

Depression Classification using Harris Hawk Optimization (HHO) based Recurrent Fuzzy Neural Network (FRNN) for Sentiment Analysis: An Applied Nonlinear Analysis

¹ Mrs. P. Suganya, ² Dr. G. Vijaiprabhu, ³ Dr. N. Shanmugapriya, ⁴ Dr. M Praneesh, ⁵ Dr. M. Menaka, ⁶ Dr. K. Sathishkumar

¹ Ph.D Research Scholar, PG and Research Department of Computer Science, Erode Arts and Science College (Autonomous) Erode, Tamilnadu, India. Email: suganya.rithikjos83@gmail.com

² Assistant Professor, PG and Research Department of Computer Science, Erode Arts and Science College (Autonomous) Erode, Tamilnadu, India. Email: gvprabhu7@gmail.com

³ Assistant Professor & Head, Department of Information Technology, Erode Arts and Science College (Autonomous) Erode, Tamilnadu, India. Email: sweetypriya26@gmail.com

⁴ Assistant Professor, Department of Computer Science with Data Analytics, Sri Ramakrishna College of Arts & Science, Coimbatore, Tamilnadu, India. Email: raja.praneesh@gmail.com

⁵ Assistant Professor & Head, Department of Computer Technology, Erode Arts and Science College (Autonomous) Erode, Tamilnadu, India. Email: menarameac@gmail.com

⁶ Assistant Professor, PG and Research Department of Computer Science, Erode Arts and Science College (Autonomous) Erode, Tamilnadu, India. Email: sathishkumark@iecee.org

Article History:

Received: 22-03-2024

Revised: 12-05-2024

Accepted: 22-05-2024

Abstract:

The research introduces a novel approach for sentiment analysis by combining Harris Hawk Optimization (HHO) with a Recurrent Fuzzy Neural Network (FRNN). Sentiment analysis, crucial in natural language processing, often struggles with the complexities of language. The proposed methodology addresses these challenges by leveraging HHO's adaptability and FRNN's temporal modeling capabilities. FRNN, with recurrent connections, effectively captures temporal dependencies in sequential data, making it suitable for sentiment analysis. Additionally, fuzzy logic in FRNN handles linguistic uncertainty, enhancing model robustness. The methodology involves preprocessing tweets, feature extraction using techniques like Latent Semantic Analysis (LSA), Independent Component Analysis (ICA), and lexicon-based analysis. Feature selection methods like PMI, chi-square, and DFS are employed to reduce dimensionality. The sentiment classification utilizes Kernel Extreme Learning Machine (KELM), where KELM's kernel function eliminates the need for setting hidden node size. The sentiments are then detected using the HHO-based FRNN, optimizing weight parameters for classification accuracy. HHO's hunting phases aid in the convergence of the algorithm, reducing computational complexity. Experimental evaluations demonstrate superior performance over existing techniques, offering a promising avenue for sentiment analysis in natural language processing research.

Keywords: Depression Classification, Sentiment Analysis, Recurrent Fuzzy Neural Network, Harris Hawk Optimization, Machine Learning, Mental Health, Applied Nonlinear Analysis.

1. Introduction

Twitter data mining is the process of discovering high quality information or knowledge from the huge volume of unstructured data. It is also known as Twitter Data Mining [1]. Unstructured data consists

of text, charts and multimedia information. There is no common format for unstructured data and it does not fall into any predefined representation [2, 3]. It generates many irregularities and uncertainties hence it is hard for a computer to understand. It deals with the natural language text that is available either in the form of semi-structured or unstructured and it is a multidisciplinary field which incorporate and integrate the various domains like, data mining, machine learning, information retrieval, statistics, natural language processing and computational linguistics [4].

Sentimental Analysis is the task of determining whether the given piece of text is positive, negative, or neutral [5]. A sentimental analysis system integrates natural language processing and machine learning techniques to assign a probabilistic score to each object, topic, sentiment, theme, or class. Data analytics industries often incorporate third-party application interfaces into their customer experience management system to offer users insights to their end-users [6, 7]. The sentimental analysis conducted in the tweets is a straightforward process. Initially, each text tweet is broken down into different components (sentences, phrases, tokens, and parts of speech). The next step is to identify the sentiments present in each component. After that, the sentiment score is assigned to each component, and these scores are integrated with conducting a multi-layered sentimental analysis [8].

