Utilizing Deep Learning Techniques for Sentiment Analysis during disasters

MSc Research Project
Data Analytics

Mahesh Kumar Uppalapati
Student ID: 22176373

School of Computing
National College of Ireland

Supervisor: Mr. Hicham Rifai

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Mahesh Kumar Uppalapati
Student ID:	22176373
Programme:	Data Analytics
Year:	2023
Module:	MSc Research Project
Supervisor:	Mr. Hicham Rifai
Submission Due Date:	31/01/2024
Project Title:	Utilizing Deep Learning Techniques for Sentiment Analysis during disasters
Word Count:	8509
Page Count:	21

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Mahesh Kumar Uppalapati
Date:	30th January 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Utilizing Deep Learning Techniques for Sentiment Analysis during disasters

Mahesh Kumar Uppalapati
22176373

Abstract

The rise of social media has had a profound impact on sentiment analysis. With the widespread use of platforms like Twitter, Facebook, and Instagram, individuals can quickly and effortlessly express their thoughts, emotions, and responses to various events, including natural disasters. As a result, researchers have recognized the immense potential of social media data in understanding current public sentiment. Yet, the rapid growth of data and the need for comprehensive analysis have led to the use of more advanced methodologies. People today have the option to research topics outside of their social group. The availability of user reviews and public forums on the Internet has similarly freed businesses and organizations from relying on surveys and polls to acquire product ratings. Using machine learning to identify the tone of online comments has numerous practical uses and commercial appeals. It has been used in everything from consumer goods and services to healthcare and finance to social gatherings and political campaigns to, more lately, crisis management and natural disasters. In this work, to make predictions on the Twitter earthquake disaster dataset, we use machine learning models like Naive Bayes, Support Vector Machines (SVMs), and deep learning models like LSTMs. Our experimental results were able to demonstrate the superiority of deep learning models over machine learning models in all the evaluation metrics.

1 Introduction

Earthquakes are a catastrophic force of nature that deeply impacts the socio-economic status of nations. The unpredictable nature of these seismic events poses a significant obstacle to emergency management and public awareness (Chandrasekaran et al.; 2022). As our world becomes increasingly interconnected, social media platforms play a vital role as a source of information. These online channels provide a means for quick and widespread communication, including sharing people's personal experiences and emotions in the aftermath of an earthquake (Alanazi et al.; 2022). By utilizing sentiment analysis of social media posts regarding earthquakes, a deeper understanding of those affected's emotional needs and demands can be achieved. Thanks to the availability of data, machine learning techniques and profound learning have shown potential for predicting sentiment. Sentiment analysis, also called opinion mining, originates from natural language processing (NLP) and text mining. Initially, emotion recognition and classification in text primarily utilized lexicon-based approaches and basic rule-based systems. However, the multifaceted spectrum of human emotions (Pathak et al.; 2020a) cannot fully capture the complexity of language by these methods. Traditional methods of sentiment analysis may initially struggle to keep up with the huge amount of data generated during times of disaster. Data quality issues, such as noise and misinformation, are mostly common in the chaotic context of disasters. Yet another challenge in identifying the tone of an individual is that writing about disasters often offers complex emotional issues. Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) are two types of neural network structures that are most suited to overcome these challenges. Understanding delicate emotions requires the use of Recurrent Neural Networks (RNNs) since they can capture the sequential linkages present in the textual data. However, Convolutional Neural Networks (CNNs) are proficient at pinpointing regional similarities in disaster-related pictures and text (Ahmed and Sargent; 2014). Improves the models' ability to analyze sentiment during the setting of natural disasters by utilizing sophisticated training approaches, inclusive of transfer learning from pre-trained models. This study aims to assess the effectiveness of deep learning techniques for analyzing the sentiment of social media posts based on earthquakes. The other objective is to determine the most successful training methods and neural network structures for categorizing sentiments in the context of emergency management and public discourse (Earle; 2010). It is imperative to have an extensive understanding of the evolution of sentiment analysis and its different applications in disaster response to hold this study's significance (Oh et al.; 2010). The involvement of social media has greatly influenced sentiment analysis. People can instantly and easily share their feelings, thoughts, and reactions to events—including natural disasters—through visual and textual materials on social media sites like Facebook, Twitter, and Instagram. Researchers have acknowledged the importance of social media data to comprehend contemporary public opinion. However, the increasing amount of data has demanded the use of more complex methodologies, mainly in the domain of comprehensive sentiment analysis. The huge volume of data has been a difficult obstacle, driving the development of increasingly complex approaches to handle and analyze such huge volumes of data efficiently (Etaiwi et al.; 2021; Hasan et al.; 2018).

1.1 Motivation and Project Background

The most common and mainly researched area in the sentimental analysis is knowing the response i.e.; neutral sentiment or positive, or negative sentiment behind social media posts and blog entries. While this is the field that has provided great insight into people's thoughts on social media, its accuracy has been questioned when it comes to disasters and the complexities they bring. Based on its adaptedness deep learning is used in sentimental analysis and in detecting subtle patterns within textual data (Crooks et al.; 2013; Toriumi et al.; 2013). The usage of Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) is just a fraction of the cutting-edge deep learning methods that have been extensively studied in various research efforts for their efficiency in sentiment analysis and classification (Li and Rao; 2010; Radianti et al.; 2016). This study aims to improve and broaden our understanding of sentiment analysis by working on online social media conversations surrounding earthquakes. The possible influence of public sentiment on disaster response and communication makes this a crucial topic to explore. This work will consider a selection of suitable neural network architectures and training approaches.

1.2 Research Question

- To what degree do deep learning techniques adequately evaluate the emotional response to earthquakes expressed on social media compared to traditional machine learning techniques?
- Furthermore, would neural network architecture and training parameters influence performance quality on sentimental categorization in crisis management and public discourse?

1.3 Research Objectives

The main objective of this study is to investigate the use of deep learning technologies to perform sentiment analysis on earthquake-related social media. Emergency response and public communication teams could use these tools to speed up and improve decision-making in the hours after an earthquake, potentially saving lives and limiting damage.

In this project, aim to

- Investigate the use of deep learning technologies to perform sentiment analysis on earthquake-related social media.
- Performing efficient data pre-processing on tweets for efficient analysis of data for better prediction.
- Comparison between the performance of deep learning and machine models on performance.

1.4 Report Structure

The report structure is as follows:

- Section 2 of the study report explores cutting-edge deep learning sentiment analysis algorithms. This section summarizes recent research related to the use of deep learning in sentiment analysis.
- The methodology is explained in section 3. From the data-collection process to EDA.
- Section 4 describes the design specification and elaborately explains models used for prediction.
- Section 5 describes the implementation. It discusses sentiment analysis model construction, training methods, and issues experienced during this step.
- Section 6 discusses the results in elaborately.
- The conclusion describes a summary of the study, conclusion, and future works in section 7.

