Semantic Analysis of Sentiments through Web-Mined Twitter Corpus

Satish Chandra[a], Mahendra Kumar Gourisaria[a], Harshvardhan GM[a], Siddharth Swarup Rautaray[a], Manjusha Pandey[a] and Sachi Nandan Mohanty[b]

[a] *School of Computer Engineering, KIIT Deemed to be University, Bhubaneswar-751024, Odisha, India*
[b] *Dept of Computer Science & Engineering, ICFAITech, ICFAI Foundation for Higher Education, Hyderabad-500082, India*

Abstract

A huge amount of textual data is generated due to the boom of microblogging. Microblogging sites such as Facebook, Twitter and Google+ are used by millions of people to express their views and emotions on different subjects. In this paper, we discuss sentiment analysis on a Twitter dataset having various tweets from different users. Sentiment analysis is useful for gaining the opinion of people using large volumes of text data where texts are highly unstructured and heterogeneous. In this paper, different classification techniques like Support Vector Machine, Logistic Regression, Logistic Regression with Stochastic Gradient Descent optimizer, Decision Tree Classification, Naive Bayes, Bidirectional LSTM and Random Forest Classification have been applied to analyze the sentiment of people, i.e., whether their tweets are positive or negative. The corpus has been analyzed by plotting descriptive insights such as the word cloud and frequency of positive and negative tweets. The best classifier was selected by comparing the results of accuracy, recall, precision, F1 score, AUC score and ROC curve.

Keywords

Sentiment Analysis, Twitter, Natural Language Processing, Word2Vec, Support Vector Machine, Logistic Regression, Random Forest.

1. Introduction

With the universality of microblogging and social networking sites, Twitter, with 319 million monthly users has now become a valuable resource for several individuals and organizations for posting blogs and expressing their views and opinions on different subjects like politics, sports, movies, etc. [1]. Stimulated by the growth of social media, many companies and media organizations are trying to mine Twitter to observe people's views to understand what they feel and think about their products [2]. As a result, sentiment analysis on Twitter is an effective way of reckoning public opinion. Sentiment analysis provides the potential of observing numerous social networking sites in real-time.

Twitter has a limitation of 140 characters [3] in each tweet, which causes individuals to use phrases in their tweets. Sentiment Analysis automatically detects whether a text section contains emotions or opinioned content. It also determines the polarity of the text. Generally, the dataset consists of a group of tweets where each tweet is interpreted with a sentiment label. Commonly sentiments are labeled positive, negative or neutral. However, some datasets have mixed or irrelevant tags too, which ranges from -5 to 5 and depicting negative to positive polarity [4]. Twitter sentiment analysis is helpful to understand public temperament about different social or cultural events and forecasting the inconsistency within the stock exchange [5].

ISIC'21: International Semantic Intelligence Conference, February 25–27, 2021, Delhi, India
EMAIL: schandra1.sc@gmail.com (S. Chandra); mkgourisaria2010@gmail.com (M. K. Gourisaria); harrshvardhan@gmail.com (H. GM); siddharthfcs@kiit.ac.in (S.S. Rautaray); manjushafcs@kiit.ac.in (M. Pandey); sachinandan09@gmail.com (S.N. Mohanty)
ORCID: 0000-0002-6881-2668 (S. Chandra); 0000-0002-1785-8586 (M. K. Gourisaria);0000-0003-3592-2931(H. GM.); 0000-0002-3864-2127 (S.S. Rautaray); 0000-0002-6077-5794 (M. Pandey); 0000-0002-4939-0797 (S.N. Mohanty)

Sentiment analysis on Twitter is a sort of challenge due to its short length. The unstructured and heterogeneous data compelled us to apply the preprocessing step before feature extraction [6]. The various preprocessing steps include URLs removal, replacing negation, stopwords removal, removing numbers and expanding acronyms. The preprocessing has been done with the help of the Natural Language Tool Kit (NLTK). Then feature extraction is of two phases. First, the normal text was formed by eliminating the Twitter-specific features and then feature extraction was accomplished to extract more features [1].

This research paper is organized into different segments as follows. Section 2 briefs about related works of sentiment analysis. In Section 3 we talk about the methodology and materials which explains the data exploration, data preprocessing and feature extraction. We have also described the different classification algorithms used in the implementation namely Support Vector Machine, Logistic Regression, Logistic Regression - Stochastic Gradient Descent, Decision Tree, Naïve Bayes, Bidirectional Long Short-Term Memory (BiLSTM) and Random Forest. In Section 4 we show the results, analyses and comparison of models. Section 5 comprises the conclusion and future work.

2. Related works

With the advancement of Natural Language Processing (NLP), research on Sentiment Analysis ranges from document-level classification [7] to words and phrase-level classification [8]. The method to retrieve semantic information from a large corpus was presented by Hatzivassiloglou and McKeown. This method separates domain-dependent details and conforms to a novel domain when the corpus is substituted. Their model focuses on adjectives, intending to identify near-synonyms and antonyms from their model.

For increasing the efficiency and accuracy of the model [9] used the ensemble framework for sentiment analysis. They utilized movies reviews and multi domain datasets extracted from Amazon product reviews which includes reviews of Books, Electronics, DVD and Kitchen. They succeeded in framing the ensemble by combining various classification techniques and feature sets. They used two types of feature sets: word-relations and part-of-speech information and three types of classifiers like maximum entropy, Support Vector Machines and Naïve Bayes to form the ensemble framework. Weighted combination, fixed combination and meta-classifier ensemble techniques were used for sentiment analysis and better accuracy was attained [9]. People on social networking sites give their opinion about anything and everything. It was a challenge to recognize all types of data for training. Therefore, [2] proposed a model to study the sentiment from the hash tagged (HASH) data set, iSieve data set and the emoticon (EMOT) dataset. The authors trained their model on a variety of feature extraction techniques like lexicon features, part-of-speech (POS) features, n-gram features and microblogging features. They concluded that in the microblogging domain, the POS feature may not be useful and the benefits of the Emoticon dataset are also lessened when microblogging features are included [2].

