IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Detecting Dengue/Flu Infections based on Tweets using LSTM and Word Embedding

Samina Amin[1], M. Irfan Uddin[1]*, M. Ali Zeb[1], Ala Abdulsalam Alarood[2], Marwan Mahmoud[3], Monagi H. Alkinani[4]

[1]Institute of Computing, Kohat University of Science and Technology, Kohat 2600, Pakistan. (kustsameena@gmail.com, irfanuddin@kust.edu.pk, alizeb@kust.edu.pk)
[2]College of Computer Science and Engineering, University of Jeddah, 21959 Jeddah, Saudi Arabia. (aasoleman@uj.edu.sa)
[3]Faculty of Applied Studies, King Abdulaziz University, 21959 Jeddah, Saudi Arabia (mmamahmoud@kau.edu.sa)
[4]College of Computer Sciences and Engineering, Department of Computer science and Artificial Intelligence, University of Jeddah, 21959 Jeddah, Saudi Arabia. (malkinani@uj.edu.sa)

Corresponding author: M. Irfan Uddin (e-mail: irfanuddin@kust.edu.pk)

This work was partially supported by University of Jeddah and King Abdulaziz University, Kingdom of Saudi Arabia.

ABSTRACT With the massive spike in the use of Online Social Network Sites (OSNSs) platforms such as Web 2.0, microblogs services and online blogs, etc., valuable information in the form of sentiment, thoughts, opinions, as well as epidemic outbreaks, etc. are transferred. With the OSNSs being widely accessible, this work aims at proposing a novel approach for disease (dengue or flu) detection based on social media posts. For this purpose, an automated approach is designed with the help of LSTM (Long Short Term Memory) and word embedding techniques. Then the performance of the proposed approach is validated using a set of standard evaluation matrices. In addition, the effectiveness of the selected models is evaluated with performance measurement techniques. The accuracy of the proposed research approach is evaluated using two word embedding techniques; Word2Vec with Skip-gram (SG) and Word2Vec with Continuous-bag-of-words (CBOW). Based on the results conducted in this paper the LSTM Word2Vec with CBOW achieved better results compared to LSTM with Word2Vec SG features embedding technique. Our findings prove that the proposed method yields 94% accuracy compared to state-of-the-art approaches. Consequently, LSTM performed better than other leading methods in the detection of disease-infected people in tweets. In the end, spatial analysis is performed to identify the disease infected region.

INDEX TERMS Social Media; Disease Detection; Deep Learning; Word2Vec; and LSTM.

I. INTRODUCTION

With the growing availability of OSNSs platforms such as Facebook, microblogs services, online blogs, etc. there is a strong data repository for analyzing the public's sentiment, views, and perspectives on various topics like entertainment, sports, politics, and education, etc. These entire platforms also transfer valuable information when there is an epidemic outbreak in a region. OSNSs can be used in the context of outbreak awareness and health promotion to analyze social media content, to detect an epidemic outbreak, and provide early warnings.

OSNSs could efficiently be utilized to detect disease infected people and impacts of disease on health promotion (e.g., dengue, ILI, HIV, flu, depression, etc.) with an intervention to indorse public health[1]–[3], and early detection of mental disorders and suicidal ideation from OSNSs content [4]. With the increasing popularity of OSNSs, the trends of early alert about the epidemic outbreaks can be detected and the time that passes between occurrence and detection can be decreased. This was previously reliant on health workers and physicians to report disease incidence [5], [6]. The traditional approaches of detecting epidemic outbreaks are when people were

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3031174, IEEE Access

diagnosed with a disease they report to the local health center, which can then notify relevant health care providers to respond and deliver services for tracking that epidemic. This process often takes weeks before the information is recorded and in certain situations, valuable lives are lost before appropriate steps are taken.

During the emerging outbreaks of infectious diseases, public health is the main issue as people regularly make use of OSNSs for information. To make the right choice at the right moment, health care professionals should be informed on the epidemic outbreaks of public health and disease influencing their societies [7]. The rapid identification of an epidemic outbreak is important to deliver a faster and more efficient response for healthcare professionals. The epidemics outbreak can lead to severe diseases, such as flu or influenza-like-illness (ILI), and can cause death when that disease epidemiologically breaks in a region [7], [8]. Similarly, dengue infection is a mosquito-borne virus causing serious ILI and often ended up causing a possibly life-threatening risk factor called severe dengue fever infection. Dengue has been widespread in different countries with a high risk of outbreaks. To provide real-time monitoring and early detection of infectious diseases like flu or ILI and dengue outbreaks can bring major benefits to public health in high-risk areas [7], [9].

In global policy, early warning of disease detection could reduce the impact of seasonal outbreaks (i.e., dengue or flu) to promote public health. Now OSNSs can be utilized to track epidemic outbreaks to track the risk of an outbreak faster than health care practitioners and government agencies such as the American Center of Disease Control and Prevention (CDC) [7]. CDC uses the Influenza-like-Illness Surveillance Network (ILINet), a platform configured by health practitioners for monitoring early warnings of ILI. Although it is a reliable method but overpriced and time-consuming as it requires days or weeks to make information available. Consequently, various researchers, concentrate on developing solutions using OSNSs for tracking ILI and identifying early alerts regarding epidemic outbreaks to conduct real-time research. OSNSs platform like Twitter produces a large amount of information related to epidemic outbreaks. These information can be configured for the detection of an epidemic outbreak in a region to track early warnings [10], [11]. Health care practitioners can be notified by OSNSs to deliver appropriate resources to track an epidemic. Another motivated works were carried out in [6], [12], on Sentic patient reported outcome measures (PROMs) In their work they proposed that patients, in fact, are typically able to express their views and sentiments in free text, instead of simply filling in a questionnaire, either to express their happiness or for cathartic complaining. Sentic PROMs help patients to assess their health status and experience in a semi-structured way and thus collect input data through Sentic computing while detecting physio-emotional sensitivity of patients [6].

According to our knowledge, none of the existing works utilized LSTM with Word2Vec approaches for disease detection in OSNSs. This paper designs a novel approach to combine numerous data mining and artificial intelligence techniques to detect disease-infected people in tweets, such as data preprocessing, Natural Language Processing (NLP), and Deep Learning (DL) techniques and analyzing the most affected regions during the outbreak. The novelty contributions of the proposed work are as follows: 1) to understand the sentiment of tweets regarding two diseases dengue or flu, and make a group of infected people and analyze if a group is increasing in size at an alarming rate. 2) To propose a novel LSTM [13] method for disease (e.g., dengue or flu) detection with word embedding techniques to capture the semantic relationships among the words in OSNSs text for better detection. 3) To improve the performance of the proposed work compared to other state-of-the-art techniques. The new approach of detecting disease outbreak on OSNSs using LSTM with word embedding techniques leads to the research hypothesis as follows: H_1) the proposed LSTM based approach detects disease infected people in tweets with Word2Vec techniques such as (CBOW and SG). H_2) LSTM with Word2Vec can achieve better performance over state-of-the-art Machine Learning (ML) techniques. The proposed approach is applied to tweets data (dengue/flu). The development of the empirical analysis confirmed these hypotheses in Section IV.

The remaining sections of this paper are outlined as follows: Section II provides preliminaries literature on current solutions. A methodology of the proposed work is presented in Section III, while the results and evaluation are presented in Section IV. Section V presents spatial analysis while Section 6 concludes the paper and stretches recommendations for future research development.

II. PRELIMINARIES LITERATURE

This section presents a review of related studies conducted: I) on disease detection in OSNSs using ML algorithms and II) related studies on DL methods such as Recurrent Neural Networks (RNNs) with LSTM and Word2Vec approach.

A. Literature on Disease Detection in Social Media

This section presents a review of related studies conducted: 1) on disease detection in OSNSs using ML algorithms and 2) related studies on DL methods such as Recurrent Neural Networks (RNNs) with LSTM and Word2Vec approaches.

1) LITERATURE ON DISEASE DETECTION IN SOCIAL MEDIA

The related work conducted on flu and dengue outbreaks using OSNSs data as follows.

i) Early Detection of Flu Outbreak in Tweets

OSNSs platform like microblog service (Twitter) provides opportunities for real-time outbreak surveillance as it

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3031174, IEEE Access

produces a massive amount of data regarding epidemic outbreaks. N. Collier et al. [14] has concentrated on monitoring the influenza-like illness in tweets by using 5283 tweets related to influenza and presented a strong level of inter-annotator agreement i.e., kappa 0.85. This work applied ML approaches using two supervised learning classifiers Naïve Bayes (NB) and Support Vector Machine (SVM) with unigram, bigram.

A social media information can be used efficiently to detect epidemic outbreaks and provide an appropriate alarm for public awareness [15]. The main goal of the work was to present a model that detects ILI using three modules classification, visualizing, and prediction of the flu outbreak using linear regression. Similarly, another study proposed a model for flu detection in tweets using SVM which is an ML approach for classification. The study demonstrated a strong correlation as 0.89 to the gold standard [16]. Moreover, health-related twitter data for monitoring public health in real-time was analyzed by A. Jimeno Yepes et al. [17], to consider an appropriate tweet that contains a health entity name. The entire data was run in a domain-tagger name entity to identify disease-related tweets. The work ignored the people infected from the disease in tweets. S. Wakamiya et al. analyzed [18] the public sentiment in Japan regarding swine flu and H1N1 (Hemagglutinin 1 Neuraminidase 1). The work deployed regression and statistical techniques to utilize the information from tweets messages including different cities in Japan to detect tweets containing the keyword *#flu* by identifying the ILI in the states of Japan. This study applied NLP based classification techniques, to classify the tweet messages regarding flu or ILI.