2 Literature Review

2.1 Introduction

Traditional sentiment analysis can be performed using traditional machine learning techniques such as Naive Bayes and Support Vector Machines (SVM) (Wang; 2017; Earle et al.; 2011). These methods work, but they may overlook language complexity and intricacies. Deep learning models on the other hand may understand complex language patterns and relationships by learning hierarchical attributes and representations from data using neural networks like RNNs or LSTMs (Mateus et al.; 2021). This approach works well with large datasets and complex situations to improve sentiment predictions.

Table 1: Comparative Analysis

Reference	**Research Problem**	**Method**	**Dataset**	**Results**
Jan et al., 2019	Sentiment Analysis on Twitter	DCNN & LSTM	3,813,173 tweets, (140,414 positive & 33,349 negative tweets)	Greater accuracy compared to Naive Bayes and support vector machines. Original sentence accuracy is higher than shuffled sentences.
Dong et al., 2020	Sentences Sentiments Analysis	CNN	Movie reviews from rottentomatoes.com	Model accuracy of 45.5%, outperforming existing models.
Pathak et al., 2020	Classification and segmentation of relevant features	Supervised and unsupervised learning	Supportive dataset	More than 80% accuracy
Robertson et al., 2019	Classification based technique	Regression based	Clustering and calcification dataset	Accuracy exceeding 90%
Current Research	Classification based ML & DL Techniques	SVM, Naïve Bayes, LSTM	Kaggle.com	LSTM accuracy surpasses SVM and Naive Bayes, exceeding 95%

Table 1 specifies a comparative analysis that explains the differences and accomplishments observed in the literature. Traditional methods and Deep learning could be combined to perform sentiment analysis. Hybrid models combine neural network learning with rule-based methods for improved results (Yue et al.; 2019; Yadav and Vishwakarma; 2020). This integration addresses language expression issues by merging traditional procedures with rule-based systems (Yadav and Vishwakarma; 2020). Finally, many models were present to improve sentiment analysis for different languages and with the help of different datasets by using different methods. Deep learning methods learn from features autonomously by utilizing neural networks, while traditional approaches need feature engineering and some explicit rules. Hybrid methods combine well (Qaiser et al.; 2021; Yue et al.; 2019). some examples of social media are Facebook, Twitter, and Instagram and how they have expanded and the different types of data that may be used in sentiment analysis (Robertson et al.; 2019). These are a few of many platforms that have given the user an easily accessible and powerful means of communication and their reactions to a wide range of events, inclusive of natural disasters. As the result of an unprecedented flood for its visual and textual data, that has given invaluable insight into user sentiment. People add delicate and sensitive expressions of feelings to communications by including emoticons, images, and other social media information in the inclusion of traditional text. Moreover, the continuous growth of data also starts as a major challenge for sentiment analysis (Robertson et al.; 2019; Qaiser et al.; 2021). This vast amount and variety of data that is found on social media networks could be challenging to process by utilizing traditional approaches. Advanced methods that can efficiently maintain and categorize large amounts of data are required for the analysis of these huge datasets. Hence, social media communications tend to be context-dependent and informal. Researchers require more tools and methods that can accurately interpret slang, acronyms, memes, and cultural nuance (Pathak et al.; 2020b; Rawat et al.; 2021). The impact of social media on present public opinion is far-reaching and substantial. It's a potent instrument for public opinion and market research polls because of the accurate and instant data that delivers on how people perceive and react to current products, events, or services. Product designers, marketers, and political campaign strategists can all benefit from analyzing social media data (Pandya et al.; 2021; Pathak et al.; 2020b). Another major use for this technology is crisis management, where it can help authorities and organizations scale how the public is reacting to calamities (Naqvi et al.; 2021; Ramadhani and Goo; 2017). Hospitals, businesses, governments, and emergency response teams will benefit from utilizing social media sentiment research. This explains why, in the age of social media, we need trustworthy and sophisticated sentiment analysis techniques.

2.2 Advanced techniques in deep learning applied to sentiment analysis

Deep learning methods, particularly Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) have been studied thoroughly in terms of sentiment analysis (Hasan et al.; 2018; Lavanya and Sasikala; 2021). Although CNNs are well recognized for their application in image processing, their extensive knowledge also makes them appropriate for text analysis. They are particularly valued for sentiment analysis, where the identification of particular phrases and keywords might reveal a person's intended tone. In other way, RNNs are very well suited for text analysis based on their ability to interpret sequential input (Kardakis et al.; 2021; Etaiwi et al.; 2021). They may analyze a statement or document while keeping the context in mind by picking out on dependent terms.

This feature helps RNNs to understand the sensitivity in sentiment like negation or sarcasm, these are challenging for other approaches to understand. There are huge advantages to using deep learning in sentiment analysis (Dong et al.; 2020; Jan et al.;

2019). They are good at finding sensitive data in textual data that regular machine learning methods may overlook. Sentiment classifications that take into account factors like context, tone, and sentiment polarity perform better. In inclusion, deep learning models can adjust for new linguistic norms and new ways of expressing emotions, making them useful in the ever-changing environment on social media. Apart from their benefits, deep learning approaches for sentiment analysis are not without drawbacks (Bianco et al.; 2018; Chandrasekaran et al.; 2022). For achieving optimal performance, the domains with sparse annotations require a huge amount of labeled data for training. Not only that these can be very resource-intensive and expensive, requiring a lot of storage space and processing time. Therefore it can be challenging to understand the accurate reasons behind deep learning models' predictions, especially in difficult circumstances, and interpretability remains a challenge. As a result, choosing the right deep learning architecture and fine-tuning it for a given sentiment analysis task is crucial (Demirci et al.; 2019). Successful sentiment analysis by using deep learning approaches based on striking a balance between these complications and advantages. These are unique challenges to analyzing public opinion during complex events like natural catastrophes.

2.3 Analysing and comparing different deep learning architectures

Comparing deep learning architectures like Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs) is important for sentiment analysis during the aftermath of natural disasters (Pathak et al.; 2020a; Alanazi et al.; 2022). Based on their superior pattern and picture recognition capacities, convolutional neural networks(CNNs) are very useful for processing disaster-related visual data. In other terms, recurrent neural networks(RNNs) will understand the complicated sentiments of material written once the disaster happens since they capture textual data context. Though recurrent neural networks (RNNs) (Etaiwi et al.; 2021) are better at picking up on linguistic sensitivity, convolutional neural networks (CNNs) (Nezhad and Deihimi; 2019) excel at spotting visual clues. By including the two architectural styles, an accurate complete method of sentiment analysis will be achieved. Moreover, obstacles presence will be there, such as the accountability of deep learning models and their necessity for the huge amounts of labeled data, both of which will limit their usefulness in emergency situations (Hasan et al.; 2018; Lavanya and Sasikala; 2021)).