The authors from the paper [10] discussed about social network analysis and Twitter being a rich source for sentiment analysis and proposed a model to implement Twitter sentiment analysis by fetching the data from Twitter APIs. Their analysis is based on different queries of job opportunities. The dataset has positive, negative, and neutral labels. They noted that the neutral sentiments are high in comparison to positive or negative which shows that there is a need to improve Twitter sentiment analysis [10]. Twitter has become increasingly popular in the field of politics. A real-time sentiment analyzer towards the incumbent of Ex. president Barack Obama and the nine other challengers have been designed by [11]. They used IBM's InfoSphere Streams platform (IBM, 2012) for speed and accuracy and pipelining real-time data. Using the Twitter "firehouse" they constructed logical keyword combinations to recover relatable tweets about candidates and events. They achieved an accuracy of 59% [11].

Some researchers have tried to determine the public point of view on different subjects like politics, movies, news, etc. from the Twitter posts [12]. The authors of the paper [13] used IMDB, a popular Internet database containing movie information and Blippr, a

social networking site where reviews are in the form of 'Iblips'. Their analysis gave the F-score as high as 0.9 using SVM and demonstrated domain adaptation as a useful technique for sentiment analysis. They introduced a new feature reduction technique, Relative Information Index (RII), which combines with another popular technique 'thresholding' to form a good feature reduction technique that not only reduces the features but also improves the F-score [13]. The importance of sentiment analysis has increased so much that it has been in use in various industries, such as hotel management. In this regard, [14] classified the public reviews of a hotel into positive and negative. They collected 800 reviews from TripAdvisor and performed the preprocessing step by NLTK in Python. They used various classifiers like Logistic Regression, Random Forest, Stochastic Gradient Descent Classifier, Naïve Bayes and Support Vector Machine. Their analysis was that Naïve Bays classifier was best among them but Stochastic Gradient classifier also worked well. The analysis was based on the results of accuracy, recall, precision and F1-score [14].

Table 1

Tabular presentation of the related work

Authors	Year	Dataset used	Models implemented	Observation/ Results
[9]	2011	Movies review, Multi domain dataset from extracted from Amazon which includes reviews of Books, DVD, Electronics and Kitchen.	They used two types of feature sets: word-relations and part-of-speech. Maximum entropy, Support Vector Machines and Naïve Bayes. Weighted combination, fixed combination and meta-classifier ensemble techniques were also used.	They observed that ensemble technique was very much efficient in obtaining the accurate results.
[2]	2011	Hash tagged (HASH) data set, iSieve data set and the Emoticon (EMOT) dataset	The model was trained on a variety of feature extraction techniques like lexicon features, n-gram features, part-of-speech (POS) features and microblogging features.	The best result was obtained from n-gram features along with lexicon features. POS features may not be useful in microblogging domain
[10]	2019	The data was obtained from Twitter API for different job opportunities queries.	They used NLTK for find the different categories of the tweets like positive, weakly positive, strongly positive, neutral, negative, strongly negative, weakly negative.	The concluded that the neutral tweets are significantly high in most of the queries. Thereby showing the improvement in Sentiment Analysis.
[11]	2012	The data was obtained from Twitter API during the US presidential election in 2012.	Designed a real-time sentiment analyzer towards the incumbent of Ex. president Barack Obama and the nine other challengers. They used IBM's InfoSphere Streams platform (IBM, 2012) for speed and accuracy and pipelining real-time data. Using the Twitter "firehouse" they constructed logical keyword combinations to recover relatable tweets about candidates and events.	They achieved an accuracy of 59%.

[13]	2011	IMDB, a popular Internet database containing movie information and Blippr, a social networking site where reviews are in the form of 'Iblips'.	They used SVM and introduced a new feature reduction technique, Relative Information Index (RII), which combines with another popular technique 'thresholding' to form a good feature reduction technique that not only reduces the features but also improves the F-score	Their analysis gave the F-score as high as 0.9 using SVM and demonstrated domain adaptation as a useful technique for sentiment analysis.
[14]	2018	They classified the public reviews of a hotel into positive and negative by collecting 800 reviews from TripAdvisor	They used various classifiers like Logistic Regression, Random Forest, Stochastic Gradient Descent Classifier, Naïve Bayes and Support Vector Machine.	Their analysed that Naïve Bays classifier was best among them but Stochastic Gradient classifier also worked well. The analysis was based on the results of accuracy, recall, precision and F1-score.

3. Materials and methods

The study of computer algorithms that improves automatically by learning from itself is known as machine learning. The data and output are fed into the machine learning model and the machine creates its programming logic to predict the result. The dataset is split into two halves i.e., training part, which contains input feature vectors and their labels, and the testing part. A classification model with the help of a specific algorithm is developed using the training part to observe a pattern. The testing part is used to obtain the accuracy of the model, which tells whether a model is a good fit, underfit or overfit.

3.1. Data exploration

The dataset used in this work was taken from UCI/Kaggle [15] in csv (comma separated values) which contains 1.6 million tweets. Preprocessing the data was done which includes tokenization, stemming, stopword removal to clean the text. A feature vector was created using relevant features. Data mining classification algorithms such as Decision Tree, Logistic Regression, Random Forest, SVM, Naive Bayes and LR-SGD classifiers were used to gather the accuracy by classifying the tweets into positive or negative tweets. Fig. 1 shows the algorithm adopted for sentiment analysis.

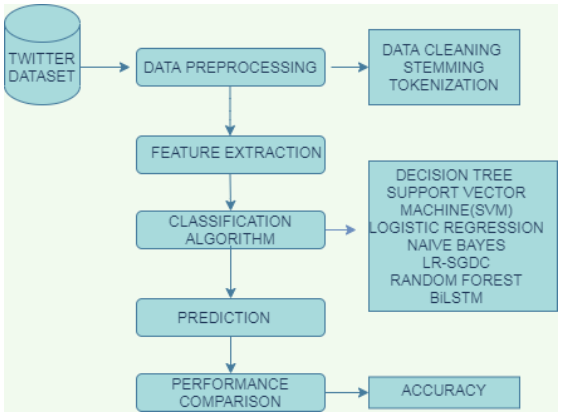

Figure 1: Workflow for Twitter sentiment analysis

Exploring the data has a key role in machine learning as it helps us to visualize the types and statistics of data [16]. Here, the dataset consists of 0.8M positive and 0.8M negative tweets shown in Fig. 2 (a). As it is text data, the word cloud can also be visualized, as shown in Fig. 2 (b).