Sentiment or opinion Analysis (SA) is one of the ongoing trending topics in NLP considering the fact that before 2000. Goal of opinion extraction is to specify automatic system possible to pull out the subjective statistics from the text, including evaluations, sentiments, that allows you to make primarily based and knowledge utilized in decision guide system [9]. Sentiment analysis (SA) is a computational manner to extract opinion as powerful or bad from text approximately an entity. Unsurprisingly, there was some dilemma between the researchers about the distinction among sentiment as well as opinion, therefore there is an argument whether it sphere need to be referred to as sentiment assessment or opinion assessment [10].

This research presents a novel approach for sentiment analysis leveraging a hybrid framework combining Harris Hawk Optimization (HHO) with a Recurrent Fuzzy Neural Network (FRNN). Sentiment analysis is critical in deciphering subjective information from text data, yet conventional methods often struggle with the inherent complexities of natural language. By integrating the adaptability of HHO with the temporal modeling capabilities of FRNN, our methodology adeptly addresses these challenges. Through recurrent connections, the FRNN effectively captures temporal dependencies in sequential data, making it well-suited for sentiment analysis tasks. Additionally, the incorporation of fuzzy logic enables the FRNN to handle linguistic uncertainty inherent in natural language expressions, enhancing model robustness and interpretability. Experimental evaluations on standard sentiment analysis datasets demonstrate significant performance gains over state-of-the-art techniques in terms of accuracy, precision, recall, and F1-score metrics. Moreover, the interpretability facilitated by fuzzy logic enhances transparency, providing deeper insights into sentiment patterns within textual data. This research underscores the potential of the proposed HHO-based FRNN framework to advance sentiment analysis capabilities, offering a promising avenue for future research in natural language processing and computational intelligence.

2. Related Works

The main aim of the Support Vector Machine (SVM) is to train the decision function to accurately classify the new data instances present in the input labeled dataset. Wen Long et al. (2018) [11] presented an Universum SVM to use neutral samples (Universum) to analyze the investor sentiments. The Universum is the samples that don't belong to the positive or negative classes and stand as neutral. The Universum comprises prior knowledge for classification, and the additional information obtained can be used to retrieve better results. Using the Universum samples, designed a two-class (positive and negative) and three-class problem (positive, negative, and neutral). The dataset of this paper was obtained from the east money stock forum china, which comprises the individual investor sentiment review. The positive and negative samples are represented as the bullish and bearish samples in this work. The dataset consists of 1010 bullish samples, 1212 bearish samples, and 3768 neutral samples. However, the dataset used in their proposed methodology is less abundant due to the limited resource constraints, and the 3 class classification is still in its empirical form.

The main aim of the document level classification is to predict the overall user sentiments in a single document. A document usually comprises different opinions and sentiments for a particular aspect. The overall sentiment present in the document is important in predicting sentiments at the document level. Since the state-of-art techniques treat every sentiment present in the document equally, it is necessary to consider the overall sentiments of different aspects. To overcome this problem, Xiaojia Pu et al. (2019) proposed a structural SVM technique to efficiently identify the overall opinion sentence present in the text document. The appropriate overall opinion sentences are identified by integrating multiple features. The overall opinion sentences are considered as the hidden variables of the structural SVM. The methods are evaluated in five different real-time datasets such as Oscar data, Mcauley data, Lui data, IMDB(S), and IMDB (L) data. The proposed model offers an accuracy of 0.765, 0.896, 0.932, 0.8830, and 0.9011 for the Oscar data, Lui data, Mcauley data, (S), and IMDB (L) datasets [12].

Mohammad AL-Smadi et al. (2018) conducted an Aspect based sentimental analysis in the Arabic hotel reviews using the SVM technique. The features are extracted using the stemming and Bag of Words approach. This work also takes into account the set of morphological, syntactic, and semantic features. The SVM technique outperforms the deep learning technique in terms of aspect class identification, aspect sentiment polarity identification, and aspect opinion target expression identification. However, the training and testing time of the model is higher when compared with the Recurrent Neural Network (RNN) [13].

The ANN mainly extracts the features using a linear combination of input data and processes it to yield a nonlinear function comprising these features. It mainly imitates the human brain in processing information. The ANN is mainly trained using gradient descent techniques. Rodrigo Moraes et al. (2013) presented an ANN model for document-level sentimental analysis. The experimentation is conducted on four different datasets: movie reviews, GPS, books, and cameras. Every dataset comprises 2000 reviews which are either classified as positive or negative. The ANN classifier classifies the reviews with more than 3-star ratings as positive and reviews below 3-star ratings as negative. However, the proposed technique offers low performance for the unbalanced dataset [14].