3 Methodology

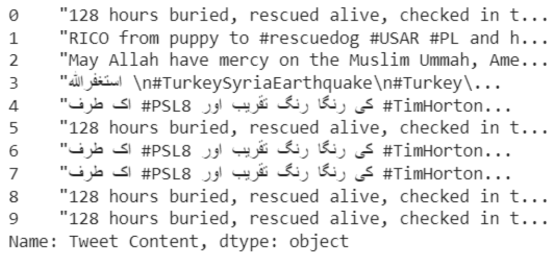
The complete data mining procedure in the current study of sentiment analysis follows a set of steps in terms of CRISP-DM methodology.

3.1 Data Collection

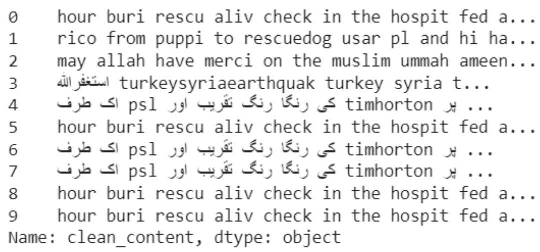
The data collected from Kaggle is about the "Turkey Syria Earthquake Tweets Dataset" can be found in the link [1].

There are 42,650 unique tweet IDs in the dataset, all of which represent tweets about the Turkey earthquake that appeared on Twitter between February 13 and February 15,

[1] `https://www.kaggle.com/datasets/kumari2000/turkey-syria-earthquake-tweets-dataset/`



(a) Before pre-processing



(b) After pre-processing

Figure 1: Figure 1a refers to the text before pre-processing and Figure 1b refers to the text after applying various pre-processing steps.

2023. Trackmyhashtag(ref), was used to gather the dataset. There is (How much) information associated with each tweet, which includes the tweet ID, the tweet URL, the tweet's timestamp, the tweet's content, and more. Graphs and dashboards can be generated to visually represent the data, and thereafter analyze it to deduce insights regarding individuals' emotional responses and any discernible patterns after the earthquakes in Turkey and Syria.

3.2 Exploratory Data Analysis

Exploratory data analysis, often known as EDA, is necessary to unearth previously unknown patterns and connections hidden within a dataset. Twitter sentiment, which includes tweets and their associated sentiments (negative, positive, or neutral). Data features like emotion distribution, tweet lengths, and frequencies of the words can be visually inspected.

Figure 2 can show many tweets with positive, negative, and neutral and can observe negative tweets are higher than the others.

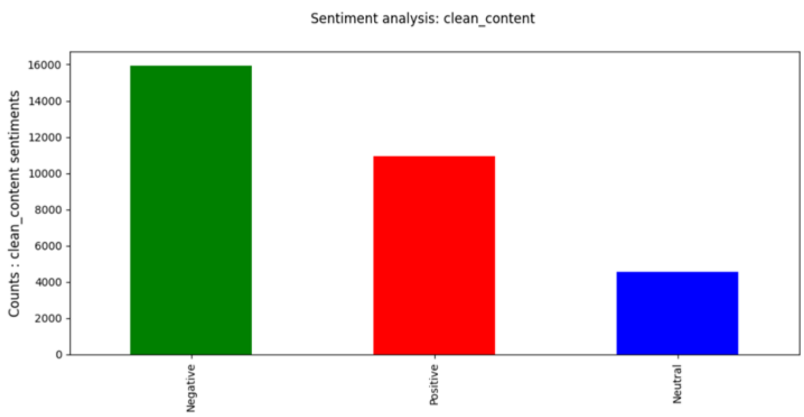


Figure 2: Count of Output label

Given the tweets taken from Twitter, it has lots of anomalies that need to be addressed. So we aim to perform actions to clean the content to fit the project:

- Removing links or URLs that are present in the text, to ensure that web addresses do not interfere with the analysis.

- Removing the digits or numeric values that are not required for the analysis
- removing punctuation marks and pre-processing them for further cleaning and preparing for analysis.
- performing tokenization using word_tokenize, which involves breaking the paragraph into individual words or tokens.
- Remove Stop Words: Removing common stop words to focus on more meaningful words and reduce noise in the text.

After applying all the above pre-processing to the data, we created a cleaned input of the original paragraph that is ready for further analysis or modeling, as shown in Figures 1a and 1b.

Figure 3 shows the number of unique languages and corresponding tweets, and we can understand that most of the tweets belong to the English language.

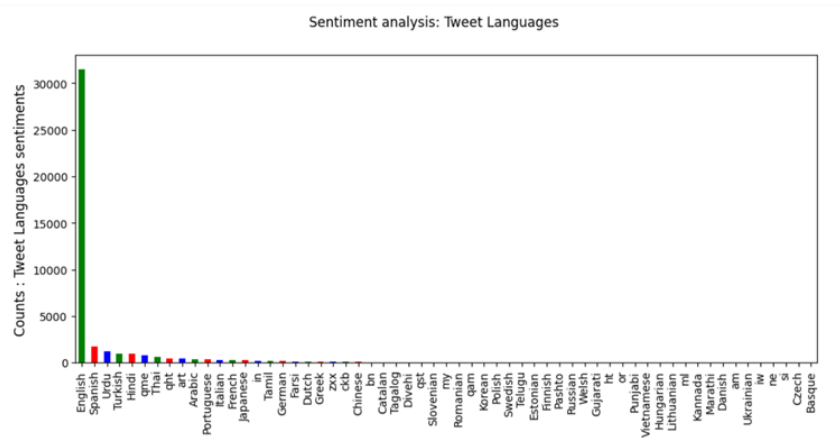


Figure 3: Count of languages

In a word cloud in figure 4 a visual representation of text data-words is displayed in varying sizes, with more frequently used words appearing larger and less often used words appearing smaller. It is a method for drawing readers' attention to the most important or conspicuous terms in a book, providing a succinct yet understandable summary of the most pertinent terms.