K. Lee et al. [19] used ML approaches for identifying disease tasks in tweets. Two diseases (such as flu and cancer) were analyzed on frequency-based using the OSNSs platform Twitter. The objective of their geographic analysis in US states was to monitor the flow of epidemic in US regions by analyzing the quantity of these two diseases produced in the states. In order to analyze the location information, the details about the location were generated from the user timelines. The dataset contained data of all users who mentioned the information about flu or cancer and has accurate US regions details. The novelty of the work was to detect disease outbreak regarding flu or cancer in the US states and to provide real-time surveillance. Significant features including infected people, sentiment about the disease, and alarming situations were not considered in this study and thus seem the main limitations of this research. Similarly, H. Xue. Et al [20] used Support Vector Regression (SVR) ML based model for estimating regional-based influenza incidence in the US. In their work, they configured two datasets: one that supports CDC data and the other US-based tweets data to optimize to train the SVR model.

ii) Early Detection of Dengue Outbreak in Tweets

In the 20th century, the dengue was first found epidemiological as the exhaustive mosquito infections in humans [21]. Epidemiology outbreaks are a serious risk for health. Precise and early surveillance of the epidemic risk and advancement can reduce its effects. A seasonal disease outbreak can be monitored and assessed through OSNSs and can be used to assess the activity of the outbreak through social media posts [9]. The dengue virus pattern is complicated to predict and track because of an expensive and slow monitoring system. The novelty of the work in [9], was to experimentally evaluate the significance of Tweet information for the early identification and tracking of a dengue outbreak on a weekly basis, regarding both at country and community level in Brazil. An ML based regression method was trained for assessing the number of disease tweets in the same week to forecast epidemic in the future week.

Another study has been conducted by J. Albinati et al. [22], where they deployed the potential of online data sources and proposed an epidemiological model by analyzing tweet messages for the detection of the dengue outbreak in Brazil. Similarly, the potential of Twitter data for monitoring dengue in the Philippine was considered in [23]. The study revealed that the geographically distributed information was compared to the geographic information of new cases of dengue outbreak as reported by public health departments in the Philippines. The focus of the work was to collect the tweet messages that contain the keyword *#dengue* in Philippine states using ML approaches such as the SVM and regression model to show the result of their analysis.

K. Espina et al. [24] analyzed the 2017 dengue outbreak in the Philippines by using different ML approaches. Their work showed the use of real-time data from OSNSs particularly from microblog platform Twitter, to enhance the current initiatives and to monitor epidemic outbreaks. To classify health-related tweets their work has shown a range of dengue rates and typhoid fever in the Philippine by utilizing SVM for classification and regression was for possible disease rate. Based on the Philippines Health Ministry these both outbreaks were highly correlated (e.g., correlation (R) > .75) between tweet messages and monitoring information [24]. According to a recent survey in China, the CDC presented the rate of dengue cases has been increased from 0.00089 to 3.5471 per 10,000 people. Then the forecasting model for dengue was built by generalizing ML models including SVR, gradient boosted regression tree, linear regression model, and binomial regression model [25], [26].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3031174, IEEE Access

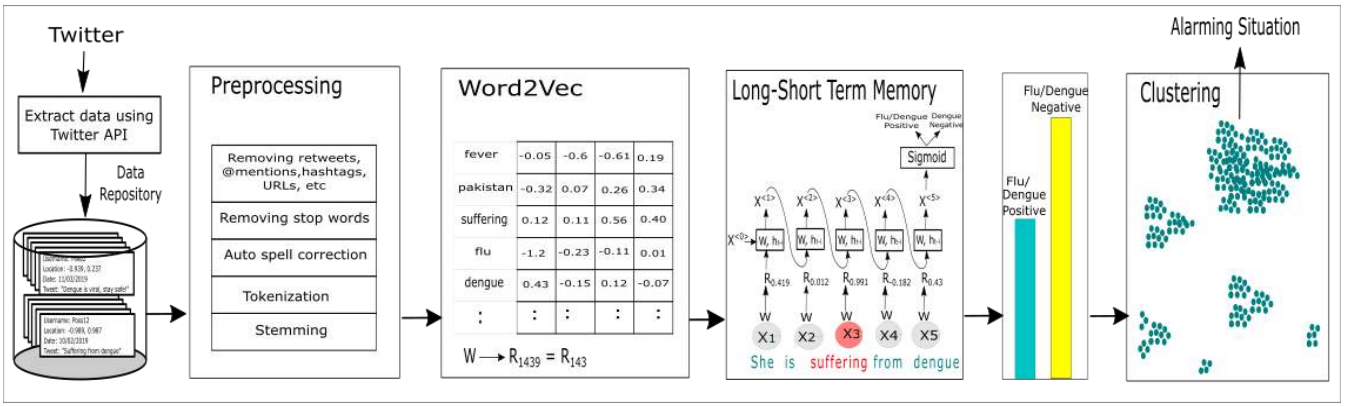


FIGURE 1: A proposed methodology for identification of disease infected people in tweets and clustering based on regions

2) LITERATURE ON DL METHODS (RNNS/LSTM AND WORD2VEC)

In addition to the aforementioned literature, some other studies have also been conducted on Twitter and other microblogs data by utilizing DL approaches, for instance, Word2Vec [27], LSTM [13], Convolutional Neural Networks (CNNs) [28], and word embeddings techniques [29]. The literature regarding these approaches is discussed below.

i) Related Works: Based on RNNs/LSTM using Tweets Data

Some studies have proposed events detection methods from OSNs by utilizing ML and DL approaches [30]. Hernandez-Suarez et al. [30], proposed a method for monitoring natural disasters by configuring the Spatio-temporal information regarding the user-posted comments and opinions. The goal of their work is to present a useful geo-temporal framework that may seem at the time of event occurring and after the event, that can be useful in assessing the extent of the damage. They used RNN with a word embedding technique to ensure high accuracy of classification for sequence data. A predictive model was built for riots during England's 2011 and antisocial events [31]. The novelty of their work was to identify the disease from tweets to interact with the public and technology, to comprehend how events are reported using OSNSs and how to extract meaningful information from social media posts, which supports to make conclusions and lead to fruitful achievement.

In a broad context, the research work has many potential applications, such as sarcasm detection in the text, which is already an established field where RNN is commonly used [32], [33]. Zhang et al. developed [33] a model where they focused on syntactic and semantic information of tweets, using a hybrid of Gated Recurrent Unit (GRU) and Artificial Neural Network (ANN). Moreover, Bark et al. [34], developed DL approaches to analyze the performance of sarcasm detection from tweets, specifically by implementing two models of RNN, one with LSTM cells and one with GRU cells with augmentation of CNN.

Another interesting work was carried out in [35], where word embeddings techniques are used to capture the context incongruity in the absence of sentiment words to show the benefit of features for sarcasm detection. There has been recent work in 2018 where Ajao et al. [36] developed a model that specifically focused on the propagation of fake news on twitter. Their study involved implementing two models such as LSTM and CNN for fake news detection and classification. B. Jang et al. presented [37], a CNN with Word2Vec approaches by classifying news article and tweets into related and unrelated.

In contrast to many other studies, the former studies are motivating as they focus on the detection of reported diseases on social media, whereas the later studies are interesting as they present some fascinating ideas about how to make an efficient approach for disease detection and automatic surveillance in real-time to reduce the risk caused by epidemic outbreaks using DL approaches. To address the limitations in the previous studies, this research work is motivated to discover a new method to detect disease outbreaks in social media. Consequently, the work proved (presented in the next section) that tweets data can be utilized to identify disease infected and non-infected people from tweets as a result of their posting actions/behavior on social media text.

III. PROPOSED METHODOLOGY

With a promising role of OSNSs for epidemic outbreaks, the proposed model designs a novel approach for disease detection based on OSNSs contents. The proposed architecture tackles the issue of detecting disease infected people (dengue/flu infected) to utilize DL techniques such as LSTM and Word2Vec architectures (CBOW and SG). The methodology of the proposed approach is followed to provide a novel way to analyze tweets data to identify disease-infected people in tweets that can be seen in Figure

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3031174, IEEE Access

Pseudocode 1: Data Collection and Data Wrangling

```
Begin
Input:  Raw data (unstructured data)
        Output: Preprocessed tweet (structured data)
        Parameters: Tweet: T^w, URL, Re-Tweets: RT_w,
        Symbols: Symb
# Data Preprocessing
1.  Load data file
2.   foreach tweet do
3.         language = check_language (tweet)
4.    if (language = "english")
5.          then
6.             save_to_database (tweet)
7.          else
8.             delete_from_database (tweet)
9.    end if
10.    if (tweet =! lower case)
11.         convert_to_lower_casing (tweet)
12.         then
13.            preprocess (tweet)
14.    end if
15.   repeat preprocess until
16.        remove RT_w, #Tags, URL, Symb, emoticons
17.  end for
18.  do step 2-17 for all tweets
19. end
End
```

Pseudocode 2: Data preprocessing(tokenization, stemming)

```
Begin
        Input:  Corpus of tweets (dengue and flu)
        Output: Word tokenization, word stemming
        and Vector representation of each word
1.do breakdown a tweet into word tokens using NLP
2.    techniques
3.    foreach tweet in corpus do
4.           token ← word_tokenize (tweet)
5.    end for
6.  end tokenization
7.  do stemming using porter stemmer
9.    for each tweet in corpus do
10.       StemWord ← PorterStemmer.stem (tweet)
11.   end for
12. end stemming
13. Save relevant features into database
14. repeat step 1-13 for all data
15. do transform text into vectors using word2vec
16. techniques
17.   for each tweet in corpus do
18.   tweetVec ← word2vec (tweet, min_count = 3)
19.     return
20.           vector representation of each word
21.   end for
22. end embedding
End
```

1. The proposed method comprises of the four modules to explore tweets' data to detect
dengue/flu infected people in tweets. Each of the modules is explained in the following subsections.

A. Data Collection

This section analyzes data collection and data preprocessing. A dataset consisting of dengue/flu corpus extracted from twitter by deploying Twitter Streaming API (Application Programming Interface)[1]. Twitter streaming API is a free and open-source platform that permits researchers to access tweet data in real-time. With the help of this scraper, 551,900 tweets are gathered from August 2018 to December 2019, distributed over the keywords #flu, #influenza, #dengue, #denguefever. The breakdown is given in Table.1.