Zeeshan Shaukat et al. (2019) presented an ANN model for identifying the sentiments present in the movie review and provide a rating along with it. This helps one identify the different parts of the movie and the reviewer’s opinions. The proposed model has been trained using the Movie review database and provided a final accuracy of 91%. Kalaraniet al. (2019) presented an accurate segmentation model using the ANN for analyzing movie reviews. They applied the model for both balanced and unbalanced datasets. The POS tagging approach is used for extracting the unigrams and bi-tagged features. Using relevance and redundancy, the feature information is computed and provided as the input to the ANN. The proposed model gives an accuracy of 81.5% in the balanced movie review dataset and gives an accuracy of 88% in the unbalanced movie review dataset [15, 16, 17].

3. Proposed Methodology for Sentiment Analysis from tweet

Sentiment analysis, a crucial task in natural language processing, aims to extract subjective information from text data, discerning the sentiment conveyed within. Traditional sentiment analysis approaches often encounter challenges with handling the complexities of natural language, including ambiguity, context dependence, and linguistic variations. To address these issues, this research proposes a novel framework integrating Harris Hawk Optimization (HHO) with a Recurrent Fuzzy Neural Network (FRNN) for sentiment analysis tasks. This work analysis the Kernel Extreme Learning Machine (KELM) for detection. Before detection, the input TWEET needs to be preprocessed using stemming, stop word removal, tokenization, Part of speech (POS) tagging, and microblogging before detection.

The features from the preprocessed data are extracted using Latent Semantic Analysis (LSA), Independent Component Analysis (ICA), and Lexicon-based features analysis. Next, the dimension of extracted features is reduced using a Chi-square, Point-wise Mutual Information (PMI), and Distinguishing Feature Selector (DFS) approaches. Then, use these extracted feature vectors as input for training and testing the detection algorithm. The process flow of the proposed sentiment analysis approach is shown in Figure 1. Moreover, detecting the sentiments from the classified TWEET is necessary, which evaluates the emotions from the classified TWEET.

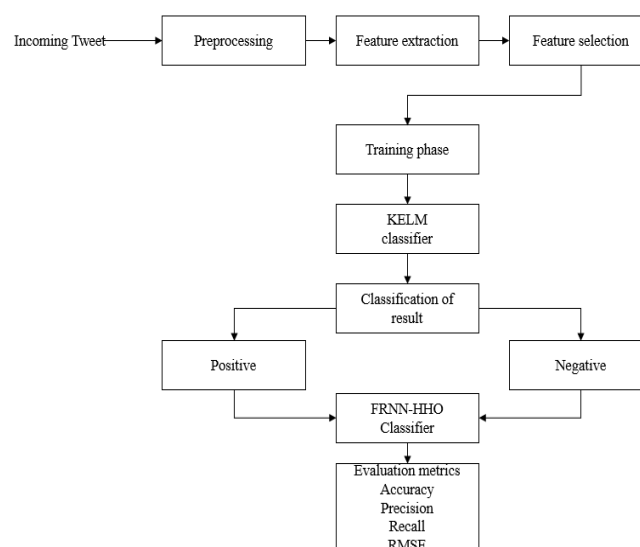


Figure 1: Flowchart for Proposed Sentiment Classification using Optimization based fuzzy RNN

An optimization-based deep learning approach identifies the sentiments; present in the detected TWEET. The Fuzzy Recurrent Neural Network (FRNN) is used for this sentiment analysis. A metaheuristic optimization algorithm Harris Hawk Optimization (HHO) is a hybrid algorithm along with the proposed neural network to maximize the classification accuracy. The weight parameter is optimized to the optimal or near-optimal solution using the HHO algorithm. This merit makes it attain the fastest convergence rate. The training procedure carried out in FRNN intends to decide the connection weights among several neurons for error reduction. The dataset does not contain any sentiment labels, therefore to apply label for that dataset need a Text blog based sentiment analysis approach. Noises. The preprocessing involves some steps to classify the input Short Message Service using several methods. The four steps are: Tokenization, Stopping, Stemming, and Part of speech (POS). Each of these steps is explained in detail in Chapter 3. Features are extracted using Latent Semantic Analysis (LSA), Independent Component Analysis (ICA), and lexicon-based features extraction.