Preparing textual content for efficient input into the neural network necessitates a series of critical techniques known as "pre-processing" for deep learning models used in Twitter Sentiment Analysis. Cleaning the text entails standardizing capitalization and removing any non-printing letters or symbols. The process of tokenization includes breaking down the text into individual words, or tokens, before removing stop words. This helps to reduce fluff and improve readability. Padding is important to ensure that sequences have a consistent length, which is a prerequisite for deep learning models. Words are represented as compact vectors that capture semantic relationships using pre-existing word embeddings like GloVe and Word2Vec. For the sake of model processing, a vocabulary is built in which words are assigned numeric values, and sentiment labels and text sequences are also converted into numerical representations. To speed up the learning process, data is first sorted into training, validation, and test sets, and then shuffling and batching methods are used. Data normalization, class feature engineering,

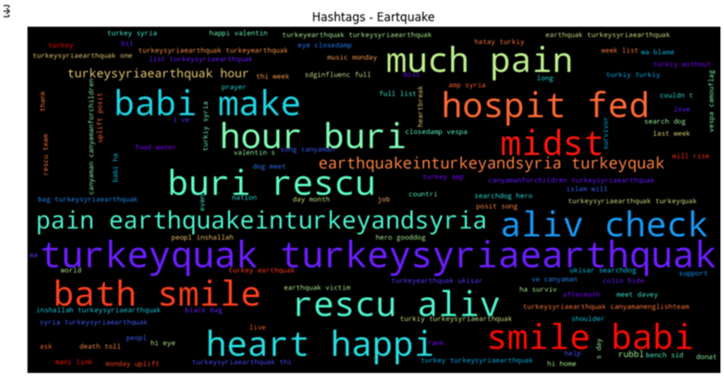


Figure 4: Word cloud (Nezhad and Deihimi; 2019)

imbalance correction, and missing data management all are potential further steps. For data type consistency and categorical features one-hot encoding is used and the same is checked. The precision of the dataset is based on pre-processing and preparation for input to the LSTM model based heavily on the results of thorough data verification. For the model to precisely assess Twitter sentiment data, this is important.

4 Design specifications and Models

Different important measures have to be considered for the implementation of sentiment analysis by using Long Short-Term Memory (LSTM) networks, Support Vector Machines (SVM), and Naive Bayes. Text data is preprocessed by tokenization, deduplication, and employing techniques those are lemmatization or stemming constitutes an initial stage in Naive Bayes and SVM. TF-IDF are some of the Text analysis approaches that are used to extract valuable information from the text data. Sequentially, models go through training utilizing a partitioned dataset, and the performance of the models is evaluated on a separate testing set. The LSTM preparation phase involves performing sequence padding, tokenization, and using pre-trained word embeddings (if they are available). The architecture of the model consists of LSTM layers for embedding layers, dense layers, and latent state transition. Long Short-Term Memory (LSTM) models are similar to Naive Bayes and Support Vector Machines (SVM), these are trained using a dataset that has been divided into segments, and the performance is discussed using certain metrics. Finally, the output of each model is calculated and contrasted, by taking into consideration metrics such as F1-score, recall, precision, and accuracy to determine the superior-performing approach.

To efficiently comprehend and classify emotions expressed through the text data, it is imperative to create a well-designed model and sequentially implement it to perform sentiment analysis on the incoming data in real time. Refining, optimizing hyper-parameters, and continuously iterating are all essential to optimizing these models for application in many scenarios.

4.1 Naive Bayes Algorithm

To classify data, naive Bayesian models are used as a well-known Bayesian theorem and make explicit assumptions about the independence of data features. At first sight, this

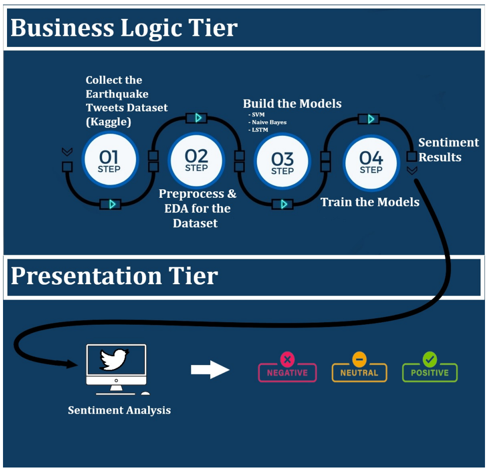


Figure 5: Design specifications

may seem like an unusual theory. This is generally accepted that the word before the

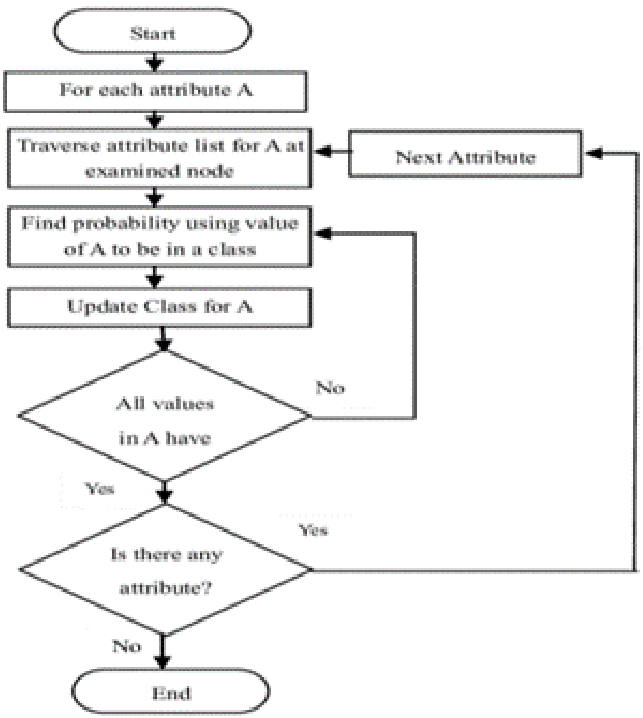


Figure 6: Naive-bayes Algorithm (Nezhad and Deihimi; 2019)

current word affects the current word and the next preceding word after it. However, these beliefs simplify mathematics and work very well during practice.

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)} \quad (1)$$

Classification is the act of grouping data into categories based on their similarities. A Naive Bayes classifier is a probabilistic classifier that determines the likelihood of multiple attributes of data being associated with a given class. The British scientist Thomas Bayes developed a theory of probability that uses historical data to predict future outcomes; this theory is used by the Naive Bayes classification method.

The theorem states that the conditional probability of event A, given event B, is equal to the product of the likelihood of event B given event A and the probability of event A, divided by the probability of event B. Learning and testing are distinct stages in the Naive Bayes categorization process. During the training phase, a model is constructed by utilizing several data points from the information category. During the phase of testing, the model is generated and evaluated using novel data to assess its performance.

4.2 SVM Algorithm

Support vector machine(SVM), which is widely utilized in supervised learning techniques specifically made for classification tasks. For distinguishing and categorizing the incoming inputs, this system has utilized a hyperplane. This results in the improvement of outputs. Let us draw a graph with the space of n dimensions that assign the value of each new feature is described as the value of a particular coordinate. Sequentially, the difference between these two categories might determine the optimal hyperplane.