TABLE 1: Total number of retrieved tweets split over dengue and flu

Disease	Number of Tweets
Dengue	359,410
Flu/Influenza	239,501
Total	598,911

The tweet containing the word "*dengue", "denguefever", "flu"* or *"influenza*" are retrieved in Java Object Notations (JSON) and stored in a data repository. During preprocessing the important parameters including user id, tweet text, tweet spatial information, and tweet time stamp are extracted from the corpus. After preprocessing all the parameters are stored in a machine-readable form such as comma-separated values (.csv) for further analysis.

B. Data Preprocessing

Once the raw unstructured data are extracted then retrieving valuable information from an unorganized text corpus is a challenging task. Data is preprocessed by deploying the NLP preprocessing techniques to normalize the data to be applied as a feature. This comprises hyperlinks, @mentions, retweets, punctuations, URLs, stop words removal, special characters, stemming, and tokenization. This process is aimed at presenting the corpus in a form that can be processed efficiently to enhance their output by eliminating the contents that are unrelated to the corpus. In tokenization, the tweet text is split into word tokens while in the stemming technique the word is reduced to its root or base word by utilizing NLTK porter stemming methods in Python[2].

[1] https://developer.twitter.com/

[2] https://www.nltk.org/_modules/nltk/stem/porter.html

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3031174, IEEE Access

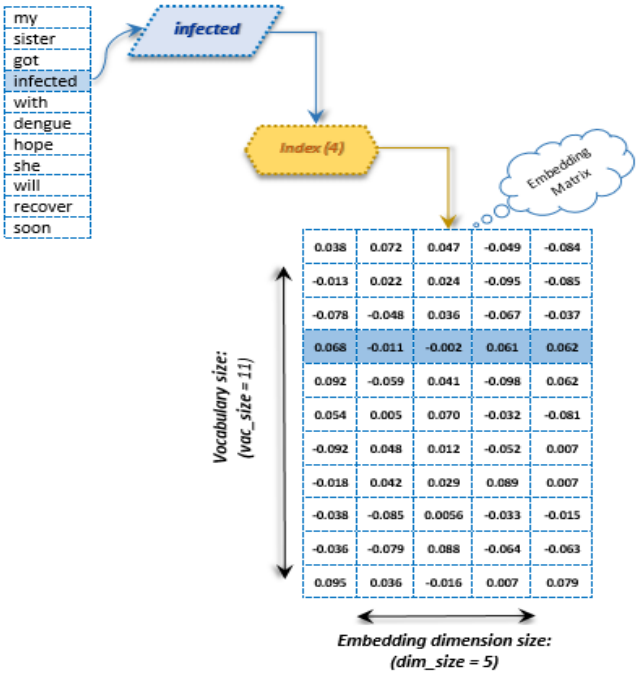


FIGURE 3: Word representation in vector shape

After preprocessing a subset of *6000 tweets* was labeled by three individuals to remove the biases in labeling. A tweet that represents that someone is infected from dengue or flu a label is 1 marked and the remaining tweets that represent some information about that specific diseases a label 0 is marked. After that, the labeling are acknowledged through the inter-annotator agreement level by deploying Cohen's Kappa test [38], and found it strong (kappa=0.843) [39]. The pseudocodes for data collection and data preprocessing are demonstrated in pseudocode 1 and 2 respectively.

C. Features Extraction: Semantic Analysis using Word Embedding techniques

The text data are embedded in numbers for computation using word embedding techniques. Since ANNs or RNNs [40], [41] are not able to process text data directly. Alternatively, the text data are converted into numbers for computation. The traditional method that converts the text data into numbers is a Term Frequency Inverse Document Frequency (TFIDF) [42]. However, there are more promising word embedding techniques such as Glove [29], Doc2Vec [43], and Word2Vec [27], etc. In this paper, the Word2Vec embedding technique is utilized that is presented below.

1) WORD2VEC

In the literature, *TFIDF* is most commonly utilized for word embedding. Our proposed model is evaluated on the advanced word embedding technique such as *Word2Vec* (*Word to Vector*) proposed by Mikolov et al. [27]. Word2Vec model generates the number of vectors (dimension) for each word in the corpus to search at the context based on how the text (words) occurs in a sentence. In word2Vec, all words that have a similar context are positioned around each other in the vector space.

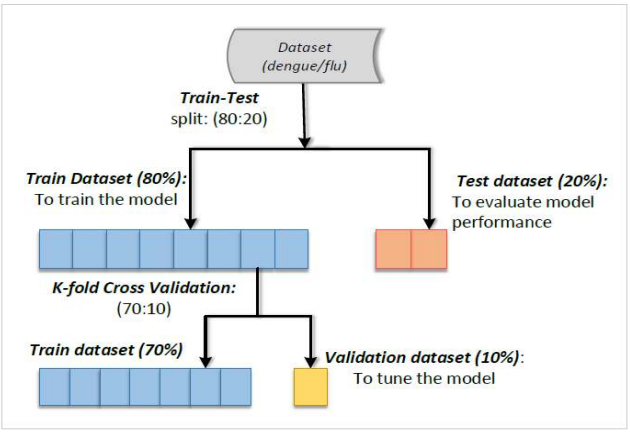


FIGURE 2: Data splitting (Train, validation and test set)

To understand the insight of word embedding we demonstrate a tweet example: "*My sister got infected with dengue hope she will recover soon".* Preprocessed word tokens of the tweet are fed into n-dimensional embedding (5-dimensional in our example) where the word "*infected*" with index 4 is holding vectors *[0.068, -0.011, -0.002, 0.061, 0.062]* indicated by the fourth row as highlighted in Figure 2.

To allow the model to understand, every word in the corpus is converted into vectors by utilizing Keras embedding layer techniques. For the entire dataset, we created a vocabulary consisting of 10,000 features extracted from the labeled data. For the whole train data, a vector of 100 dimensions is deployed using a genism library[3] in Python for Word2Vec (see Table 2).

There are two general learning methods in Word2Vec, namely *Skip-gram (SG)* and *Continuous Bag-of-Words (CBOW)* model. The efficiency of these two models (CBOW and SG) have been compared in [44]. The literature studies indicate that CBOW architecture is particularly used for a medium dataset (small) while the SG architecture is designed for large data size.

i) Skip-gram (SG)

SG model performs efficiently with a large number of training data. To provide a word at the center (target word), the SG model tries to predict neighboring context words [45]. Neighboring words are defined by the windows size. To understand the insight of the SG approach, take an example tweet such as, *"She is suffering from dengue."* If a windows

[3] https://radimrehurek.com/gensim/models/word2vec.html

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3031174, IEEE Access

size of two and source (target) word is "*suffering"*, in this case, the neighboring words are *(she, is, from, dengue)*, and hence the input and target word pairs will be *(suffering, she), (suffering, is), (suffering, from), (suffering, dengue).* It is considered that the distance of the word to the target word performs no significant role throughout the sample windows. To train the model the context words *(she, is, from, and dengue*) are preserved the same.

ii) Continuous Bag of Word (CBOW)

Another approach as *CBOW* attempts to detect the target word dependent on the neighboring context words [45]. It is an important architecture that works on a small or medium size of training data to learn the relationship between words. For better understanding, consider an example: "*she is suffering from dengue".* CBOW will understand the neighboring context words, and subsequently, CBOW will determine "*suffering*", "*infected*", *"diagnosed*" or *"down"* are the most likely words (probably words) at a randomly picked nearby by *suffering*. Words like "*sitting*" acquire less of the network's consideration as it is programmed (trained) to determine (predict) the most likely word. To capture the broader context of the word in social media text (tweet), this paper prefers the *Word2Vec (a self-supervised technique)* models (i.e., SG and CBOW) for better detection and classification results.

D. Train – Validation and Test Split

In this section, the data splitting technique is presented, the corpus dataset is split into three parts: trainset, validation dataset, and test dataset. To develop a model, *6000* labeled and preprocessed tweets are selected from the dengue/flu corpus. To utilize the k-fold cross-validation approach [46], [47], we split the dataset into an *80:20* ratio, where *80%* for model training while *20%* are utilized to test the model. The train, validation, and test dataset splitting are graphically depicted in Figure 4.

Train set: the training dataset is utilized for model training. We used 80% data for training and it can differ according to the model's performance criteria. The training set contains the input values as well as the detected (known) output, consequently the model trains (learns) on this train data to be adapted to unseen data (new data) subsequently. Therefore, it incorporates the input data as well as the output values.

Validation set: as discussed above k-fold cross-validation is utilized for data splitting. Therefore, the train set is further divided into a validation set. The validation set is used to avoid under-fitting and overfitting [46],[47]. It commonly occurs when the model's performance accuracy is decreased to test set while the training period's accuracy is strong than test accuracy. To apply parameter-tuning techniques, a *10%* dataset is assigned for validation to minimize the model's performance (output) error on test data. Aimed at this purpose, automated dataset evaluation is employed to offer fair (impartial/unbiased) model assessment to minimize the overfitting or under-fitting and to ensure better accuracy and efficiency of the model.

Test set: to evaluate the model performance and efficiency, the remaining data *(20%)* is allocated for model testing. To see whether the machine being trained is predicting (performing) well on unknown data (unseen) based on its training set by applying some performance matrices. Subsequently, the test set is operated for the ultimate assessment/evaluation once the model is properly equipped (trained).

E. Baseline methods and Long Short Term Memory

Traditional methods are typically developed for classification and detection purposes [15], [18], but there are some challenges when the sequence data (sentences, time series, or sounds) are treated. For example, in ANN [40], [48] parameters for input and output are static in size, which may not be the case when textual data are interpreted. In this paper, tweets data are deployed and the size of the input parameters is different than the size of the output parameters. Furthermore, ANN does not share parameters adapted across different positions in a text corpus. To allow better identification of a sequence of textual data, learned parameters must be distributed around various positions in a network layer. For this purpose, RNN is developed to process sequence data [41],[49].