Feature Extraction

Feature extraction is a process that converts a set of input data into its corresponding features. It is considered an essential step in text processing as it directly affects different clusters or classes. Moreover, identifying effective features from unstructured data is considered a difficult task. Two different categories of feature extraction techniques are introduced in this framework, one is for text classification, and the next one is for sentiment classification. The details about the two categories of feature extraction are discussed below.

Latent Semantic Analysis (LSA)

Latent Semantic Analysis is defined as the characteristic of algebraic statistical methods, and it removes the hidden structure from the words or sentences. The LSA was also known as Singular Value Decomposition (SVD) introduced. It identifies the unstructured data hidden in the input document and identifies the relationship between the words or sentences. SVD has able to reduce noise and increase efficiency. LSA undertakes the words were nearer to the meaning and occurs in similar pieces of text. SVD is used to perform the mathematical technique in LSA to optimize the length of the text.

The LSA consists of four main steps which are described below.

- Term document matrix is a collection of large text have detached words into smaller units of passage or sentence for each application.
- Transformed term document matrix as a replacement of operational with the rare term frequency, the accesses of term document matrix were often changed. Hence, it obtains the frequency in a sub linear fashion, $\log(fr_{ij} + 1)$.
- Dimension reduction it reduces the rank of the matrix by using SVD. Assume that L is taken as the largest singular value and the remainder is set to zero. The SVD technique is also closely interrelated with some features such as Eigen analysis, Factor analysis, Principal components analysis, and linear neural networks.

- Retrieval in reduced space it is used to reduce the space in term-document matrix. For example; document, term, document term these words represented in the same space so they need to compute.

The matrix of SVD was represented as X. The mathematical view of LSA is detailed as follow.

$$X = T^* S^* D^T \text{-----}(1)$$

Where T* and S* are represented as an orthonormal matrix, D^T was represented as a diagonal matrix. The problematic representation of X was using orthogonal dimension. The SVD uses the largest L singular value to optimize the dimensions was used in LSA.

$$X = T_L^* S_L^* D_L^* \text{-----}(2)$$

where, T_L term vectors in LSA, D_K Document vectors in LSA.

Independent Component Analysis (ICA)

The aim of ICA is used to identify the missing text and valuable information from the input document. The ICA is used to separate the neural method words; were varied in an unidentified way. The ICA also includes linear transformation methods such as principal component analysis, factor analysis, and projection pursuit. The problematic description of ICA was detected in two forms defined in Equation (3).

$$x_1(t) = a_{11}s_1 + a_{12}s_2 \text{-----}(3)$$

$$x_2(t) = a_{21}s_1 + a_{22}s_2 \text{-----}(4)$$

Where, a_{11} , a_{12} , a_{21} and a_{22} are some of the parameters to find the distance of the signal. and are used to estimate the duplicate text, $s_1^{(t)}$ and $s_2^{(t)}$ estimate original text. The matrix of ICA is denoted as A. Here the lower case of bold letters indicates vector and upper case bold letters indicate matrix.

$$X = As \text{-----}(5)$$

"A" is the column of a matrix. Sometimes the matrix is denoted as a_j and the model is rewritten as Equation (6)

$$x = \sum_{i=1}^n a_j s_j \text{-----}(6)$$

The ICA is a generative model in which the data were generated with the mixing components of s_j .

Lexicon-Based Feature Extraction

In this section, the information about the Domain Specific Emotion Lexicon (DSEL) that is used for extracting the range of features suitable for emotion classification is discussed. The feature vectors that are obtained using the lexicon knowledge are mostly of length |E|, where |E| indicates the total number of emotion classes present in the available dataset. The following features are considered to represent the emotions in the document.

Total Emotion Count (TEC)

It is a kind of feature extraction that captures the emotion-based words from the document. The feature vector corresponding to the word in the given document is represented as d_{TEC} . For emotion j , the feature value is computed using Equation (7).

$$d_{TEC}[e_j] = \sum_{\omega \in I} I(e_j = \text{argmax} Lex_k(w, k) \times \text{cont}(w, d)) \text{-----}(7)$$

The indicator function which is set as either 0 or 1 (for true or false arguments). The total number of times the word w occurred in document d is represented as $\text{count}(w, d)$. TEC captures the features that are suggested by the lexicon alone (i.e., the features showing the highest value in the lexicon). However, the emotions shared by each word in the document are not similar. For example, the word beautiful is subjected to both love and joy, whereas the TEC provides a count of 1 for one class and 0 for another class.