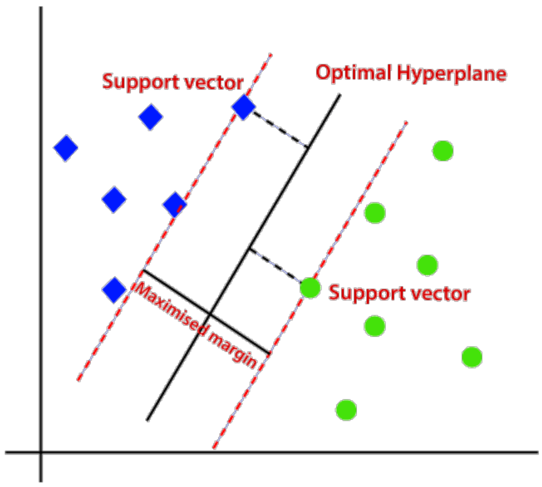


Figure 7: SVM Classification (Qaiser et al.; 2021)

The Support Vector Machine (SVM) identifies the hyperplane which potentially separates two classes by maximizing the margin.

$$(\mathbf{x} \cdot \mathbf{y}) + b = \sum_i y_i(\mathbf{x} \cdot \mathbf{y}) + b = 0 \tag{2}$$

where:

$$\begin{aligned} \mathbf{x} &= \text{nth-dimensional input vector,} \\ y_i &= \text{its output values,} \\ \mathbf{w} &= \text{weight of vector (the normal vector),} \\ a_i &= \text{Lagrangian multiple.} \end{aligned}$$

If $\mathbf{w} \cdot \mathbf{x} + b \geq 0$, then it is considered as positive class; otherwise, it is considered to be Negative Class.

Classification technique which is used here is Support Vector Machine (SVM) which is based on multiple mathematical principles. the basic goal of this technique is to identify a hyper plane with in the feature space which optimally differences between two classes. The main goal is to create a decision boundary which has the greatest possible margin, that refers to the distance between the closest data points and the hyper plane from each class. Support vectors are the data points that are nearest to the hyperplane and always have a crucial impact on defining its orientation and position. The main goal of Support Vector Machines (SVM) is to maximize the margin between the data classes while decreasing misclassification errors. The weight vector's norm is minimized, and that can be achieved by solving the optimization problem and by clarifying that all the data set points are distinguished successfully. SVM depends on a "soft margin" strategy, which is regulated by the hyperparameter 'C', to strike a difference between classification errors and margin size when confronted with non-linearly separable noise or data. Support Vector Machines (SVM) can process non-linear data just by converting it into a higher-dimensional space by utilizing kernel functions. The mathematical rationale behind SVM enables it to accurately categorize data in both non-linear and linear situations, making it a potent tool as well as versatile for a range of machine learning tasks.

4.3 LSTM Algorithm

The LSTM stands for "long short-term memory". For a particular kind of problem that is based on sequence predictions, there is one particular subtype of RNN that is extraordinarily effective. this is done by analyzing the text that has come before and predicting which can be further based on a prediction base analysis. Irrelevant data is filtered out and the long short-term memory is capable of preserving useful information throughout the processing of input data. The problem of vanishing gradients, which can hinder the capacity of normal RNNs to remember the information for an extended time period, this is the reason that LSTMs were particularly designed to solve the problem. Forget gate is where the crutial information is stored, while the system discards the

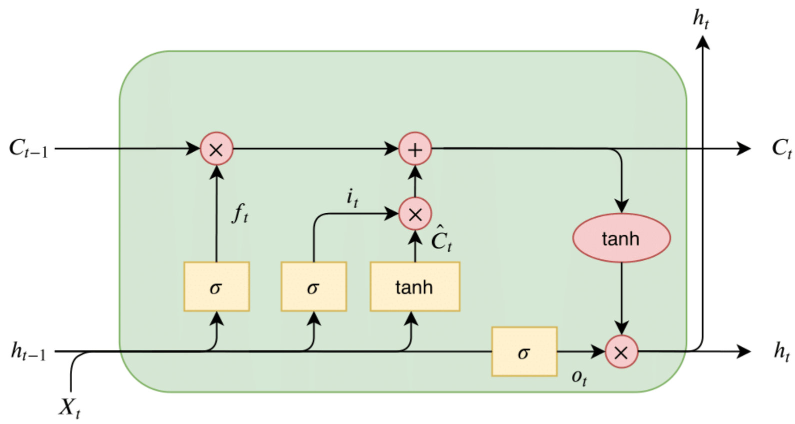


Figure 8: Basic Structure of LSTM Classification (Mateus et al.; 2021)

information which is no longer relevant.

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f) \tag{3}$$
$$i_t = \sigma(w_i[h_{t-1}, x_t] + b_i) \tag{4}$$
$$c_t = \tanh(w_c[h_{t-1}, x_t] + b_c) \tag{5}$$
$$o_t = \sigma(w_o[h_{t-1}, x_t] + b_o) \tag{6}$$
$$h_t = o^t \cdot \tanh(c_t) \tag{7}$$

In order to generate the forget factor (ft), the preceeding hidden gate (ht-1) and the cur rent vector of event (xt) were put together to produce the [ht-1, xt] pair, which is further then multiplied with the help of weight matrix (Wf), and then it was sent through the sigmoid activation function based on a certain amount of bias. The input gate is important in storing the current state of the information. The input gate serves to determine the magnitude of new information being escorted by the input. To construct it, the up coming hidden gate, (ht-1), and the current event state, (xt), are combined into a single value. This is the value which is then multiplied with the weight matrix and passes from the sigmoid activation function, including with a specific bias. To produce (ct), the inclusion of the previous hidden state (ht-1) and the present input (xt) is multiplied by the weight matrix and passed from the tanh activation function, along with a bias term. For processing subsequential data, researchers utilized the Long Short-Term Memory (LSTM) technique. Because it accepts the significance of word order, it masters at analysing text for sentiment. Because of the built-in memory cells present in LSTMs, they are able to make the far-flung connections within text that are important to understanding the linguistic context data. Gates, including input, output, and forget, stop the flow of data in and out of these memories. Word embedding's are numerical representations of vectors for words, and they are basically created from the input text. One of the major benefits of the LSTM algorithm is that it can easily adjust to new document or phrase lengths with its flexible processing of sequences of varied lengths. Back propagation through time (BPTT) is a model used for training Long Short-Term Memory (LSTM) models by propagating backward gradients through the sequential data in order to update the model's weights. LSTM gates' sigmoid activation functions play a very important role in controlling how the information is conveyed and how decisions can be made. Many LSTM models are implemented for sentiment analysis and are programmed to terminate with a single neuron with a sigmoid activation function. Because of the setup, they are great for determining whether a given sentiment is negative or positive. Cross-entropy loss is mostly used to enhance Long Short-Term Memory (LSTM) models for training. The loss function calculates how far the model's predictions can deviate from the actual labels. Metrics like precision, recall, F1-score, and accuracy are used to evaluate LSTM models for sentiment analysis because they are able to expose important information about how well the models can classify sentiment in textual data.