RNN has a vanishing gradient and exploding gradient problems as it is not able to process long sentences [50] [51]. During backpropagation to go back to adjust the weights then the signal becomes either too weak or too strong to process long sentences, which leads to the vanishing or exploding gradient problems. To avoid this problem, RNN based variants GRU [52] and LSTM [13] are developed. In this paper, LSTM is adapted to efficiently process sequence data.

i). Long Short Term Memory

LSTM is a modified variant of RNN also known as a fancy recurrent neural network proposed by S. Hochreiter et al. [13], which has a memory state that has the ability to remember information and learn long-term dependencies for long periods. According to Colah [53] LSTM model is in the form of a chain structure in nature. The repeating module is structured differently rather than a neural network as it has four interacting layers to perform their task efficiently. A standard LSTM architecture consists of the cells called memory blocks. The LSTM architecture consists of four gates (such as forget, input, candidate, and output gate), which are used to decide which information should be preserved and which should be removed. LSTM architecture has the ability that helps to focus when to forget and how to prolong to keep the state information via the adapted gates and memory blocks. The secret to LSTM is the cell state C_t that allows the uninterrupted flow of information. To allow the information pass, the cell state is governed by three gates. The forget gate (Equation 3) is adapted to manage which cell

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3031174, IEEE Access

state C_{t-1} will be ignored that is not required. While σ is a sigmoid function and W_f is a weight matric and b_f is a bias matric.

$$f_t = \sigma\left(W_f\left[h_{t-1}, X_t\right] + b_f\right) \quad (1)$$

The following equations are calculated to determine and save new input X_t information in the cell state as well as updating the state of the cell. This consists of two functions such as sigmoid and *tanh* functions. Initially, the sigmoid function determines whether to update or neglect the latest information (i.e., between 0 and 1) and subsequently, the *tanh* function generates weights to the value yielded in the range between (-1, 1). To multiply the two values for updating the current cell configuration. Then this current memory state is provided to the previous memory state C_{t-1} likely to result in C_t.

$$i_t = \sigma\left(W_i\left[h_{t-1}, X_t\right] + b_i\right) \quad (2)$$
$$\tilde{c}^{<t>} = tanh\left(W_c\left[h_{t-1}, X_t\right] + b_c\right) \quad (3)$$
$$C_t = \left(C_{t-1}\, f_t + \tilde{c}^{<t>} i_t\right) \quad (4)$$

In the final process, the output (h_t) is calculated based on the state of the output cell (O_t) but will be a filtered version. After that, the sigmoid function (σ) determines which cell state part makes it to predict the output. Then $tanh$ is applied to (C_t) the element-wise multiplication is calculated with the output (O_t) is multiplied produced the resulting values between (-1 and 1).

$$O_t = \sigma\left(W_o\left[h_{t-1}, X_t\right] + b_o\right) \quad (5)$$
$$h_t = O_t\, tanh\left(C_t\right) \quad (6)$$

For other layers, the hyperparameters are maintained with fixed sizes including vocabulary size, embedding layer, the maximum length of words in the vocabulary, minimum words in a tweet, windows size, and dropout, etc. (see Table 2).

F. Evaluation Matrices

In order to compare the state-of-the-art techniques with the proposed LSTM method, a confusion matrix [54] is generated. Then a set of standard evaluation matrices such as accuracy, precession, F1 score, recall, and Receiver Operating Character Curve (ROC Curve) are configured by evaluating the performance of the proposed model on test data over baseline methods. The equations for these matrices are depicted in (9), (10), (11), and (12). Where accuracy is a proportion of correctly classified classes; precession is a proportion of the positively detected classes; recall measures the proportion of correctly detected positive classes and the F1 score is a harmonic mean of recall and precession. In confusion matrices, the number of false negative (TN), false positive (FP), true positive (TP), and true negative (TN) are determined as shown in equations (9. 10, 11, and 12) as follows.

$$accuracy = \frac{TF + TN}{TF + TN + FP + FN} \quad (7)$$

$$precession = \frac{TF}{TF + FP} \quad (8)$$

$$recall = \frac{TP}{TP + FN} \quad (9)$$

$$f1\ score = \frac{precession \times recall}{precession + recall} \quad (10)$$

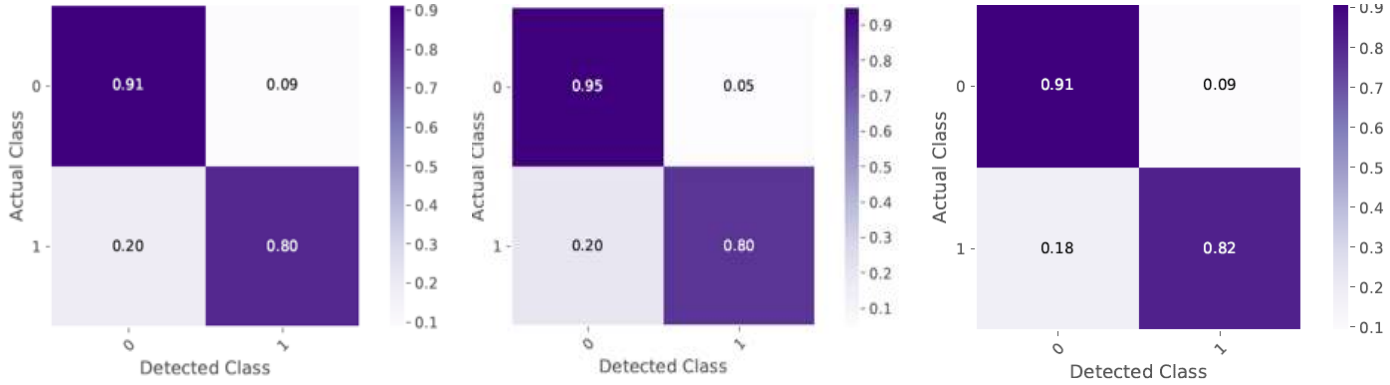
ROC is a performance evaluation measurement for supervised learning problems. ROC is deployed to determine the classification problem. ROC demonstrates how much the trained model is efficient to differentiate among classes. It reveals that the higher the performance the better the model would be to detect the actual positive class as positive (dengue/flu infected people) and actual negative class as negative (no dengue/flu).

TABLE 2: Overview of the model design

Hyperparameters	Values
Vocabulary_size	10,000
Embedding_layer	100
Max_length	10,000
Minimum words	3
Learning_rate	0.001
Windows_size	2
Dropout	0.5
Epochs size	40

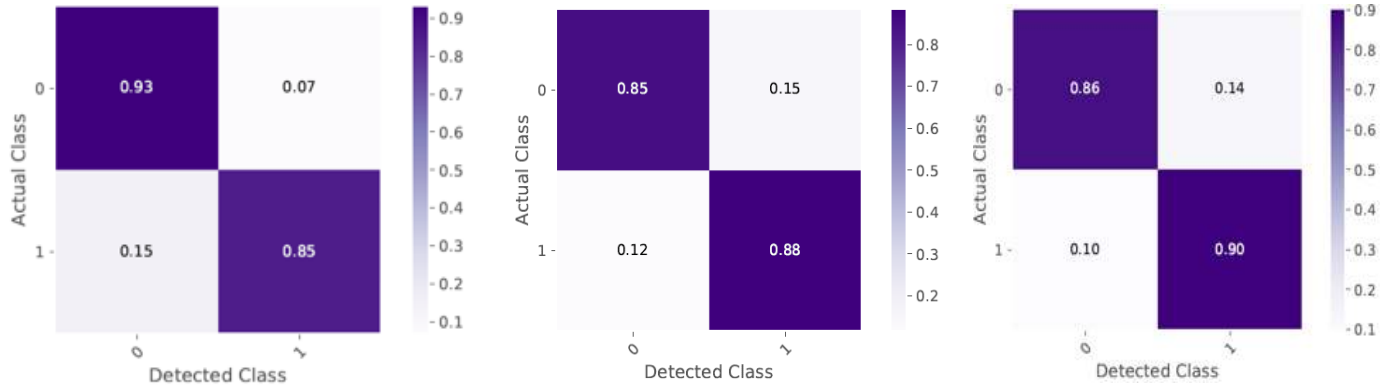
IV. RESULTS AND EVALUATION

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3031174, IEEE Access



(a) LR (b) NB (c) SVM

FIGURE 4: Confusion matric for SG (a) LR, (b) NB, and (c) SVM



(a) ANN (b) DNN (c) LSTM

FIGURE 5: Confusion matric for SG (a), ANN, (b) DNN, and (c) LSTM

This section presents the ML and DL techniques by utilizing Word2Vec features extraction methods (SG and CBOW). Results are conducted by using Anaconda Python 3.6 with the tensor flow, Keras[4] and Scikit learn modules[5]. The following sections break down the performance results for each model in detail.

A. Detection and Classification Results with Word2Vec (SG and CBOW)

This section presents the results for the proposed method using Word2vec (SG and CBOW) embedding techniques. The embedded vectors of Word2Vec are fed to ML and DL methods (i.e., LR, SVM, NB, ANN, DNN, and LSTM) to train the model for classifying them into disease positive or disease negative tweets and detecting dengue/flu infected people.

1) Results of the Proposed LSTM Method over Baseline Methods Using Word2Vec with SG Embedding Technique

In this section, the results for Word2Vec with the SG feature extraction technique are demonstrated. Table 3 illustrates the accuracy of the proposed method for the selected models when Word2Vec with SG is applied as the features extraction technique.

TABLE 3: Train, test, and validation accuracy using Word2Vec with SG technique

Method	Train Accuracy (%)	Validation Accuracy (%)	Test Accuracy (%)
LR	89.3	87.0	87.3
NB	84.8	83.7	83.7
SVM	89.4	87.4	87.5
ANN	92.1	90.5	90.4
DNN	93.0	90.8	91.3
LSTM	**94.3**	**92.6**	**93.2**

[4] https://keras.io/getting_started/

[5] https://www.anaconda.com/products/individual

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3031174, IEEE Access

Table 4 reveals the results of precession, recall, and f1-score achieved by each of ML and DL techniques on dengue/flu data when used with the Word2Vec feature extraction method (SG).