Total Emotion Intensity (TEI)

The sum of the scores from the emotional intensity of each word in the document is obtained by TEI. Generally, TEC use coarse integer counts, but TEI utilizes emotion intensity scores obtained from DSEL to extract the emotion-based features from the document having a number of emotion classes. The feature vector extracted by TEI from the document d is represented as d_{TEC} . For emotion j , the feature value is computed using Equation (8).

$$d_{TEC}[e_j] = \sum_{\omega \in L} Lex(w, e_j) \times \text{cont}(w, d) \text{-----}(8)$$

Max Emotion Intensity (MEI)

Instead of identifying the average score, identifying the term that shows the highest sentiment bearing to the sentiment class from the entire document is considered the major MEI concept. Therefore, with MEI, the intensity score of the word having the highest emotion bearing from the given document is estimated. The feature vector extracted by MEI from the document d is represented as d_{MEI} . For emotion j , the feature value is computed using Equation (9).

$$d_{MEI}[e_j] = \text{arg}_{w \in d} \max Lex(w, j) \text{-----}(9)$$

Graded Emotion Count (GEC)

The concept of high-intensity emotional words is extended to extract the document representation. For such extraction, the TEI and TEC variants are developed. Both the variants considered all the words in the document without concerning about their emotional intensity. However, understanding the effect of high-intensity words on emotion classification is valuable to achieve better performance. The processing principle of both GEC and TEC are similar, except for the fact that it extracts the total number of words and their respective emotions within a threshold value θ from the document. It quantifies the relation between the emotion and its respective classes in the probability distribution form. The resultant intensity scores lie between the intervals 0 and 1. In this work, the available interval is divided into 4 different quartiles they are (0, 0.25), (0.25, 0.5), (0.5, 0.75) and (0.75, 1). Out of these four intervals, three values are selected as threshold values they are 0.25, 0.5, and 0.75. The features that are extracted using GEC are mostly for these three intervals. The feature vector extracted

by GEC from the document d is represented as d_{GEC} . For emotion j , the feature value is computed using Equation (10).

Graded Emotion Intensity (GEI)

Like GEC, develop a variant of TEI, named GEI, which is estimated by integrating the words intensity scores found within the fixed threshold from the given document d . The thresholds mentioned earlier are used for extracting GEI features using DSLs. Given a document d , and its corresponding feature vector d_{GEI} , the feature value for the j th emotion is computed as follows.

$$d_{GEI}|e_j| = \sum_{\omega \in e} Lex(w, e) \times cont(w, d) \text{-----}(10)$$

Feature Selection

Feature selection is a process that intends to reduce the number of input variables by selecting only the variables that are useful for the classification process. The main merit of this feature selection is that it reduces the complexity of the learning algorithm: furthermore, it increases processing speed and accuracy. In this work, used three different and efficient feature selection approaches PMI, chi square, and DFS.

Chi-Square

Chi square is a filter based feature selection approach that determines whether the occurred feature is class dependent or independent. Large values obtained by chi-square represent that both the class and feature are found independent.

$$X^2 = \sum \left(\frac{F_0 - F_e}{F_e} \right)^2 \text{-----}(11)$$

Where, expected and observed frequency for each class and feature is represented as F . and F_0 respectively.

Distinguishing Feature Selection

Distinguishing feature selection is an ideal filter based feature section approach that assigns a high score for distinctive features and a low score for irrelevant features. It ranks the terms based on four different conditions they are:

If a term is present in one class and not found in any other classes, then consider such term as distinctive and assign a high score. If a term rarely present in one class and does not found in other classes, consider that terms as irrelevant and assign a low score.

- If a term is frequently found in all classes, then consider such a term as irrelevant and assign a low score.
- Retrieval in reduced space it is used to reduce the space in term-document matrix. For example; document-document, term-term, document-term these words represented in the same space so they need to compute.
- If a term occurs in few classes, then consider such term as relatively distinctive and assign a relatively high score.

The formula applied by DFS for feature selection is given in Equation (12).

$$DFS(t) = \sum_{a=1} \frac{P(C_a \setminus t)}{P(t \setminus C_a) + P(t \setminus C_a) + 1} \quad \text{-----}(12)$$

Where, the conditional probability obtained for term that is found in classes other than Co is represented as $P(C_a \setminus t)$.