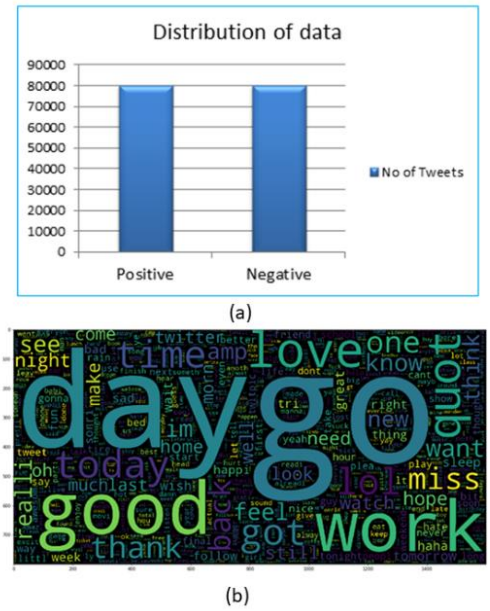

Figure 2: (a) Statistics of positive and negative dataset (b) Word cloud of the dataset

3.2. Data preprocessing

As the Twitter datasets are composed of unstructured, heterogeneous, ill-formed words, irregular grammar and non-dictionary terms, the tweets were cleaned by various NLTK methods before feature extraction [1]. Preprocessing steps are [12] -

- Eliminating all non-English characters and non-ASCII from the text.
- Removal of all URL links as they do not provide any information about the sentiment.
- Numbers are removed as they are not useful in finding sentiment.
- Stop words are the most frequent words in a language, such as "as", "an", "about", "any" etc. There are many stopwords in English literature. These stopwords do not play any role in finding the sentiments so they are removed from the dataset.
- Stopwords also contain "not", but are not removed from the tweets as they are crucial in analyzing negative reviews.
- Stemming is the process to bring back the words into their original form such as "loved" becomes "love", "worst" becomes "bad" and so on.

3.3. Feature extraction

In feature extraction, the vector space model is used for document representation. A vector is created whose dimension is equal to the size of English vocabulary and each element is initially initialized to 0. If a text data features that vocab word, one '1' will be put in that dimension, as shown in Eqn. 1, Eqn. 2 and Eqn. 3. Every time, a text that features the vocab word is encountered, the count will be increased, leaving 0's everywhere for the words which were not found even once. The 2nd Edition of the Oxford dictionary contains 171,476 words [17] in current use. So, if a vector is made with all these words the model will be of high variance and here feature selection comes into account. For proper weighting and feature extraction, the count vectorizer method was used which keeps track of the frequent terms as well as rare words. The vector space model improves the accuracy. The feature extraction method is used for dimensionality reduction by removing the non-informative words and rare words. Bag of Words model is created which contains the most frequent words from the feature vector to improve the accuracy [1].

$$Love = [0,0,0,1,0,0,0 \dots \dots \dots \dots .0] \quad (1)$$

$$Good = [0,0,0,0,2,0,0 \dots \dots \dots \dots \dots .0] \quad (2)$$

$$Day = [0,0,0,0,0,5,0 \dots \dots \dots \dots \dots \dots 0] \quad (3)$$

Other than Bag-of words model Tokenization was also used for Bidirectional Long Short-Term Memory, in which raw texts are broken up into unique texts i.e., tokens. Each of the tokens has its unique token id's. In tokenisation, a vector is created with a size equivalent to the number of unique words in the corpora. A sequence of tokens is created and they are represented as a vector as shown in Eqn. 4 and Eqn. 5. As each of the tweets has a different length so its token represented sequence has also a different length which makes it difficult to feed into the Deep Learning algorithms as it requires sequences of the same length [18] . To counter this problem, padding and truncating steps come into account where the length of the padded sequence is defined. If the length of the tokenised sequence is larger than the padded sequence then the tokens of the sequence after the length of the tokenised sequence would be truncated, i.e., they are removed. If the length of the tokenised

sequence is smaller than the padded sequence then the tokens of the sequence after the length of the tokenized sequence would be padded with "0". If the length of the padded sequence is chosen to be 6 then Eqn. 4 will be truncated as shown in Eqn. 6 and Eqn. 5 will be padded as shown in Eqn. 7.

$$\text{What consumes your mind controls your life} = [32,13,21,122,781,45,23] \quad (4)$$

$$practice\ makes\ a\ man\ perfect = [53,321,32,48,44] \quad (5)$$

$$\text{What consumes your mind controls your life} = [32,13,21,122,781,45] \quad (6)$$

$$practice\ makes\ a\ man\ perfect = [53,321,32,48,44,0] \quad (7)$$

3.4. Classification Algorithms

Classification algorithms are the most important part of supervised learning in machine learning. The classification algorithm is used to indicate the class of the data. In this paper, classification algorithms play a crucial role in labeling the tweets positive or negative.

3.4.1. Bidirectional Long Short-Term Memory (BiLSTM)

A traditional neural network can't remember the previous inputs, for predicting the next word previous information is a must. Recurrent Neural Network (RNN) has the potential of remembering everything from the past as they have the loop and hidden layer in them. The loops in RNN allows the network to persist information. Recurrent neural network translates the independent activations to dependent activations by furnishing equal biases and weights to complete layers, thus the complexity of increasing the parameters is reduced and the result of one layer is the input to the following hidden layers [19]. Long Short-Term Memory (LSTM) is a special form of Recurrent Neural Network (RNN) which has the potential to learn long-term dependencies. LSTMs are accomplished to abstain from the long dependencies problem. In LSTM the hidden layer of RNN is restored by the Long Short-Term Memory cell. The LSTM memory cell can be achieved by the Eqn. 8-12.

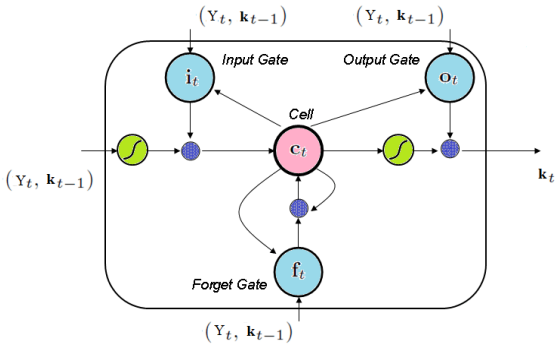

Figure 3: Memory cell of LSTM

$$i_t = \sigma(W_{yi}y_t + W_{ki}k_{t-1} + W_{ci}c_{t-1} + b_i) \quad (8)$$

$$o_t = \sigma(W_{yo}y_t + W_{ko}k_{t-1} + W_{co}c_{t-1} + b_o) \quad (9)$$

$$f_t = \sigma(W_{yf}y_t + W_{kf}k_{t-1} + W_{cf}c_{t-1} + b_f) \quad (10)$$

$$c_t = f_t c_{t-1} + i_t tanh(W_{yc}y_t + W_{kc}k_{t-1} + b_c) \quad (11)$$

$$k_t = o_t \tanh(c_t) \quad (12)$$

Where σ represents a logistic sigmoid function, c, o, i and f represent cell vectors, output, input and forget gate. These have the same dimension as the hidden vector k [19].