TABLE 4: Accuracy, precession, recall, and f1-score using Word2Vec with SG

Model	Accuracy (%)	Precession (%)	Recall (%)	F1-Score (%)
LR	87.0	87.4	87.3	87.5
NB	83.7	86	85	86
SVM	87.5	88	88	88
ANN	90.4	90	90	90
DNN	91.3	91	91	91
LSTM	**93.2**	**92**	**92**	**92**

The data presented in the above table (see Table 3 and 4), it can be concluded that the learning model LSTM delivers the best result than other models. The experimental results show that the performance of the other models is also enhanced.

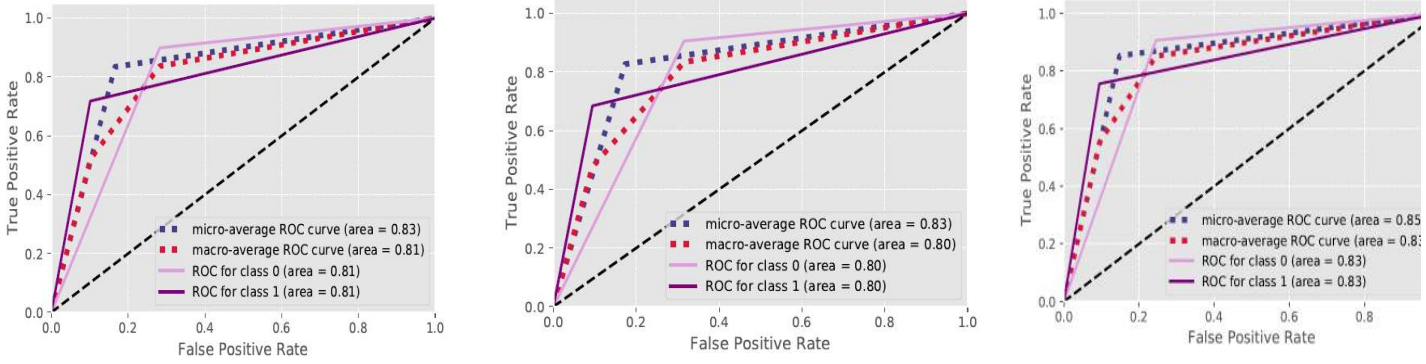
i) Confusion Matrices for Word2Vec with SG

In this subsection, the performance of the model is graphically visualized using the Word2Vec embedding technique (SG) by generating a confusion matrix and ROC curve using different hyperparameters evaluating the model performance on the test data. Figures 6 and 7 demonstrate the confusion matrices for the selected ML and DL techniques to detect the predicted class from test data in accordance with the train data.

From the below plots of confusion matrices, it is inferred that the results of the LSTM with the SG embedding technique (see Figure 7 (c)) are higher for the correctly predicted positive and negative classes. Similar inference can be drawn for the rest of the models with the SG embedding technique.

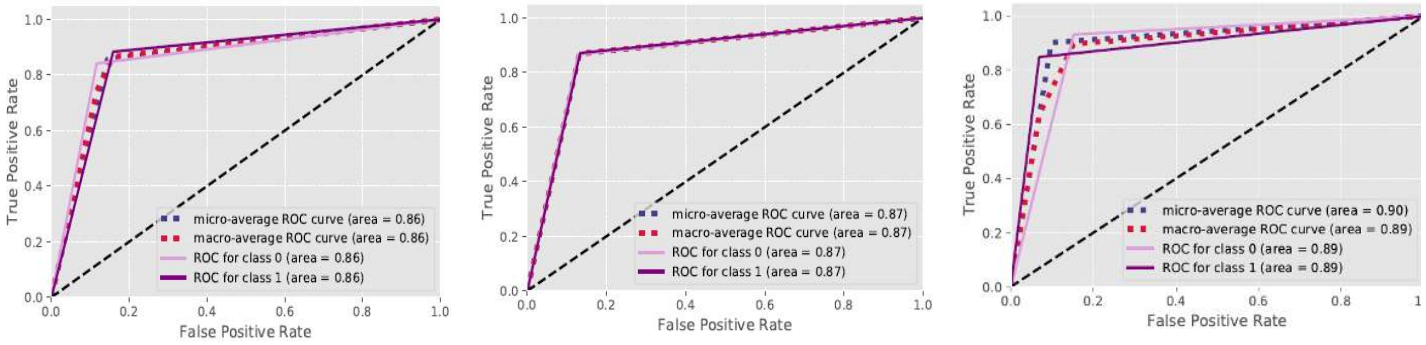
ii) ROC Curve for Word2Vec with SG

In this section, we present the ROC Curve for the selected models with the Word2Vec embedding technique SG. Figures 8 and 9 yielded that the proposed LSTM efficiently performed with the SG embedding technique as compared to other models. Similar inference can be drawn for the other models when compared to traditional ML techniques.



(a) LR **(b) NB** **(c) SVM**

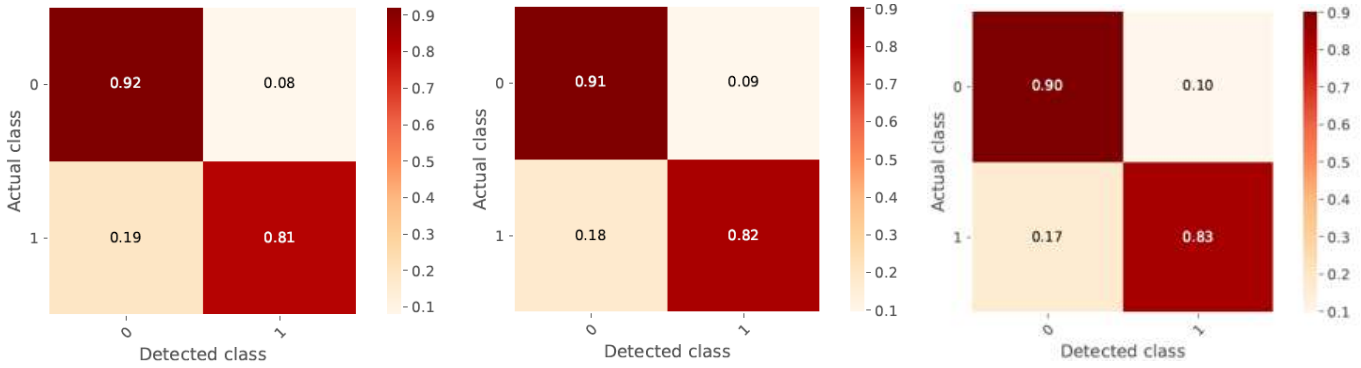
FIGURE 6: ROC Curve for SG (a), LR, (b) NB, and (c) SVM



(a) ANN **(b) DNN** **(c) LSTM**

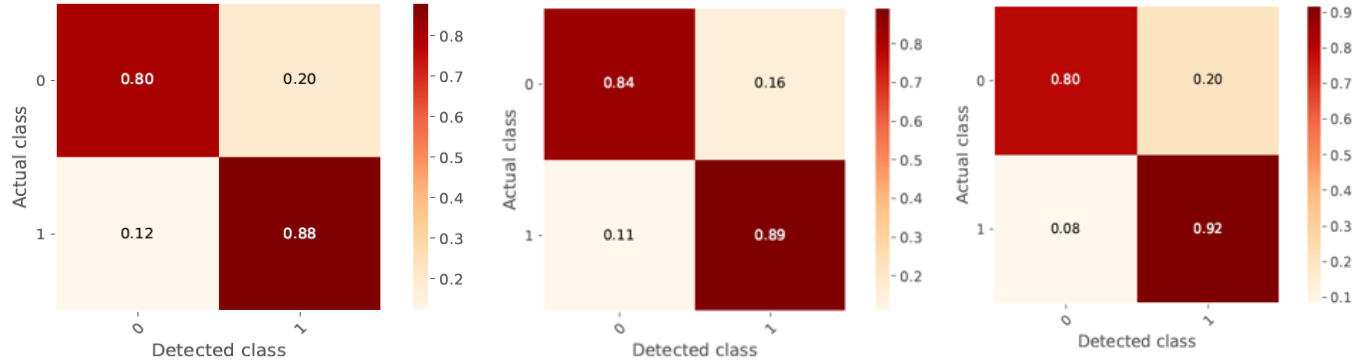
FIGURE 7: ROC Curve for SG (a), ANN, (b) DNN, and (c) LSTM

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3031174, IEEE Access



(a) LR **(b) NB** **(c) SVM**

FIGURE 8: Confusion matric for CBOW (a) LR, (b) NB, and (c) SVM



(a) ANN **(b) DNN** **(c) LSTM**

FIGURE 9: Confusion matric for CBOW(a), ANN, (b) DNN, and (c) LSTM

1) Results of the Proposed LSTM Method over Baseline Methods Using Word2Vec with CBOW Embedding Technique

This section analyzes the results for Word2Vec with the CBOW feature extraction technique. Table 5 reveals the accuracy of the proposed method for the selected ML and DL techniques when Word2Vec with CBOW is applied. Table 6 shows the performance measurements (precession, recall, and f1-score) for each model with CBOW.