5 Implementation

5.1 Architectural Details

5.1.1 LSTM

The code was created in a Keras model for sentiment analysis by using a mix of recurrent neural network and convolutional neural network layers. Using a Sequential model, the neural network could be built from the scratch. Word indices are first transformed into compact vectors with the help of an Embedding layer. A 1D convolutional layer where 32 filters were used and a kernel size of 3 is later used. After this, an activation function called a rectified linear unit (ReLU) was applied. Then, dimensionality is dropped via max pooling. The 32-unit Bidirectional Long Short-Term Memory (BiLSTM) layer is utilized in the model is that gives the network its impressive ability to effectively capture sequential patterns in both the forward and backward directions. Dropout layer is used to deal with the over fitting patter and the dropout rate is 0.4, a Dense layer will be added, which uses a softmax activation function to give permissions for the classification of input data into three different emotion categories. The Stochastic Gradient Descent (SGD) optimization is done with the pre-determined learning rate and momentum. This architecture is well designed for sentiment analysis tasks hence it seeks to combine the advantages of recurrent and convolutional layers will efficiently capture both fine-grained patterns and long-term dependencies in textual input. For LSTM following attributes and values are used

- epochs=20
- learning_rate = 0.1
- decay_rate = 0.05
- momentum = 0.8
- vocab_size = 5000
- embedding_size = 32

5.1.2 SVM

Support Vector Machine (SVM) classifieris used for sentiment analysis that employ stochastic gradient descent optimization through the Scikit-learn module. This process is present within a pipeline with three major steps. Starting with the textual data input, the count function is used to create a matrix that counts how regularly each word appears in the given documents. Using frequency-inverse document frequency (TF-IDF), important attention is given to rare words. At the end, want to run the linear Support Vector Machines (SVM) trained with a small number of iterations and a small alpha parameter via the SGDclassifier. And then, the 'Xtrain' and 'Ytrain' training data are used to fine-tune the pipeline.

After predictions, we will always produce a full classification report that details F1-score, recall, and precision information for each emotion data category. The list ['Negative', 'Neutral', 'Positive'] defines the sentiment classes. This pipeline incorporates text preprocessing, SVM classification, and feature extraction, delivering comprehensive techniques for sentiment analysis with an emphasis on efficacy and generalization. For SVM following attributes and values are used:

- kernel='rbf'
- degree=3
- gamma='scale'

- coef=0.0
- shrinking=True

5.1.3 Naive Bayes

Sentiment analysis using a Multinomial Naive Bayes classifier within a Scikit-learn pipeline, employing popular text processing techniques. Three distinct phases are present in pipeline. It all begins with the counting the vectors, which takes your raw text and turns it into a matrix of token accurately counts on reflecting how often each word appears in your documents. Term Frequency-Inverse Document Frequency (TF-IDF) normalization for the count matrix, giving greater weight to words that are more discriminative for the job of sentiment analysis. When working with discrete data for counting of words, naive bayes classifier is used because it implements the Multinomial Naive Bayes technique.To use it, simply supply training data ('Xtrain' for features and 'Ytrain' for labels) and the pipeline will be trained automatically.

Using the predictions, the model makes predictions about the sentiment labels for the same training data after training is complete. The model's performance may be assessed by comparing the predicted labels ('y pred') to the actual labels ('Ytrain') and then printing the accuracy of the model. Clearly depicting the three distinct emotional states, the sentiment classes are specified as ['Negative,' 'Positive,' and 'Neutral']. Because of its ease of use and proven track record with text data, this method is frequently deployed for sentiment analysis tasks, particularly when working with vast and sparse feature spaces. For naive bayes following attributes and values are used

- priors=None
- var_smoothing=le-9

6 Evaluation

This study shows how deep learning methods can be used to evaluate the emotional responses of social media users to earthquakes. Techniques from machine learning and deep learning are applied, and the results are examined. Sentiment analysis is one area in which deep learning techniques, in particular neural network topologies, have shown significant promise in natural language processing applications. But it's important to give serious thought to how well they capture the complex emotional reactions to earthquakes on social media.

In Figure 9, we can see that naive-bayes predictions of classes negative, positive, and neutral with an accuracy of 88%. There were a few false positives and false negatives as well still SVM managed to get good performance. As naive-bayes assumes independence across the attributes, it might lead to bad performance in tasks where attributes have a relation among them

In figure10, the Naive Bayes model has shown the F1 score of 88%, recall of 88%, precision of 90%, and accuracy of 88% which is comparatively poor performance compared to SVM and deep learning models. The complexities of the neural network architecture utilized, the training parameters selected, and the contextual relevance of the dataset used to construct the model all affect how effective these strategies are. Through exploratory data analysis, the insight information from the tweets has been gathered, and it can be

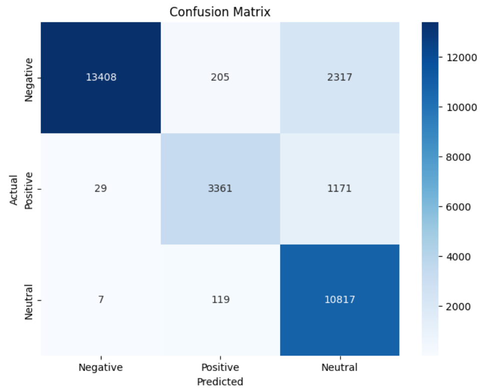


Figure 9: Naive Bayes Classifier Confusion matrix

```
accuracy 0.8775847808105872
              precision    recall  f1-score   support

         neg       1.00      0.84      0.91     15930
         pos       0.91      0.74      0.82      4561
       unsup       0.76      0.99      0.86     10943

    accuracy                           0.88     31434
   macro avg       0.89      0.86      0.86     31434
weighted avg       0.90      0.88      0.88     31434
```

Figure 10: Naive Bayes Classifier performance metrics

helpful to comprehend the emotions. Wordcloud provides information from numerous tweets. The model is trained and tested using a dataset that is selected at random. Accuracy, precision, recall, and f1 score are calculated for LSTM, SVM, and naive Bayes models. The information regarding the conducted experiments is shown in the table below.

In Figure 11, we can see that SVM can predict negative, positive, and neutral with an accuracy of 91%. There were a few false positives and false negatives as well still SVM managed to get good performance.

In Figure 12, shows precision, recall, and f1-score metrics for the SVM classifier, which demonstrates the effectiveness of the proposed approach as most of the values are better than naive-bayes. For sentiment analysis, LSTMs are preferred over SVM and Naive Bayes due to their capacity to handle sequences of varying lengths and identify sequential dependencies. They have a strong grasp of temporal dynamics, quickly recognize pertinent elements, and comprehend contextual subtleties. LSTMs enable end-to-end learning and do away with the necessity for human feature engineering, in contrast to conventional algorithms. Task difficulty, data volume, and interpretability constraints are some of the elements that influence the decision between LSTMs and traditional approaches. Fig. 13 discusses whether the mode loss is being optimized properly or not using loss plots.