Bidirectional Long Short-Term Memory (BiLSTM**)** is an extension of LSTM, which can be designed by putting two independent LSTM. The structure permits the neural network to have both forward and backward information at every time step. This will run the data in two ways, one from future to past and one from past to future so by this method the model will be able to preserve information from both the

future and past. Fig. 4 shows the Bidirectional LSTM [20].

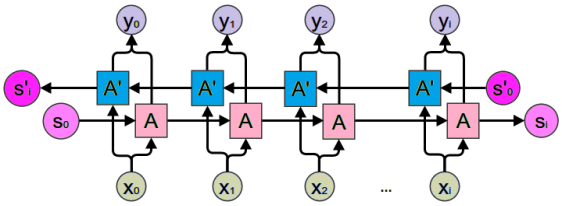

Figure 4: A Bidirectional LSTM Network

3.4.2. Logistic regression

Logistic regression is an example of a linear classifier that is used to classify the class of data. Logistic regression determines the link between the independent and dependent variables by estimating probabilities [16]. It returns the probability by transforming the output with the help of the logistic sigmoid function. Fig. 5 shows the linear regression graph and its equation is given by Eqn. 13 as,

$$Y = B_0 + B_1 X \tag{13}$$

The equation of sigmoid function [22] is,

$$P = \frac{1}{1 + e^{-y}} \tag{14}$$

Now, applying Eqn. 14 to Eqn. 13 and solving for y to get Eqn. 15 i.e., logistic regression equation

$$\ln\left(\frac{P}{1-P}\right) = B_0 + B_1 X \tag{15}$$

The graph is now converted into a logistic regression graph shown in Fig. 6.

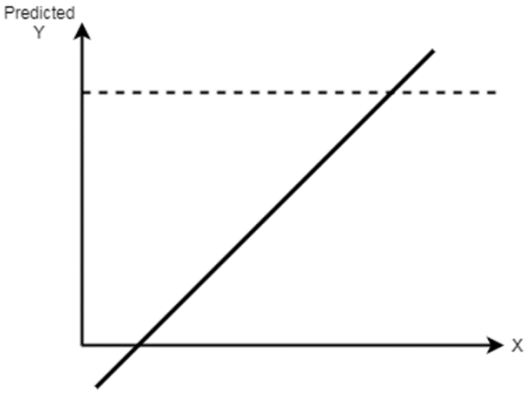

Figure 5: Linear regression graph

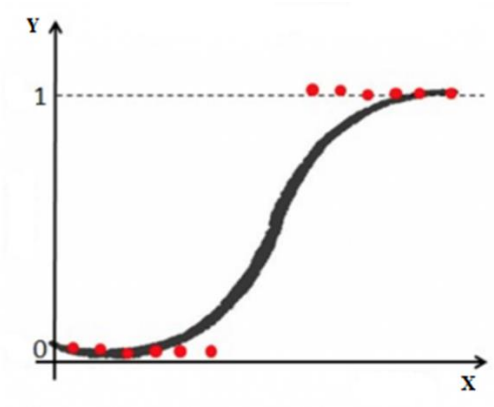

Figure 6: Logistic Regression curve

3.4.3. Logistic Regression-Stochastic Gradient Descent Classifier

Logistic Regression-Stochastic Gradient Descent (LR-SGD) is a type of linear model, known as Incremental Gradient Descent [14]. Logistic Regression-Stochastic Gradient Descent (LR-SGD) classifier is an effective way to selective learning of linear classifiers under different loss functions and penalties such as Logistic Regression and Support Vector Machines. The 'log' loss function is used to optimize Logistic Regression while the 'hinge' loss function is used for optimizing the Support Vector Machine. LR-SGD Classifier has recently gained much significance in the field of large-scale learning although it has been around in the machine learning association for a long time [21]. The sparse and large-scale machine learning problems, which can be encountered in sentiment analysis, often make use of the LR-SGD classifier and this fact motivated us to use the LR-SGD classifier in our problem with 1.6M tweets [22]. One of the strengths of the LR-SGD classifier is the hyperparameter tuning which can be used to solve error functions also called the cost function.

3.4.4. Support Vector Machine

The Support Vector Machine can be regarded as a linear model for regression and classification tasks [23]. The Support Vector Machine finds the optimal separable hyperplane to separate the tweets into two parts [24]. It is applied to noisy data. The hyperplane line separates the tweets in a very efficient way shown in Fig. 7. Support Vectors are the

locations which are quite close to the line from both the classes. The distance between them is often called a margin [25]. The Support Vector Machine is easier to implement and scales well for high dimensional data. It is implemented with kernels that transform non-separable problems into separable problems by adding more dimensions to it. The most commonly used kernel is the Radial Basis Function (RBF) kernel. Mathematically, it can be defined by Eqn. 16,

$$P(y, y_i) = e^{(-gamma*sum(y-y_i{}^2))} \quad (16)$$

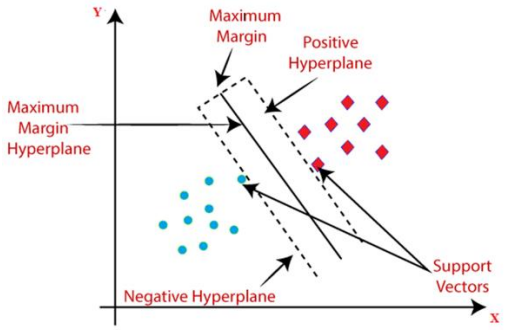

Figure 7: SVM classifier graph showing hyperplane

3.4.5. Naïve Bayes Classifier

Naïve Bayes [26] is the most common supervised machine learning technique for classification. It is also known as the probabilistic classification technique as it is based on probability [27]. It is completely dependent on the famous probability theorem i.e., Bayes' theorem. Bayes' theorem is correlated to conditional probability. It finds the probability of an occurring event when the probability of another occurred event is already given [27]. Mathematically, it can be stated by Eqn. 17,