TABLE 5: Train, test, and validation accuracy using Word2Vec with CBOW technique

Method	Train Accuracy (%)	Validation Accuracy (%)	Test Accuracy (%)
LR	89.3	87.2	87.5
NB	85.5	84.4	84.7
SVM	89.2	88.1	88.6
ANN	92.1	90.6	90.5
DNN	93.3	90.8	91.7
LSTM	**94.5**	**92.9**	**93.8**

TABLE 6: Accuracy, precession, recall, and F1-score using Word2Vec with CBOW

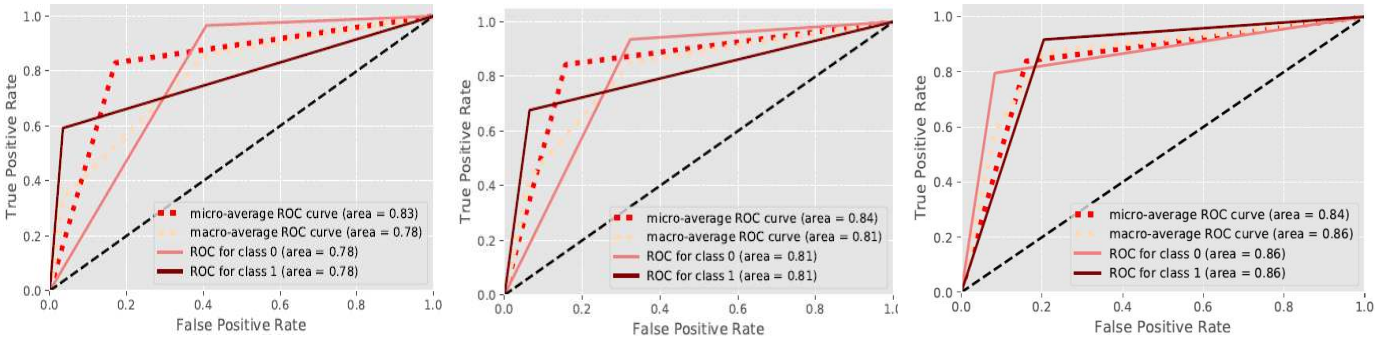
Model	Accuracy (%)	Precession (%)	Recall (%)	F1-Score (%)
LR	87.5	87.5	87.4	87.6
NB	84.7	87	85	87
SVM	88.6	89	88	89
ANN	90.5	91	90	91
DNN	91.7	92	90	92
LSTM	**93.8**	**93**	**92.4**	**94.2**

i) Confusion Matrices for Word2Vec with CBOW Features Extraction Technique

Similarly to what was done for Word2Vec with SG, we also explored the confusion matric for CBOW that is effective in solving classification problems to detect predicted classes with respect to actual classes. Figures 10 and 11 present the confusion matric for the selected ML and DL models.

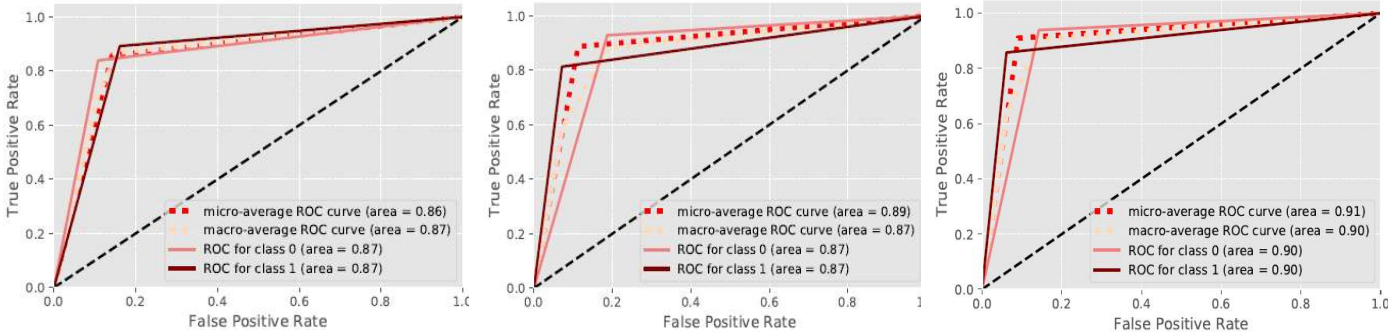
Following the results from the below plots, it can be seen that the LSTM model yielded the best accuracy over the other state-of-art-techniques with CBOW. For other DL

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3031174, IEEE Access



(a) LR **(b) NB** **(c) SVM**

FIGURE 10: ROC Curve for CBOW (a), LR, (b) NB, and (c) SVM



(a) ANN **(b) DNN** **(c) LSTM**

FIGURE 11: ROC Curve for CBOW (a), ANN, (b) DNN, and (c) LSTM

models, similar conclusions can be made from ML methods using the CBOW embedding technique.

ii) ROC Curve for Word2Vec with CBOW

In this section, the performances of the ROC curve for the proposed approach with Word2Vec embedding CBOW are explored. We used ROC for the positive and negative class as well as micro average ROC for both classes. The results of the ROC curve for the selected models are plotted in Figures 12 and 13 as follows.

C. Comparison of the Proposed Method with Baseline Methods

From our experiments, we found that the proposed LSTM with SG and CBOW performed well compared to other models. The result of the LSTM is slightly improved with CBOW compared to SG. This is because CBOW efficiently works for medium-sized text data while the SG model for large corpus. This is a regularizing influence of CBOW that comes out to be beneficial for medium-sized corpus (as stated the size of the annotated data i.e., 6000 tweets).

V. Special Analysis

The geographical analysis of Twitter users is also addressed in this paper. To explore at the country level, the spatial pattern of people infected by dengue, and their sentiment on the disease is also extracted. The geographical data (latitude and longitude) is collected from Twitter by utilizing Twitter API[6]. Figure 14 depicts the highest number of dengue patients originated from the Philippines compared to other surrounding countries.

A. Clustering Techniques

Generally, unsupervised learning techniques are effective for unlabeled data, and when the result or outcome is not actually known that is being tried to predict. Such techniques usually come in two ways i.e., I) clustering techniques and II) dimensionality reduction techniques. In this paper, the K-means clustering technique is deployed for spatial analysis as follows.

[6] https://developer.twitter.com/en/docs/tutorials/%1cltering-tweets-by-location

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3031174, IEEE Access

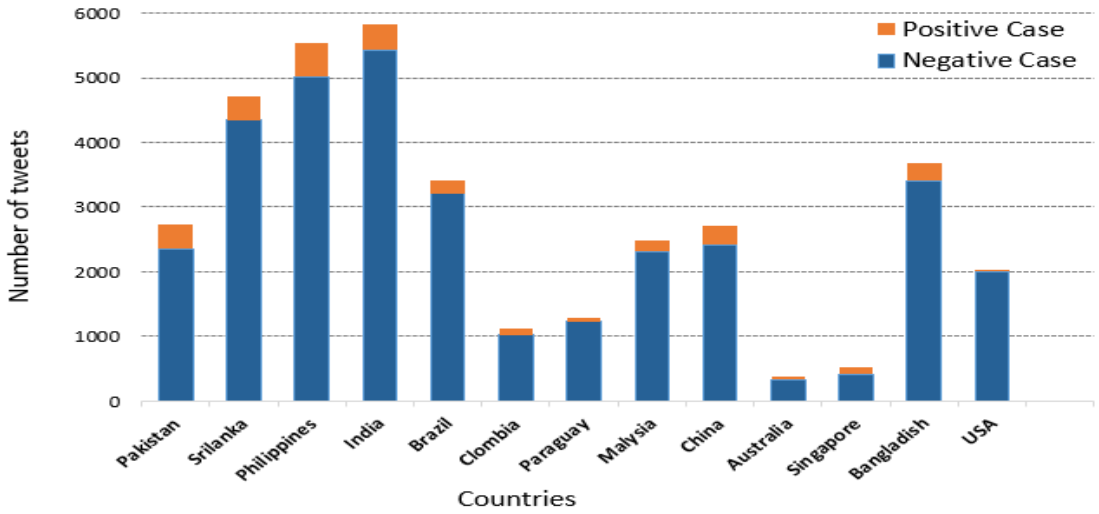


FIGURE 12: Region-wise detection of dengue positive cases

1) K-means Algorithm

In this section, clustering techniques like the k-means cluster algorithm is discussed [55]. There are some other clustering techniques like Density Clustering (i.e., DBSCAN) [56], and MeanShift algorithm [57], which includes cluster analysis in image processing and computer vision. Kmeans [58] is much efficient than DBSCAN as K-means utilizes distance-based measure to assess the similarities among the data sets. It is required that data be standardized to have a zero mean and a standard deviation of one, as almost all the parameters in any data set will have various measurement units. A cluster includes the collection of data sets that are compiled around each other due to certain commonalities.

i) Elbow Curve to determine the number of K-Means clusters

To verify (validate) the number of clusters the Elbow test approach is used. The concept of the elbow approach is to perform the k-means clustering on the data for several dimensions of k means (assume, K from 1 to 20) and also to measure the Sum of Squared Error (SSE) for each value of k. When k rises, the centroids are closer to the centroids groups. The enhancement would drop drastically at a certain point to generate an elbow shape. The value at that point is the optimum number for k value. As graphically visualized the elbow curve in Figure 15, which demonstrates that the level of the elbow curve off gradually after three clusters. Thus the optimal value for the number of clusters is (k =4). It suggests that adding more clusters is not going to benefit us that much.

ii) Clusters of Dengue Infected Regions via K-Means

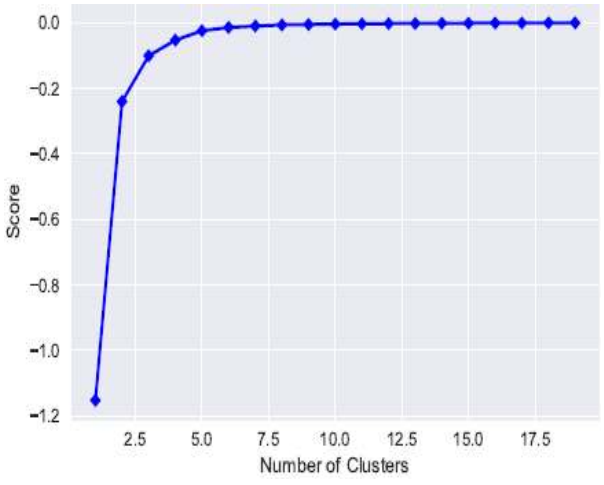


FIGURE 13: Elbow curve – to determine the number of k clusters

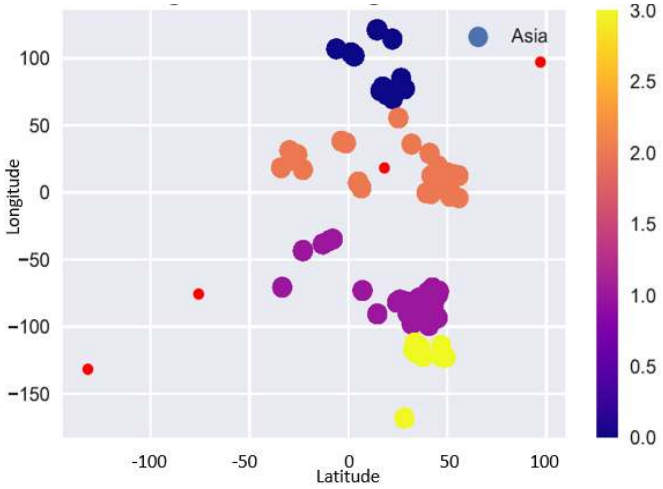


FIGURE 14: Clusters of dengue infected regions via K-Means

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3031174, IEEE Access

This subsection presents a graphical visualization of the dengue-infected areas by extracting latitude and longitude from users' profiles then the k-means clustering algorithm is applied. It can be seen in Figure 16, the infected areas are clustered using the K-means clustering algorithm. For this purpose, four clusters are generated using the elbow curve, which helps to compute the optimum number of clusters in the K-Means method (discussed above).