Point-wise Mutual Information (PMI)

The Point-wise mutual information approach determines the relationship between the two features. A high PMI value indicates the frequent co-occurrence of two features. A feature selection process is extensively applied to identify the mutual information shared among the terms and particular classes. PMI determines the ratio between the estimated co-occurrence for term t, and class Ca which is defined in Equation (13).

$$PMI(C_a, t_j) = \log \frac{F(t_j)P(C_a, t_j)}{F(t_j)P(C_a)} = \log \left(\frac{P(C_a, t_j)}{P(C_a)} \right) \text{-----}(13)$$

The features selected by three filter-based techniques are then provided to KELM sentiment for classification. The selected features may contain sentiment -based features. Sometimes it carries sentiment words; therefore, using sentiment-based feature, it is essential for the KELM classifier to accurately classify the tweet.

Kernel Extreme Learning Machine (KELM) based Classification

Extreme Learning Machine (ELM) is a type of Single hidden Layer Feed-forward Neural Network (SLFNN) whose architecture is depicted in Figure 2. Three different layers are present in ELM; input, hidden, and output layers. The usage of the nonlinear activation function makes the hidden layer nonlinear, whereas the output is linear as it does not comprise any activation function. The ELM fails to attain better results in a few cases due to the random selection of bias and weight parameters between the input and hidden layers. To overcome such limitation, the kernel function is introduced in the ELM approach, which eliminates the weight initialization procedure in the input and hidden layers by including the kernel matrix. KELM shows some merits; while using KELM, the determination of hidden layer size is not necessary. Moreover, it effectively explores the non-linear features. But, selecting the best kernel function is considered the major contribution of KELM during classification. Let x represents the total training samples, the output from the neural network is represented as f(x). The SLFNN having i hidden nodes is represented using the Equation (14).

$$f_{ELM}(x) = W^T \times K(w, b, x) \text{-----}(14)$$

Where, the activation function of the hidden layer is indicated as K (w, b, x), indicates the bias weight of hidden layer, weight obtained between the hidden and output layer is denoted as $W=[\beta_1, \beta_2, \dots, \beta_3]$ and the input weight connecting the input and hidden layer is denoted as w. The output from the hidden layers and the ELM model is represented in Equation (15).

$$f_{ELM}(x) = h(x)H^T \left(\frac{I}{P} + HH^T \right) T \text{-----}(15)$$

The output of the KELM model is represented in Equation (16).

$$f_{KELM}(x) = h(x)H^T \left(\frac{I}{P} + \lambda \right) \text{-----}(16)$$

Where, $\frac{1}{p}$ represents the regularization parameter and T represents the target output. KELM is found better than ELM as it produces less computational time, whereas it does not contain any random feature mappings. Further, the usage of the kernel in ELM eliminates the need of setting the desired number of hidden nodes. Next, the sentiments present in and ham need to evaluate to identify each TWEET's sentimental intention. Therefore, to identify such sentiments from classified TWEET, the optimization hybrid FRNN is used. It takes the selected features as input and determines the sentiments (positive, negative, and neutral) from the and ham TWEET. The architecture of ELM.

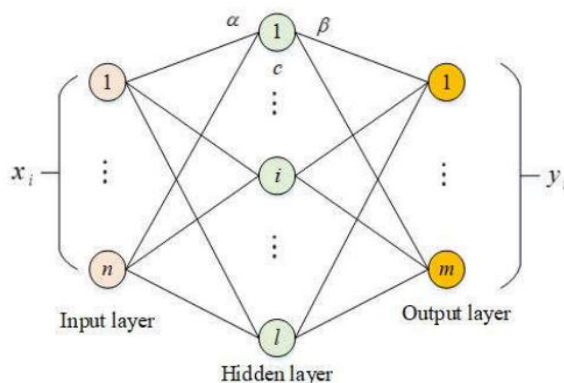


Figure 2. Architecture of ELM

Harris Hawk Optimization (HHO) based Re- current Fuzzy Neural Network (FRNN) for Sentiment Analysis

Fuzzy Recurrent Neural Network (FRNN)

Fuzzy Neural Network is extensively used in various fields, among that FRNN is identified as the notable one. FRNN contains four different layers they are input layer, membership layer, fuzzy layer, and output layer. The input and output provided to the node i of the layer k is represented as $o_i^{(n)}$ and $u_i^{(k)}$. The procedure for FRNN is discussed below:

Layer 1: Initial layer is the input layer which contains N nodes and a parameter has resembled in each layer. The features that are selected using the PMI, DFS, and chi-square are given as an input in this first layer.