$$P\left(\frac{M}{N}\right) = \frac{P(M)P(^N/_M)}{P(N)} \quad (17)$$

Where, $P\left(\frac{M}{N}\right)$ refers to posterior i.e. probability of M when N is given, $P(^N/_M)$ represents likelihood i.e., probability of N when M true, P(M) is the prior i.e., probability of M and P(N) represents marginalization i.e. probability of N [28]. After implementing the model in classifier the equation is [16], [29] given by Eqn. 18 as,

$$M = argmax_M P(M) \prod_{i=1}^{n} P(\frac{N_i}{M}) \quad (18)$$

3.4.6. Decision Tree Classifier

A feasible approach to the multistage decision is to use the Decision Tree classifier [30]. In the multistage approach complex decisions are broken up into several simple decisions to obtain the desired solution. A complete multistage recognition has been reviewed by [31]. It is used where data is regularly split. Decision Tree can be applied for both - regression models to predict the continuous value and classification models for predicting probability. As our model is a binary classifier having positive and negative labels, the Decision Tree classifier has been implemented [32]. It is robust, easy and simple to implement and not sensitive to irrelevant features [33]. Fig. 8 (a) shows how the dataset was split into different categories using the Decision Tree classifier and (b) demonstrates a general Decision Tree.

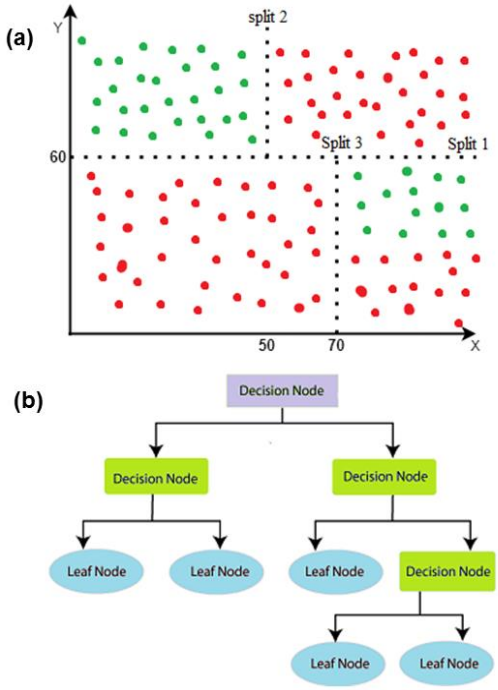

Figure 8: (a) Portioning of a two-dimensional feature space (b) Overview of a Decision Tree

3.4.7. Random Forest Classifier

The Random Forest classifier is a supervised ML technique and a very popular classifier. Just like the Decision Tree, it can also be implemented on both classification and regression models. It is an ensemble learning method of classification that builds a set of multiple decision trees from the training data and outputs mode of class [34]. It is used in applications like search engines, image classification, etc. It constructs a decision tree from each sample and gives the output. The best solution is selected by voting. It is easier to implement, fast and scalable but it easily overfits the data [34]. Fig. 9 shows the complete sketch of the Random Forest classifier.

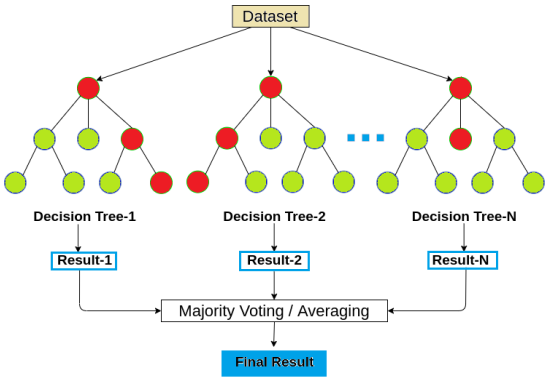

Figure 9: Overview of Random Forest classifier

4. Implementation and result

The dataset was collected from Kaggle. Implementation was done on Python and NLTK was used for cleaning and training the model. The various classifiers used are Logistic Regression, Naïve Bayes, Support Vector Machine, Random Forest, LR-SGD classifier, Bidirectional Long Short-Term Memory and Decision Tree. The dataset consists of 1.6 million out of which 1,280,000 were used for training and 320,000 for testing [15].

Evaluating the models is very important for observing the performance and correctness of the different models on the test data and finding the best among them. The performance of a classifier can be described by the confusion matrix on a set of data for which true values are known. With the help of the confusion matrix, different evaluating metrics such as accuracy, recall, precision, F1-score and AUC score have been evaluated to validate and verify the quality of the results [35], [36]. The confusion matrices for various classifiers have been shown in Fig. 10, Fig. 11, Fig. 12 and Fig. 13. Table 2 compares the different classification models based on these evaluating metrics. Fig. 14 graphically depicts the performance of the different classifiers concerning the accuracy, recall, precision, F1-score and AUC score.

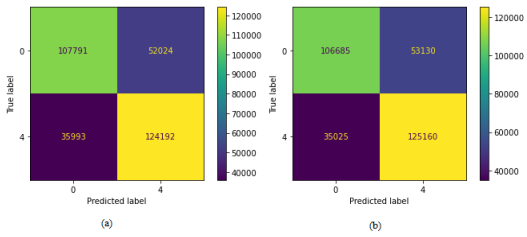

Figure 10: Confusion matrix of (a) Logistic Regression (b) Support Vector Machine