The K-Means clustering technique presented here for the geographical visualization of infected areas can be enhanced/improved. One promising approach is the enhancement of visualization of infected regions' names. For instance, to show the infected region name on the map at an alarming rate. Another drawback is that to identify the location of disease, the information regarding location data is rare with approximately 2% of texts comprising accurate geographical coordinates. These drawbacks to be explored in future work.

VI. CONCLUSION

With increasing popularity in the OSNSs platform, people use to share their opinions and transmit information. Using DL techniques, valuable features can be derived from this data and a different analysis can be conducted which assists us to promote the quality of life. In this paper, a novel approach has been developed that mines information from tweets whether a person is infected from disease or the tweet contains general information about a disease. The model has used LSTM with Word2Vec techniques. The experiment has demonstrated that the proposed approach outperforms the current state-of-the-art ML techniques. In future, we aim to evaluate the performance of LSTM with Glove and Fasttext. We also aim to utilize the CNN and LSTM-CNN approaches with different embedding and optimization techniques to analyze the epidemic outbreaks.

REFERENCES

[1] M. J. Paul *et al.*, "Social Media Mining for Public Health Monitoring and Surveillance," *Biocomput. 2016 Proc. Pacific Symp.*, pp. 468–479, 2016, doi: 10.1142/9789814749411_0043.

[2] A. Charalambous, "Social Media and Health Policy," *Asia-Pacific J. Oncol. Nurs.*, vol. 6, no. 1, pp. 24–27, 2019, doi: 10.4103/apjon.apjon.

[3] A. Khatua, A. Khatua, and E. Cambria, "A tale of two epidemics : Contextual Word2Vec for classifying twitter streams during outbreaks," *Inf. Process. Manag.*, vol. 56, no. 1, pp. 247–257, 2019, doi: 10.1016/j.ipm.2018.10.010.

[4] S. Ji, X. Li, Z. Huang, and E. Cambria, "Suicidal Ideation and Mental Disorder Detection with Attentive Relation Networks," *arXiv Prepr. arXiv*, pp. 1–15, 2020, [Online]. Available: http://arxiv.org/abs/2004.07601.

[5] S. A. Moorhead, D. E. Hazlett, L. Harrison, J. K. Carroll, A. Irwin, and C. Hoving, "A new dimension of health care: Systematic review of the uses, benefits, and limitations of social media for health communication," *J. Med. Internet Res.*, vol. 15, no. 4, pp. 1–16, 2013, doi: 10.2196/jmir.1933.

[6] T. Kendrick, M. Moore, S. Gilbody, R. Churchill, B. Stuart, and M. El-Gohary, "Routine use of patient reported outcome measures (PROMs) for improving treatment of common mental health disorders in adults," *Cochrane Database Syst. Rev.*, no. 7, 2016, doi: 10.1002/14651858.CD011119.

[7] A. Alessa and M. Faezipour, "A review of influenza detection and prediction through social networking sites," *Theor. Biol. Med. Model.*, vol. 15, no. 2, pp. 1–27, 2018, doi: 10.1186/s12976-017-0074-5.

[8] E. Lau *et al.*, "Twitter-Based Influenza Detection After Flu Peak via Tweets With Indirect Information : Text Mining Study," *J. Med. Internet Res.*, vol. 4, no. 3, pp. 1–27, 2019, doi: 10.2196/publichealth.8627.

[9] C. de A. Marques-Toledo *et al.*, "Dengue prediction by the web: Tweets are a useful tool for estimating and forecasting Dengue at country and city level," *PLoS Negl. Trop. Dis.*, vol. 11, no. 7, pp. 1–20, 2017, doi: 10.1371/journal.pntd.0005729.

[10] M. J. Paul and M. Dredze, "Social Monitoring for Public Health," *Synth. Lect. Inf. Concepts, Retrieval, Serv.*, vol. 9, no. 5, pp. 1–183, 2017, doi: 10.2200/s00791ed1v01y201707icr060.

[11] E. Cambria, A. Hussain, T. Durrani, C. Havasi, C. Eckl, and J. Munro, "Sentic computing for patient centered applications," *Int. Conf. Signal Process. Proceedings, ICSP*, pp. 1279–1282, 2010, doi: 10.1109/ICOSP.2010.5657072.

[12] E. Cambria, T. Benson, C. Eckl, and A. Hussain, "Sentic PROMs: Application of sentic computing to the development of a novel unified framework for measuring health-care quality," *Expert Syst. Appl.*, vol. 39, no. 12, pp. 10533–10543, 2012, doi: 10.1016/j.eswa.2012.02.120.

[13] T. U. M. Sepp Hochreiter and I. Jurgen Schmidhuber, "Long Short-Term Memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[14] N. Collier, N. T. Son, and N. M. Nguyen, "OMG U got flu? Analysis of shared health messages for bio-surveillance," *Anal. Shar. Heal. Messag. bio-surveillance. J. Biomed. Semant.*, vol. 2, no. 5, pp. 1–10, 2011.

[15] A. Alessa and M. Faezipour, "Preliminary Flu Outbreak Prediction Using Twitter Posts Classification and Linear Regression With Historical Centers for Disease Control and Prevention Reports : Prediction Framework Study," *Jmir Public Heal. Surveill.*, vol. 5, no. 2, pp. 1–17, 2019, doi: 10.2196/12383.

[16] E. Aramaki, M. Sachiko, and M. Morita, "Twitter Catches The Flu : Detecting Influenza Epidemics using Twitter The University of Tokyo The University of Tokyo National Institute of," *Proc. Conf. Empir. methods Nat. Lang. Process. Assoc. Comput. Linguist.*, pp. 1568–1576, 2011.

[17] A. Jimeno Yepes, A. MacKinlay, and B. Han, "Investigating Public Health Surveillance using Twitter," *Proc. Work. Biomed. Nat. Lang. Process.*, pp. 164–170, 2015, doi: 10.18653/v1/w15-3821.

[18] S. Wakamiya, Y. Kawai, and E. Aramaki, "Twitter-based influenza detection after flu peak via tweets with indirect information: Text mining study," *J. Med. Internet Res.*, vol. 20, no. 9, pp. 1–27, 2018, doi: 10.2196/publichealth.8627.

[19] K. Lee, A. Agrawal, and A. Choudhary, "Real-Time Disease Surveillance Using Twitter Data : Demonstration on Flu and

Cancer," *Proc. 19th ACM SIGKDD Int. Conf. Knwledge Discov. data mining, Chicago, Illinois, USA.*, pp. 1474–1477, 2013, doi: 10.1145/2487575.2487709.

[20] H. Xue, Y. Bai, H. Hu, and H. Liang, "Regional level influenza study based on Twitter and machine learning method," *PLoS One*, vol. 14, no. 4, pp. 231–253, 2019, doi: 10.1007/s10916-016-0545-y.

[21] A. H *et al.*, "Dengue Fever in Pakistan, Episodes of Epidemic to Endemic: Treatment Challenges, Prevention and Current Facts," *J. Bioequiv. Availab.*, vol. 09, no. 05, pp. 473–476, 2017, doi: 10.4172/jbb.1000347.

[22] J. Albinati, W. Meira, G. L. Pappa, M. Teixeira, and C. Marques-Toledo, "Enhancement of epidemiological models for dengue fever based on twiter data," *ACM Int. Conf. Proceeding Ser.*, pp. 109–118, 2017, doi: 10.1145/3079452.3079464.

[23] J. S. Coberly *et al.*, "Tweeting fever: Can twitter be used to monitor the incidence of dengue-like illness in the Philippines?," *Johns Hopkins APL Tech. Dig. (Applied Phys. Lab.*, vol. 32, no. 4, pp. 714–725, 2014.

[24] K. Espina, M. Regina, and J. E. Estuar, "Infodemiology for Syndromic Surveillance of Dengue and Typhoid Fever in the Philippines," *Procedia Comput. Sci.*, vol. 121, no. 1, pp. 554–561, 2017, doi: 10.1016/j.procs.2017.11.073.

[25] P. Guo, T. Liu, Q. Zhang, L. Wang, and J. Xiao, "Developing a dengue forecast model using machine learning: A case study in China," *PLoS Negl. Trop. Dis.*, vol. 11, no. 10, pp. 1–22, 2017, doi: 10.1371/journal.pntd.0005973.

[26] P. Muhilthini, B. S. Meenakshi, S. L. Lekha, and S. T. Santhanalakshmi, "Dengue Possibility Forecasting Model using Machine Learning Algorithms," *Int. Res. J. Eng. Technol.*, vol. 5, no. 3, pp. 1661–1665, 2018.