$$O_i^{(1)} = u_i^{(1)} = x_i, \text{ where, } i = 1 \div M \text{-----(17)}$$

Layer 2: The second layer is the membership layer. In this layer, the gauss function is used as a membership function which changes the data in the nodes of the second layer in crisp form. The neural node in this second layer is denoted as N x M, where M represents the fuzzy rules. Three parameters, o_{ij} , ij and m_{ij} are included in every node

$$O_{ij}^{(2)} = \exp \left[-\frac{(o_{ij}^{(2)} - m_{ij})^2}{\sigma} ij \right], \text{ where, } i = 1 \div N, j = 1 \div M \text{-----(18)}$$

Where, the variance and centre corresponding to the Gauss distribution function is represented as o_{ij} and m_{ij} .

$$u_{ij}^{(2)}(t) = O_i^1 + \theta_{ij} O_{ij}^2(t-1), \text{ where, } i = 1 \div N, j = 1 \div M \text{-----(19)}$$

Where, the weight corresponding to the recurrent nodes is represented as o_{ij} . In this second layer, the factor $O_{ij}^2(t-1)$ is included at each input nodes. The previous learning procedure estimates the residual data which is illustrated by the introduced factor at this second layer. Then, based on Equation (18) replace $u_{ij}^{(2)}$ from Equation (19) to obtain Equation (20).

$$O_{ij}^{(2)} = \exp \left[- \frac{[x_i(t) + \theta_{ij} O_{ij}^2(t-1) - m_{ij}]^2}{(\sigma_{ij})^2} \right] \text{-----(20)}$$

Layer 3: In the third layer, fuzzy rules are used. Layers 3 and 4 are concatenated to conclude the available fuzzy rules. Each node in this third layer is resembled with AND operation. The expression for each AND operation is given in Equations (21, 22).

$$O_j^{(2)} = \prod_{i=1}^n O_{ij}^2 \text{-----(21)}$$

$$O_j^{(2)} = \prod_{i=1}^N \exp \left[- \frac{[x_i(t) + \theta_{ij} O_{ij}^2(t-1) - m_{ij}]^2}{(\sigma_{ij})^2} \right] \text{ where, } j = 1 \div M \text{-----(22)}$$

Layer 4: The output layer is the fourth layer which contains P nodes. Initially, the P is set as 1, which is considered the river run-off value. This layer also liable to convert the fuzzy data into a crisp form.

$$y_k = O_k^{(4)} = \sum_{j=1}^M u_{jk}^{(4)} w_{jk} = \sum_{j=1}^M u_{jk}^{(3)} w_{jk} \text{-----(23)}$$

$$y_k = \sum_{j=1}^M w_{jk} \prod_{i=1}^N \exp \left[- \frac{[x_i(t) + \theta_{ij} O_{ij}^2(t-1) - m_{ij}]^2}{(\sigma_{ij})^2} \right] \text{ where, } K = 1 \div P \text{-----(24)}$$

After initializing the FRNN architecture, the HHO algorithm is integrated with FRNN to accomplish the training process. While compared with other algorithms, the behaviour shown by HHO is found inspiring in this classification process.

Harris Hawk Optimization (HHO)

In this approach, a hybrid the HHO algorithm with FRNN for identifying the optimal weight parameter. The hunting process of HHO comprises four different activities: tracking, encircling, approaching, and attacking. Normally, the entire hunting process is bagged up into three main phases: exploration, conversion from exploration to exploitation, and exploitation. The flowchart for the HHO algorithm is shown in Figure 3. After completing the searching phase, initiate the first stage by discovering the prey (rabbit) position. Then, the hawks define their position based on the randomly generated solution, X_{rand} .

$$X(t-1) = \{(X_{rabbit}(t) - X_m(t)) - r3(L_b + r_4(U_b - L_b))\}, q < 0.5 \text{-----(25)}$$

Where, X_m indicates the average position and the random number that varies from 0 to 1 is represented as, r. The formula used to estimate X_m , is shown in Equation (25).

$$X_m(t) = \frac{1}{n} \sum_{i=1}^N X_i(t) \text{-----(26)}$$

Where, the position and size of hawk is represented as X_i , and N respectively. Next, the escaping energy (E) during hunting is evaluated using the Equation (26).