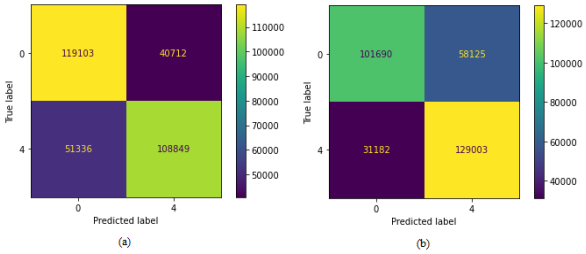

Figure 11: Confusion matrix of (a) Naïve Bayes (b) LR-SGD classifier

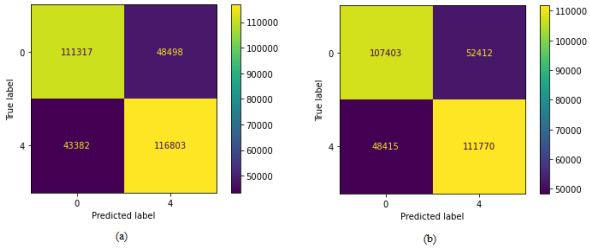

Figure 12: Confusion matrix of (a) Random Forest (b) Decision Tree classifier

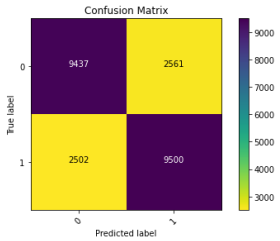

Figure 13: Confusion matrix of Bidirectional Long Short-Term Memory

Table 2
Performance measure of various classifiers

	Accuracy	Recall	Precision	F-1 Score	AUC Score
Bidirectional LSTM	0.7890	0.7889	0.7891	0.7889	0.78904
Logistic Regression	0.7249	0.7249	0.7272	0.7242	0.72489
Naïve Bayes	0.7124	0.7124	0.7133	0.7121	0.71239
LR-SGDC	0.7209	0.7208	0.7274	0.7189	0.72082
SVM	0.7245	0.7244	0.7274	0.7236	0.72445
Decision Tree	0.6849	0.6849	0.685	0.6849	0.6849
Random Forest	0.7129	0.7129	0.7131	0.7128	0.71286

Accuracy: It is the percentage of tweets that have been classified correctly by the model. The accuracy of the model can be calculated using Eqn. 19.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (19)$$

Precision: It is the ratio of actual positive tweets to predicted positive tweets. The precision of the model can be calculated using Eqn. 20.

$$Precision = \frac{TP}{TP + FP} \quad (20)$$

Recall: It is the ratio of predicted positive tweets to total positive tweets. The recall of the model can be calculated using Eqn. 21.

$$Recall = \frac{TP}{TP + FN} \quad (21)$$

F1-score: F1-score can be defined as the harmonic mean of recall and precision. The F-measure of the model can be calculated using Eqn. 22.

$$F1\ score = 2 * \frac{P * R}{P + R} \quad (22)$$

Where, TP is the True Positive, TN refers to True Negative, FP is the False Positive, FN means False Negative, P refers to Precision and R is the Recall.

AUC score: AUC score can be calculated by finding the area under the ROC curve [11]. The AUC score of the model can be calculated using Eqn. 23.

$$AUC = \frac{SP - PE\,(NO + 1)/2}{PE * NO} \quad (23)$$

Where, SP is the Sum of positive observations, PE refers to Positive Examples and NO is the Negative Observations.

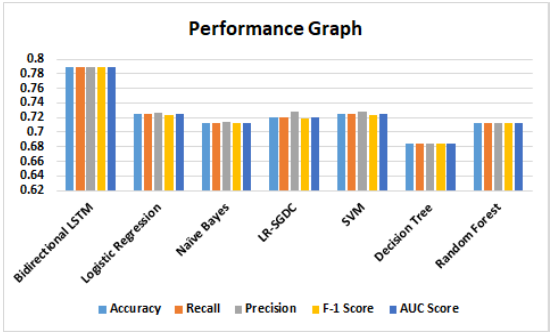

Figure 14: Performance graph of different classifiers

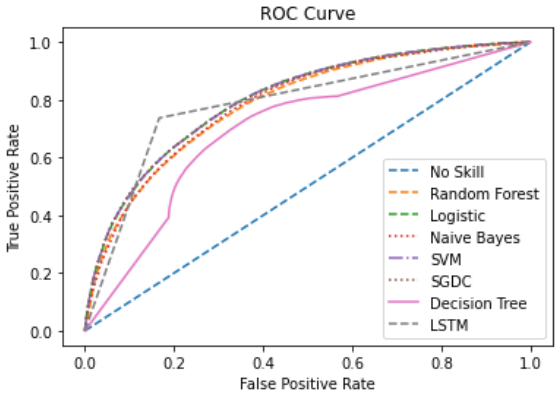

Figure 15: ROC Curve of various Classifiers

The Receiver Operating Characteristic curve (ROC) curve is a tool which predicts the probabilistic value of binary outcome [37]. The relationship between the sensitivity which is the true positive rate and the specificity which is the false positive rate is represented graphically by the ROC curve. It is a significant metric as it covers the whole spectrum between zero and one. The true positive rate is exactly equal to the false positive rate at 0.5, and this represents a random or no skilled classifier [38]. The AUC score can be calculated by finding the area under the ROC curve. The ROC curves for different classifiers have been plotted in Fig. 15.

With the help of the confusion matrix of the various classifiers showing the values of a true negative, true positive, false negative and false

positive, we have calculated precision, accuracy, F1-score, recall and roc-auc score as shown in Table 2. In this paper, we have compared various classifiers like Random Forest, Logistic regression, Support Vector Machine, Decision Tree, LR-SGDC and Naïve Bayes with the state-of-the-art approach Bi-LSTM. On observing the results of Table 2 it had been found that the Bidirectional LSTM was the best classifier with an accuracy of 78.90%, and decision tree came out as runner up with an accuracy of 72.49%, followed by Support Vector Machine and LR-SGDC Classifier with an accuracy of 72.45% and 72.09% respectively. Random Forest and Naïve Bayes also predicted well with an accuracy of 71.29% and 71.24%. it was also observed that decision tree classifier didn't came up to the expectation with just an accuracy of 68.49%.
On examining carefully, it can be observed that prediction of true positive class with respect to predicted positive class i.e., precision score of Bi-LSTM was also highest among all with a precision score of 78.91%. LR-SGDC and SVM classifier were the runner ups with a precision score of 72.74% for each, followed by Logistic Regression with 72.72% precision score. Naïve Bayes and Random Forest classifier also predicted the positive class well with a precision score of 71.33% and 71.31% respectively. The precision score of decision tree classifier was least with a score of 68.5%. Prediction of true positive class with respect to actual positive class i.e., recall score of Bi-LSTM was best with score of 78.89%, with Logistic Regression as the runner up with sore of 72.49% followed by SVM, LR-SGDC, Random Forest and Naïve Bayes with score of 72.44%, 72.08%, 71.29% and 71.24% respectively. Even here, Decision Tree was not as good with precision score of 68.49%. The F1-score and AUC score of Bi-LSTM was best of among all the classifiers. All these results of various classifiers can be visualized graphically as shown in the Fig. 14. Fig. 15 depicts the ROC curve of all the classifiers implemented in our experiments, which also shows that Bi-LSTM is the best classifier. The model can also be very useful for analyzing the tweets related to medical data [39], [40],[41], [42], [43], [44].

