 776-787). Springer, Cham.

Salve, P., Sardesai, M., & Yannawar, P. (2018, September). Classification of Plants Using GIST and LBP Score Level Fusion. In International Symposium on Signal Processing and Intelligent Recognition Systems (pp. 15-29). Springer, Singapore.

Schreiner, C., Torkkola, K., Gardner, M., & Zhang, K. (2006, October). Using machine learning techniques to reduce data annotation time. In Proceedings of the Human Factors and Ergonomics Society Annual Meeting (Vol. 50, No. 22, pp. 2438-2442). Sage CA: Los Angeles, CA: SAGE Publications.

Shariff, S. M., Zhang, X., & Sanderson, M. (2014, April). User perception of information credibility of news on twitter. In European conference on information retrieval (pp. 513-518). Springer, Cham.

Stanley Dambroski, National Science Foundation (NFS), "How rumors spread on social media during weather disasters", September 18, 2018, https://www.nsf.gov/discoveries/disc_summ.jsp?cntn_id=296519&org=NSF&from=news

Starbird , K., Palen, L., Hughes, A. L., & Vieweg, S. (2010, February). Chatter on the red: what hazards threat reveals about the social life of microblogged information. In Proceedings of the 2010 ACM conference on Computer supported cooperative work (pp. 241-250). ACM.

Storch, E. A., Shah, A., Salloum, A., Valles, N., Banu, S., Schneider, S. C., ... & Goodman, W. K. (2019). Psychiatric Diagnoses and Medications for Hurricane Harvey Sheltered Evacuees. Community mental health journal, 1-4.

Stowe, K., Anderson, J., Palmer, M., Palen, L., & Anderson, K. (2018, July). Improving classification of twitter behavior during hurricane events. In Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media (pp. 67-75).

Stowe, K., Paul, M. J., Palmer, M., Palen, L., & Anderson, K. (2016, November). Identifying and categorizing disaster-related tweets. In Proceedings of The Fourth International Workshop on Natural Language Processing for Social Media (pp. 1-6).

Stroud, C., Hick, J. L., & Hanfling, D. (Eds.). (2013). Crisis Standards of Care: A Toolkit for Indicators and Triggers. National Academies Press.

To, H., Agrawal, S., Kim, S. H., & Shahabi, C. (2017, April). On identifying disaster-related tweets: Matching-based or learning-based?. In 2017 IEEE Third International Conference on Multimedia Big Data (BigMM) (pp. 330-337). IEEE.

Verma, S., Vieweg, S., Corvey, W. J., Palen, L., Martin, J. H., Palmer, M., ... & Anderson, K. M. (2011, July). Natural language processing to the rescue? extracting" situational awareness" tweets during mass emergency. In Fifth International AAAI Conference on Weblogs and Social Media.

Vieweg, S., Castillo, C., & Imran, M. (2014, November). Integrating social media communications into the rapid assessment of sudden onset disasters. In International Conference on Social Informatics (pp. 444-461). Springer, Cham.

Vieweg, S., Hughes, A. L., Starbird, K., & Palen, L. (2010, April). Microblogging during two natural hazards events: what twitter may contribute to situational awareness. In Proceedings of the SIGCHI conference on human factors in computing systems (pp. 1079-1088). ACM.

Wassmer, M., & Eastman, C. M. (2005). Automatic evaluation of credibility on the Web. Proceedings of the American Society for Information Science and Technology, 42(1).

Xia, X., Yang, X., Wu, C., Li, S., & Bao, L. (2012, May). Information credibility on twitter in emergency situation. In Pacific-Asia Workshop on Intelligence and Security Informatics (pp. 45-59). Springer, Berlin, Heidelberg.

Zhang, Y. (2015). Detect Spammers in Online Social Networks.

Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., & Torr, P. H. (2015). Conditional random fields as recurrent neural networks. In Proceedings of the IEEE international conference on computer vision (pp. 1529-1537).