The results reported in 2 demonstrate the performance of machine learning models, namely LSTM, SVM, and Naive Bayes. The deep learning-based LSTM model shows superior performance over other models with an accuracy of 95.85%, precision of 95.92%, recall of 95.85%, F1 score of 95.89%. This shows the LSTM model can effectively predict instances. In comparison with SVM and Naive Bayes models, while still demonstrating

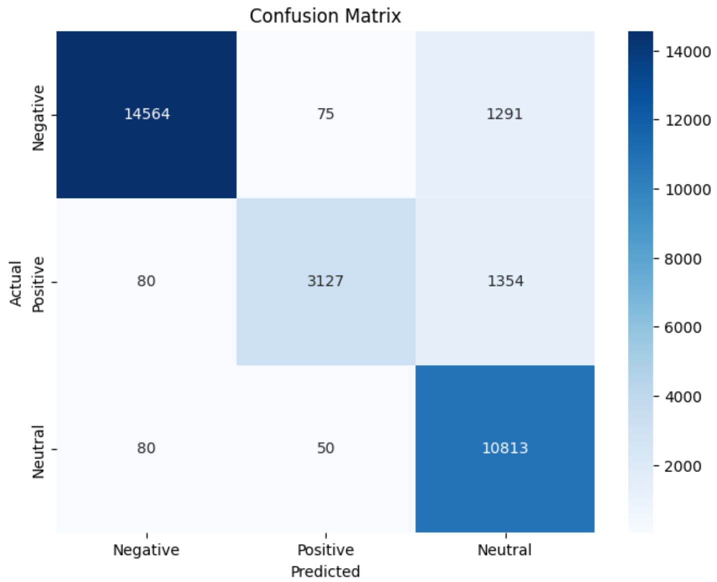


Figure 11: SVM Classifier Confusion matrix

```
accuracy 0.9067888273843608
              precision    recall  f1-score   support

         neg       0.99      0.91      0.95     15930
         pos       0.96      0.69      0.80      4561
       unsup       0.80      0.99      0.89     10943

    accuracy                           0.91     31434
   macro avg       0.92      0.86      0.88     31434
weighted avg       0.92      0.91      0.91     31434
```

Figure 12: SVM Classifier performance metrics

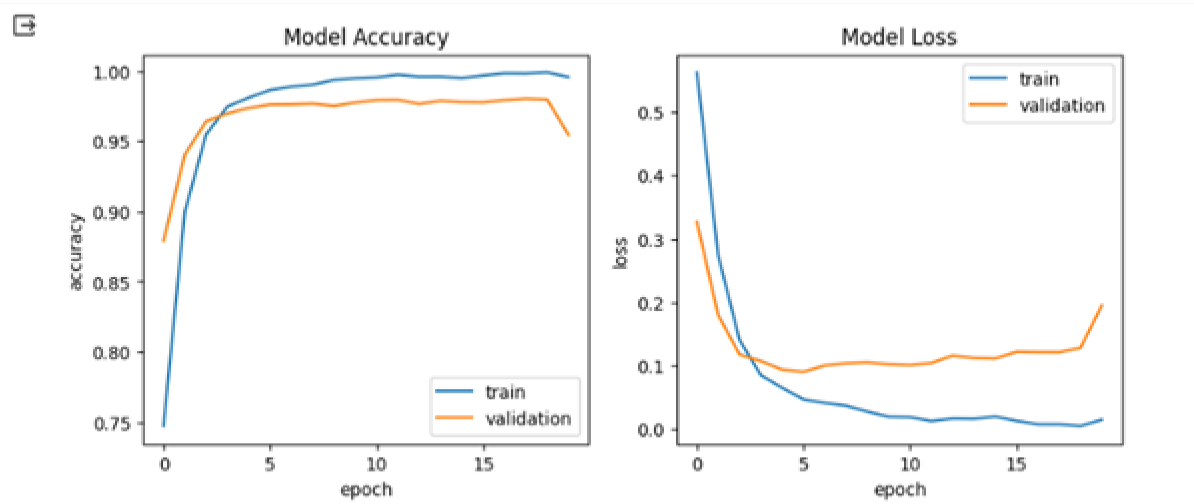


Figure 13: Model Accuracy and Loss

Table 2: Experiment Results

Experiments	Accuracy	Precision	Recall	F1 Score
LSTM	0.9585	0.9592	0.9585	0.9589
SVM	0.91	0.92	0.91	0.91
Naive Bayes	0.88	0.90	0.88	0.88

satisfactory results, they are less than or behind the LSTM in terms of accuracy and various other metrics. The SVM can achieve an accuracy of 91% and precision, recall, and F1 scores of 92%, 91%, and 91%, respectively. The naive Bayes model have shown the F1 score of 88%, recall of 88%, precision of 90%, and an accuracy of 88%.

7 Conclusion and Future Work

Overall, the study "Utilising Deep Learning Techniques for Sentiment Analysis" has worked better on how cutting-edge techniques can be applied to classify and comprehend sentiments. The outputs results in a high degree of validity confidence and improve the efficiency of deep learning methods, such LSTM, in getting the complex emotions. Deep learning approaches are core to traditional algorithms like Naive Bayes and SVM for addressing sequential dependencies and adjusting to variable-length sequences, as demonstrated by the comparison with benchmarks. The work's aim includes a detailed working of training settings, neural network architectures, and sentiment analysis implications. The study explores the delicate and sensitive emotional reactions, basically when it comes to conversations on earthquakes on social media. Even though the work makes a substantial contribution to sentiment analysis techniques, this is important to recognize the method's advantages and also disadvantages. The results can be used for sentiment analysis for a larger range of areas that merely sentiments associated with earthquakes, explaining the work's exceptional generalizability. The study's generalizability is increased by the application of deep learning techniques to a huge range of situations and datasets, rendering it a priceless resource for anyone looking into sentiment analysis in many different contexts. As we look ahead to the research on "Utilizing Deep Learning Techniques for Sentiment Analysis," some opportunities for improvement and growth become clear. To overcome the constraints—which include the requirement for noticeably large amounts of labeled data and the possibility of data bias—it may be necessary to investigate methods for reducing bias in training datasets as well as semi-supervised or unsupervised learning strategies that can decrease the dependence on large amounts of labeled data. Exploring deep learning models' interpretability may lead to further improvements. It would be more useful to develop techniques for interpreting and explaining the sentiment analysis model's judgments, particularly in delicate applications like public discourse and crisis management. Refine the model's performance in real-world circumstances by evaluating it with user feedback and domain expertise.