5. Conclusion

There are various methods of machine learning, symbolic and deep learning for the analysis of the tweets or reviews. But machine learning techniques are most common, efficient and simpler than others. In this paper, machine learning techniques were used for the analysis of tweets on a Twitter dataset. The tweets were cleaned in the preprocessing step by removing the stopwords, URL, numbers and various Twitter-specific features with the help of NLTK. To deal with the miss-spelling and non-informative words, feature extraction was done and a Bag of Words model was created with the most frequent words. The tweets were, then, classified into positive and negative by various classifiers like LR-SGD Classifier, Naïve Bayes, Random Forest, Logistic Regression, SVM, Bidirectional LSTM and Decision Tree. By observing the ROC curve and accuracy score, it was clear that Bidirectional LSTM is the best classifier with an accuracy of 78.90%. Hence, it was found that Bidirectional LSTM is very useful in finding sentiment analysis.

The model can be implemented in a website or Android applications for classifying the sentiments of people on different subjects. As the microblogging sites are blooming, sentiment analysis is very important for many organizations in implicating social intelligence and social media analytics.

The future of this research paper is to explore the data on a wider genre of different social networking sites and e-commencing sites where people do online shopping for many things like books, games, etc. Accuracy rates of these products can be found by sentiment analysis. It can also be implemented to build the human confidence model.

6. Conflict of interest

There is no conflict of interest.

7. Acknowledgement

I would like to express my heartiest gratitude to all the co-authors and special thanks to Prof. Mahendra Kumar Gourisaria and Mr. Harshvardhan GM who have been a constant source of knowledge, inspiration and support. I would equally thank my parents and friends who inspired me to remain focused and helped me to complete this research paper.

8. References

[1] Z. Jianqiang, G. Xiaolin, and Z. Xuejun, "Deep Convolution Neural Networks for Twitter Sentiment Analysis," *IEEE Access*, vol. 6, pp. 23253–23260, 2018, doi: 10.1109/ACCESS.2017.2776930.

[2] E. Kouloumpis, T. Wilson, and J. Moore, "Twitter sentiment analysis: The good the bad and the omg!. In Fifth International AAAI conference on weblogs and social media," in *Proceedings of the Fifth International AAAI Cinference on Weblogs and Social Media*, 2011, pp. 538–541.

[3] A. Hassan, A. Abbasi, and D. Zeng, "Twitter Sentiment Analysis: A Bootstrap Ensemble Framework," in *2013 International Conference on Social Computing*, Sep. 2013, pp. 357–364, doi: 10.1109/SocialCom.2013.56.

[4] M. Thelwall, K. Buckley, and G. Paltoglou, "Sentiment strength detection for the social web," *J. Am. Soc. Inf. Sci. Technol.*, vol. 63, no. 1, pp. 163–173, 2012, doi: 10.1002/asi.21662.

[5] A. Mittal and A. Goel, "Stock prediction using twitter sentiment analysis," 2012.

[6] Z. Jianqiang and G. Xiaolin, "Comparison research on text pre-processing methods on twitter sentiment analysis," *IEEE Access*, vol. 5, no. c, pp. 2870–2879, 2017, doi: 10.1109/ACCESS.2017.2672677.

[7] B. Pang and L. Lee, "Opinion mining and sentiment analysis," *Found. Trends Inf. Retr.*, vol. 2, no. 1–2, pp. 1–135, 2008, doi: 10.1561/1500000011.

[8] V. Hatzivassiloglou and K. R. McKeown, "Predicting the semantic orientation of adjectives," in *Proceedings of the eighth conference on European chapter of the Association for Computational Linguistics* -, 1997, pp. 174–181, doi: 10.3115/979617.979640.

[9] R. Xia, C. Zong, and S. Li, "Ensemble of feature sets and classification algorithms for sentiment classification," *Inf. Sci. (Ny).*, vol. 181, no. 6, pp. 1138–1152, 2011, doi: 10.1016/j.ins.2010.11.023.

[10] A. Baweja and P. Garg, "Sentimental Analysis of Twitter Data for Job Opportunities," *Int. Res. J. Eng. Technol.*, vol. 6, no. 11, pp. 2344–2350, 2019.

[11] H. Wang, D. Can, A. Kazemzadeh, F. Bar, and S. Narayanan, "A System for Real-time Twitter Sentiment Analysis of 2012 U.S. Presidential Election Cycle," *Proc. 50th Annu. Meet. Assoc. Comput. Linguist.*, no. July, pp. 115–120, 2012, doi: 10.1145/1935826.1935854.

[12] M. S. Neethu and R. Rajasree, "Sentiment analysis in twitter using machine learning techniques," *2013 4th Int. Conf. Comput. Commun. Netw. Technol. ICCCNT 2013*, 2013, doi: 10.1109/ICCCNT.2013.6726818.

[13] V. M. K. Peddinti and P. Chintalapoodi, "Domain adaptation in sentiment analysis of twitter," *AAAI Work. - Tech. Rep.*, vol. WS-11-05, pp. 44–49, 2011.

[14] N. Lokeswari and K. Amaravathi, "Comparative Study of Classification Algorithms in Sentiment Analysis," *Int. Res. J. Sci. Eng. Technol.*, vol. 4, no. 8, pp. 31–39, 2018.

[15] Kaggle.com, "Sentiment140 dataset with 1.6 million tweets," 2015. [Online]. Available: https://www.kaggle.com/kazanova/sentiment140.

[16] S. Das, R. Sharma, M. K. Gourisaria, S. S. Rautaray, and M. Pandey, "Heart disease detection using core machine learning and deep learning techniques: A comparative study," *Int. J. Emerg. Technol.*, vol. 11, no. 3, pp. 531–538, 2020.

[17] Wil, "How many words are in the English language?," *English Live*, 2018. [Online]. Available: https://wordcounter.io/blog/how-many-words-are-in-the-english-language/.

[18] Z. Jiang, L. Li, D. Huang, and L. Jin, "Training word embeddings for deep learning in biomedical text mining tasks," *Proc. - 2015 IEEE Int. Conf. Bioinforma. Biomed. BIBM 2015*, pp. 625–628, 2015, doi: 10.1109/BIBM.2015.7359756.