[27] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," *Proc. Int. Conf. Learn. Represent. (ICLR), Scottsdale, Arizona, USA*, pp. 1–12, 2013, [Online]. Available: http://arxiv.org/abs/1301.3781.

[28] A. Krizhevsky and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Adv. neural Inf. Process. Syst.*, pp. 1097–1105, 2012.

[29] J. Pennington, R. Socher, and C. Manning, "Glove: Global Vectors for Word Representation," *Proc. 2014 Conf. Empir. Methods Nat. Lang. Process. (EMNLP), Doha, Qatar*, pp. 1532–1543, 2014, doi: 10.3115/v1/D14-1162.

[30] A. Hernandez-Suarez, G. Sanchez-Perez, K. Toscano-Medina, H. Perez, J. Portillo, and V. Sanchez, "Using twitter data to monitor natural disaster social dynamics: A recurrent neural network approach with word embeddings and kernel density estimation," *Sensors*, vol. 19, no. 7, 2019, doi: 10.3390/s19071746.

[31] N. Alsaedi, P. Burnap, and O. Rana, "Can We Predict a Riot? Disruptive Event Detection Using Twitter," *ACM Trans. Internet Technol.*, vol. 17, no. 2, pp. 1–26, 2017, doi: 10.1145/2996183.

[32] Y. Tay, L. A. Tuan, S. C. Hui, and J. Su, "Reasoning with Sarcasm by Reading In-between," *Proc. 56th Annu. Meet. Assoc. Comput. Linguist. (Long Pap. Melbourne, Aust.*, pp. 1010–1020, 2018, [Online]. Available: http://arxiv.org/abs/1805.02856.

[33] M. Zhang, Y. Zhang, and G. Fu, "Tweet Sarcasm Detection Using Deep Neural Network," *Proc. COLING 2016, 26th Int. Conf. Comput. Linguist. Tech. Pap.*, pp. 2449–2460, 2016.

[34] O. Bark, A. Grigoriadis, J. A. N. Pettersson, V. Risne, A. Siitova, and H. Yang, "A deep learning approach for identifying sarcasm in text," *Master's Thesis*, pp. 1–67, 2017.

[35] A. Joshi, V. Tripathi, K. Patel, P. Bhattacharyya, and M. Carman, "Are Word Embedding-based Features Useful for Sarcasm Detection?," no. 2013, 2016, [Online]. Available: http://arxiv.org/abs/1610.00883.

[36] O. Ajao, D. Bhowmik, and S. Zargari, "Fake News Identification on Twitter with Hybrid CNN and RNN Models," *Proc. Int. Conf. Soc. Media Soc. (SMSociety), Copenhagen, Denmark*, pp. 226–230, 2018.

[37] B. Jang, I. Kim, and J. W. Kim, "Word2vec convolutional neural networks for classification of news articles and tweets," *PLoS One*, vol. 14, no. 8, pp. 1–20, 2019, doi: 10.1371/journal.pone.0220976.

[38] M. L. McHugh, "Interrater reliability : the kappa statistic," *Biochem. Medica*, vol. 22, no. 3, pp. 276–282, 2012, [Online]. Available: https://hrcak.srce.hr/89395.

[39] J. L. Fleiss, B. Levin, and M. C. Paik, "The Measurement of Interrater Agreement," *Stat. Methods Rates Proportions*, pp. 598–626, 2004, doi: 10.1002/0471445428.ch18.

[40] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.

[41] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A Critical Review of Recurrent Neural Networks for Sequence Learning," *arXiv Prepr. ariv*, pp. 1–38, 2015, [Online]. Available: http://arxiv.org/abs/1506.00019.

[42] C. P. Medina and M. R. R. Ramon, "Using TF-IDF to Determine Word Relevance in Document Queries," *Proc. first Instr. Conf. Mach. Learn. Piscataway, NJ USA*, pp. 133–142, 2003, doi: 10.15804/tner.2015.42.4.03.

[43] Q. Le and T. Mikolov, "Distributed Representations of Sentences and Documents," in *Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32*, 2014, vol. 32, pp. II–1188--II–1196, doi: 10.1145/2740908.2742760.

[44] B. Chiu, "How to Train Good Word Embeddings for Biomedical NLP," in *Proceedings ofthe 15th Workshop on Biomedical Natural Language Processing*, 2016, pp. 166–174.

[45] T. Mikolov, Q. V. Le, and I. Sutskever, "Exploiting Similarities among Languages for Machine Translation," *arXiv Prepr. arXiv*, vol. 1309, no. 4168, pp. 1–10, 2013, [Online]. Available: http://arxiv.org/abs/1309.4168.

[46] T. Fushiki, "Estimation of prediction error by using K-fold cross-validation," *Stat. Comput.*, vol. 21, no. 2, pp. 137–146, 2011, doi: 10.1007/s11222-009-9153-8.

[47] R. Kohavi, "A Study of Cross?Validation and Bootstrap for Accuracy Estimation and Model Selection," *Appear. Int. Jt. Conf. Artifcial Intell. IJCAI*, vol. 14, no. 2, pp. 1137–1145, 1995, doi: 10.1067/mod.2000.109032.

[48] S. K and S. S, "Review on Classification Based on Artificial Neural Networks," *Int. J. Ambient Syst. Appl.*, vol. 2, no. 4, pp. 11–18, 2014, doi: 10.5121/ijasa.2014.2402.

[49] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986, doi: 10.1038/323533a0.

[50] T. U. M. Sepp Hochreiter, "The vanishing gradient problem during learning recurrent neural nets and problem solutions.," *Int. J. Uncertainty, Fuzziness Knowledge-Based Syst.*, vol. 9, no. 02, pp. 107–116, 1998.

[51] Y. Bengio, P. Simard, and P. Frasconi, "Learning Long-Term Dependencies with Gradient Descent is Difficult," *IEEE Trans. Neural Networks*, vol. 5, no. 2, pp. 157–166, 1994, doi: 10.1109/72.279181.

[52] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling," pp. 1–9, 2014, [Online]. Available: http://arxiv.org/abs/1412.3555.

[53] Colah, "Understanding LSTM Networks," *Available online*, 2015. https://colah.github.io/posts/2015-08-Understanding-LSTMs/ (accessed Feb. 26, 2020).

[54] D. M. W. Powers, "Evaluation: From Precision, Recall and F-Measure To Roc, Informedness, Markedness & Correlation," *J. Mach. Learn. Technol.*, vol. 2, no. 1, pp. 37–63, 2011.

[55] X. Zhou *et al.*, "An automatic K-Means clustering algorithm of GPS data combining a novel niche genetic algorithm with noise and density," *ISPRS Int. J. Geo-Information*, vol. 6, no. 12, 2017, doi: 10.3390/ijgi6120392.

[56] T. Wang, C. Ren, Y. Luo, and J. Tian, "NS-DBSCAN: A density-based clustering algorithm in network space," *ISPRS Int. J. Geo-Information*, vol. 8, no. 5, 2019, doi: 10.3390/ijgi8050218.

[57] G. G. Godinho *et al.*, "Mean Shift, Mode Seeking, and Clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 7, no. 8, pp. 1–10, 1995, doi: 10.1016/j.rbo.2014.03.007.

[58] K. R. Zalik, "An efficient k-means clustering algorithm," *Pattern Recognit. Lett.*, vol. 29, no. 9, pp. 1385–1391, 2008, doi: 10.1016/j.patrec.2008.02.014.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2020.3031174, IEEE Access

Samina Amin received the M.Sc. degree in computer science from the Institute of Computing, Kohat University of Science and Technology, Kohat, Pakistan, where she is currently pursuing the M.S. degree. Her research interests include machine learning, deep learning, data science, natural language processing, and social media analysis. She has received the Gold Medal from Kohat University of Science and Technology.

M. Irfan Uddin has been actively involved with academia and research. He was a Research Associate with University of Peshawar, Pakistan, University of Amsterdam, the Netherlands and University of Turin, Italy. He was an Assistant Professor at Al Yamamah University, Riyadh, Saudi Arabia. He is currently working at the Institute of Computing, Kohat University of Science and Technology, Kohat, Pakistan. His research interests include machine learning, data science, deep learning, convolutional neural networks, reinforcement learning, computer vision, and parallel programming. He has published several articles in reputed journals and conference proceedings. He serves as a reviewer for different journals.

M. Ali Zeb is currently working at the Institute of Computing, Kohat University of Science and Technology, Kohat, Pakistan. His research interests include machine learning, deep learning, social media analysis, natural language processing, sentiment analysis, data science, visualization, topic modelling, and neural networks.

Ala Abdulsalam Alarood is currently working at the College of Computer Science and Engineering, University of Jeddah, Kingdom of Saudi Arabia. His research interests include machine learning, data visualization, and data science.

Marwan Mahmoud is an Assistant Professor at King Abdulaziz University, Jeddah, Saudi Arabia. He received his B.Sc. and M.Sc. degrees in Electrical and Computer Engineering from King Abdulaziz University, Jeddah, Saudi Arabia, in 2003, and 2008 respectively, and his Ph.D. degree in Electrical and Computer Engineering from the Western University, Canada in 2016. His research interest includes networks, IoT, cloud computing, and information security. Dr. Marwan worked for more than 3 years in the Technical and Vocational Training Corporation in Saudi Arabia.

Monagi H. Alkinani received his PhD. in Computer Science in 2017 from Western University, London, Canada. In 2018, he joined the Deanship of Scientific Research at the

University of Jeddah, Saudi Arabia, where he served as the vice-dean of research. He holds an assistance professor position at the Department of Computer Science and Artificial Intelligence and is a member of the Jeddah Computer Vision Team, where he supervises research activities and teaches image processing and artificial intelligence to bachelor students as well as signal processing to master students in computer science. At the Deanship, he supervised research in the field of computer vision. He has been involved in many collaborative research projects financed by various instances including the Ministry of Education and the University of Jeddah. He currently serves as coordinator of the R&D program for the Ministry of Education.