8 Acknowledgement

I would like to thank my professor, Mr. Hicham Rifai, for his invaluable support throughout the semester regarding the project. Thank you for your valuable suggestions in writing the research paper.

References

Ahmed, A. and Sargent, J. (2014). Analysis of post-crisis twitter communication: a study of the iquique, chile earthquake, ACIS.

Alanazi, S. A., Khaliq, A., Ahmad, F., Alshammari, N., Hussain, I., Zia, M. A., Alruwaili, M., Rayan, A., Alsayat, A. and Afsar, S. (2022). Public's mental health monitoring via sentimental analysis of financial text using machine learning techniques, *International Journal of Environmental Research and Public Health* **19**(15): 9695.

Bianco, S., Cadene, R., Celona, L. and Napoletano, P. (2018). Benchmark analysis of representative deep neural network architectures, *IEEE access* **6**: 64270–64277.

Chandrasekaran, G., Antoanela, N., Andrei, G., Monica, C. and Hemanth, J. (2022). Visual sentiment analysis using deep learning models with social media data, *Applied Sciences* **12**(3): 1030.

Crooks, A., Croitoru, A., Stefanidis, A. and Radzikowski, J. (2013). # earthquake: Twitter as a distributed sensor system, *Transactions in GIS* **17**(1): 124–147.

Demirci, G. M., Keskin, Ş. R. and Doğan, G. (2019). Sentiment analysis in turkish with deep learning, *2019 IEEE international conference on big data (big data)*, IEEE, pp. 2215–2221.

Dong, Y., Fu, Y., Wang, L., Chen, Y., Dong, Y. and Li, J. (2020). A sentiment analysis method of capsule network based on bilstm, *IEEE access* **8**: 37014–37020.

Earle, P. (2010). Earthquake twitter, *Nature geoscience* **3**(4): 221–222.

Earle, P. S., Bowden, D. and Guy, M. (2011). Twitter earthquake detection: earthquake monitoring in a social world, *Annals of geophysics* **54**(6): 708–715.

Etaiwi, W., Suleiman, D. and Awajan, A. (2021). Deep learning based techniques for sentiment analysis: A survey, *Informatica* **45**(7).

Hasan, A., Moin, S., Karim, A. and Shamshirband, S. (2018). Machine learning-based sentiment analysis for twitter accounts, *Mathematical and computational applications* **23**(1): 11.

Jan, B., Farman, H., Khan, M., Imran, M., Islam, I. U., Ahmad, A., Ali, S. and Jeon, G. (2019). Deep learning in big data analytics: a comparative study, *Computers & Electrical Engineering* **75**: 275–287.

Kardakis, S., Perikos, I., Grivokostopoulou, F. and Hatzilygeroudis, I. (2021). Examining attention mechanisms in deep learning models for sentiment analysis, *Applied Sciences* **11**(9): 3883.

Lavanya, P. and Sasikala, E. (2021). Deep learning techniques on text classification using natural language processing (nlp) in social healthcare network: A comprehensive survey, *2021 3rd international conference on signal processing and communication (ICPSC)*, IEEE, pp. 603–609.

Li, J. and Rao, H. R. (2010). Twitter as a rapid response news service: An exploration in the context of the 2008 china earthquake, *The Electronic Journal of Information Systems in Developing Countries* **42**(1): 1–22.

Mateus, B. C., Mendes, M., Farinha, J. T., Assis, R. and Cardoso, A. M. (2021). Comparing lstm and gru models to predict the condition of a pulp paper press, *Energies* **14**(21): 6958.

Naqvi, U., Majid, A. and Abbas, S. A. (2021). Utsa: Urdu text sentiment analysis using deep learning methods, *IEEE Access* **9**: 114085–114094.

Nezhad, Z. B. and Deihimi, M. A. (2019). A combined deep learning model for persian sentiment analysis, *IIUM Engineering Journal* **20**(1): 129–139.

Oh, O., Kwon, K. H. and Rao, H. R. (2010). An exploration of social media in extreme events: Rumor theory and twitter during the haiti earthquake 2010.

Pandya, V., Somthankar, A., Shrivastava, S. S. and Patil, M. (2021). Twitter sentiment analysis using machine learning and deep learning techniques, *2021 2nd International Conference on Communication, Computing and Industry 4.0 (C2I4)*, IEEE, pp. 1–5.

Pathak, A. R., Agarwal, B., Pandey, M. and Rautaray, S. (2020a). Application of deep learning approaches for sentiment analysis, *Deep learning-based approaches for sentiment analysis* pp. 1–31.

Pathak, A. R., Agarwal, B., Pandey, M. and Rautaray, S. (2020b). Application of deep learning approaches for sentiment analysis, *Deep learning-based approaches for sentiment analysis* pp. 1–31.

Qaiser, S., Yusoff, N., Remli, M. and Adli, H. K. (2021). A comparison of machine learning techniques for sentiment analysis, *Turkish Journal of Computer and Mathematics Education* .

Radianti, J., Hiltz, S. R. and Labaka, L. (2016). An overview of public concerns during the recovery period after a major earthquake: Nepal twitter analysis, *2016 49th Hawaii international conference on system sciences (HICSS)*, IEEE, pp. 136–145.

Ramadhani, A. M. and Goo, H. S. (2017). Twitter sentiment analysis using deep learning methods, *2017 7th International annual engineering seminar (InAES)*, IEEE, pp. 1–4.

Rawat, R., Mahor, V., Chirgaiya, S., Shaw, R. N. and Ghosh, A. (2021). Sentiment analysis at online social network for cyber-malicious post reviews using machine learning techniques, *Computationally intelligent systems and their applications* pp. 113–130.

Robertson, B. W., Johnson, M., Murthy, D., Smith, W. R. and Stephens, K. K. (2019). Using a combination of human insights and 'deep learning'for real-time disaster communication, *Progress in Disaster Science* **2**: 100030.

Toriumi, F., Sakaki, T., Shinoda, K., Kazama, K., Kurihara, S. and Noda, I. (2013). Information sharing on twitter during the 2011 catastrophic earthquake, *Proceedings of the 22nd International Conference on World Wide Web*, pp. 1025–1028.

Wang, Y. (2017). A new concept using lstm neural networks for dynamic system identification, *2017 American control conference (ACC)*, IEEE, pp. 5324–5329.

Yadav, A. and Vishwakarma, D. K. (2020). Sentiment analysis using deep learning architectures: a review, *Artificial Intelligence Review* **53**(6): 4335–4385.

Yue, L., Chen, W., Li, X., Zuo, W. and Yin, M. (2019). A survey of sentiment analysis in social media, *Knowledge and Information Systems* **60**: 617–663.