[19] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF Models for Sequence Tagging," 2015, [Online]. Available: http://arxiv.org/abs/1508.01991.

[20] A. Graves and J. Schmidhuber,

"Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Networks*, vol. 18, no. 5–6, pp. 602–610, 2005, doi: 10.1016/j.neunet.2005.06.042.

[21] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*. Cambridge University Press, 2014.

[22] M. Thenuwara and H. R. K. Nagahamulla, "Offline handwritten signature verification system using random forest classifier," in *17th International Conference on Advances in ICT for Emerging Regions, ICTer 2017 - Proceedings*, 2017, vol. 2018-Janua, pp. 191–196, doi: 10.1109/ICTER.2017.8257828.

[23] W. Yu, T. Liu, R. Valdez, M. Gwinn, and M. J. Khoury, "Application of support vector machine modeling for prediction of common diseases: The case of diabetes and pre-diabetes," *BMC Med. Inform. Decis. Mak.*, vol. 10, no. 1, 2010, doi: 10.1186/1472-6947-10-16.

[24] S. Nayak, M. Kumar Gourisaria, M. Pandey, and S. Swarup Rautaray, "Heart Disease Prediction Using Frequent Item Set Mining and Classification Technique," *Int. J. Inf. Eng. Electron. Bus.*, vol. 11, no. 6, pp. 9–15, 2019, doi: 10.5815/ijieeb.2019.06.02.

[25] S. Ghumbre, C. Patil, and A. Ghatol, "Heart Disease Diagnosis using Support Vector Machine," *Int. Conf. Comput. Sci. Inf. Technol.*, pp. 84–88, 2011.

[26] S. Nayak, M. K. Gourisaria, M. Pandey, and S. S. Rautaray, "Comparative Analysis of Heart Disease Classification Algorithms Using Big Data Analytical Tool," 2020, pp. 582–588.

[27] V. S and D. S, "Data Mining Classification Algorithms for Kidney Disease Prediction," *Int. J. Cybern. Informatics*, vol. 4, no. 4, pp. 13–25, 2015, doi: 10.5121/ijci.2015.4402.

[28] Wikipedia contributors. "Bayes Theorem". *Wikipedia, The Free Encyclopedia*. https://en.wikipedia.org/wiki/Bayes'_theorem, Last accessed 2020/8/28.

[29] H. GM, M. K. Gourisaria, M. Pandey, and S. S. Rautaray, "A comprehensive survey and analysis of generative models in machine learning," *Comput. Sci. Rev.*, vol. 38, 100285, Nov. 2020, doi: 10.1016/j.cosrev.2020.100285.

[30] S. Rasoul and L. David, "A Survey of Decision Tree Classifier Methodology," *IEEE Trans. Syst. Man. Cybern.*, vol. 21, no. 3, pp. 660–674, 1991.

[31] G. R. Dattatreya and L. N. Kanal, "Decision Trees in Pattern Recognition," in *Progress in pattern recognition 2*, 1985, pp. 189–239.

[32] S. Nayak, M. K. Gourisaria, M. Pandey, and S. S. Rautaray, "Prediction of Heart Disease by Mining Frequent Items and Classification Techniques," in *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, May 2019, pp. 607–611, doi: 10.1109/ICCS45141.2019.9065805.

[33] Wikipedia contributors. Decision tree learning. *Wikipedia, The Free Encyclopedia*,https://en.wikipedia.org/wiki/Decision_tree_learning, Last accessed 2020/8/30.

[34] A. Gupte, S. Joshi, P. Gadgul, and A. Kadam, "Comparative Study of Classification Algorithms used in Sentiment Analysis," *Int. J. Comput. Sci. Inf. Technol.*, vol. 5, no. 5, pp. 6261–6264, 2014.

[35] A. Giachanou and F. Crestani, "Like It or Not," *ACM Comput. Surv.*, vol. 49, no. 2, pp. 1–41, Nov. 2016, doi: 10.1145/2938640.

[36] G. Gautam and D. Yadav, "Sentiment analysis of twitter data using machine learning approaches and semantic analysis," in *2014 7th International Conference on Contemporary Computing, IC3 2014*, 2014, pp. 437–442, doi: 10.1109/IC3.2014.6897213.

[37] B. Jason, "How to Use ROC Curves and Precision-Recall Curves for Classification in Python," *Machine Learning Mastery*. pp. 1–48, 2018.

[38] A. H. Hossny, L. Mitchell, N. Lothian, and G. Osborne, "Feature selection methods for event detection in Twitter: a text mining approach," *Soc. Netw. Anal. Min.*, vol. 10, no. 1, 2020, doi: 10.1007/s13278-020-00658-3.

[39] S. Dey, M. K. Gourisaria, S. S. Rautray, and M. Pandey, "Segmentation of Nuclei in Microscopy Images Across Varied Experimental Systems," 2021,

pp. 87–95.

[40] R. Sharma, S. Das, M. K. Gourisaria, S. S. Rautaray, and M. Pandey, “A Model for Prediction of Paddy Crop Disease Using CNN,” 2020, pp. 533–543.

[41] M. K. Gourisaria, S. Das, R. Sharma, S. S. Rautaray, and M. Pandey, “A deep learning model for malaria disease detection and analysis using deep convolutional neural networks,” *Int. J. Emerg. Technol.*, vol. 11, no. 2, pp. 699–704, 2020.

[42] S. S. Rautaray, S. Dey, M. Pandey, and M. K. Gourisaria, “Nuclei segmentation in cell images using fully convolutional neural networks,” *Int. J. Emerg. Technol.*, vol. 11, no. 3, pp. 731–737, 2020.

[43] S. Sharma, M. K. Gourisaria, S. S. Rautray, M. Pandey, and S. S. Patra, “ECG Classification using Deep Convolutional Neural Networks and Data Analysis,” *Int. J. Adv. Trends Comput. Sci. Eng.*, no. 9, pp. 5788–5795, 2020.

[44] G. Jee, H. GM and M. K. Gourisaria, “Juxtaposing inference capabilities of deep neural models over posteroanterior chest radiographs facilitating COVID-19 detection,” *J. of Interdisciplinary Mathematics,* pp. 1-27, 2021,
doi: 10.1080/09720502.2020.1838061