$$E = 2E_o \left(1 - \frac{t}{T}\right) \text{-----}(27)$$

The flowchart for HHO is depicted in Figure 3. Where, the maximum size in terms of repetition is indicated as, T, and o represents the initial energy which ranges from 1 to 1. The decision that is taken to determine whether to initiate exploration or exploitation phase is identified by evaluating the parameter, $|E|$. During the exploitation phase, the selection of besiege using $|E|$ is also essential, i.e. if $|E|$ is < 0.5 then take the hard besiege, and if $|E|$ is greater than or equal to 0.5 then it can go for soft besiege. The weight parameter that provides less error during classification is considered as the best solution which is optimally deter-

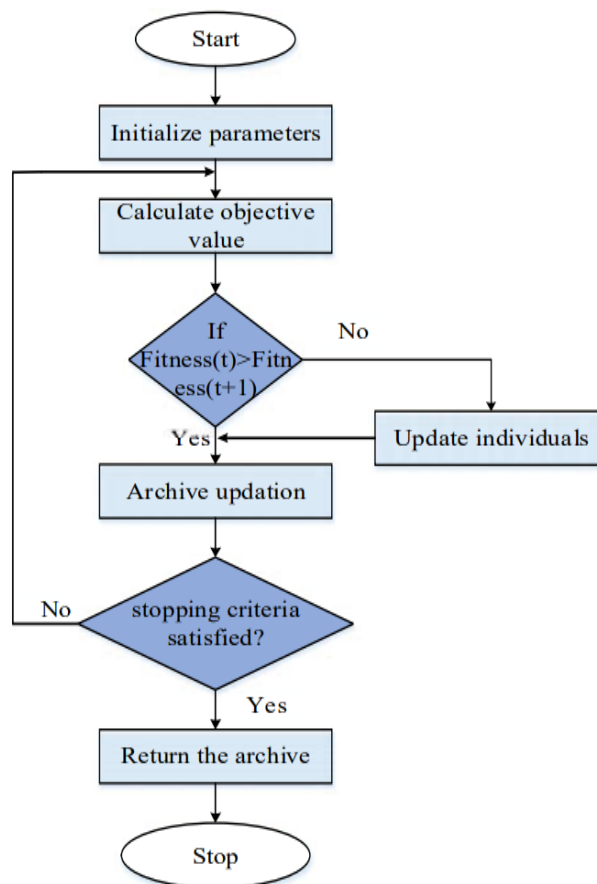


Figure 3: Flowchart for HHO Algorithm

mined by the HHO algorithm. Further, it reduces the computational complexity of FRNN during sentiment classification.

Algorithm 1 TWEET Sentiment Classification using HHO Algorithm
Input: TWEET dataset
Output: Sentiment from classified TWEET
1: Initialize input TWEET data
2: Read Document 'E'

```

3: for Each word 'w' of document E do
4: Pre-processing steps
5: Tokenization
6: Stop word removal
7: Stemming
8: POS tagging
9: end for
10: for Feature Extraction do
11: LSA
12: ICA
13: if Sentiment Lexicon then
14: Training the KELM classifier
15: The weight parameter selection of FRNN classifier using HHO
16:  $X_{Rabbit} = Fitness Value$ 
17: T=1
18: While (T≤ Max Iteration)
19: Next
20: Update  $X_{Rabbit} = X (t+1)$ 
21: end if
22: Performance evaluation to test the effectiveness of the proposed work
23: end for
    
```

4. Result and Discussion

The dataset utilized in this study is the sentiment140 dataset, comprising a vast collection of 1,600,000 tweets sourced through the Twitter API. These tweets have undergone manual annotation to assign sentiment polarity labels, where a label of 0 signifies negative sentiment and 4 represents positive sentiment, rendering it conducive for sentiment analysis tasks. Within the dataset, six key fields are present for each tweet entry. Firstly, the "target" field indicates the polarity of the tweet, with values ranging from 0 to 4, encompassing negative, neutral, and positive sentiments. Secondly, the "ids" field assigns a unique identifier to each tweet for tracking and organizational purposes. The "date" field captures the timestamp denoting the date and time of tweet creation, adhering to the UTC (Coordinated Universal Time) standard. Furthermore, the "flag" field records any associated query term if the tweet originates from a specific query; otherwise, it is marked as NO_QUERY. The "user" field contains the username of the Twitter account responsible for the tweet, while the "text" field encapsulates the actual

textual content of the tweet, encompassing hashtags, mentions, emoticons, and other textual features. This dataset serves as a valuable resource for developing and evaluating sentiment analysis algorithms and models, offering a diverse range of annotated tweets spanning various sentiments.

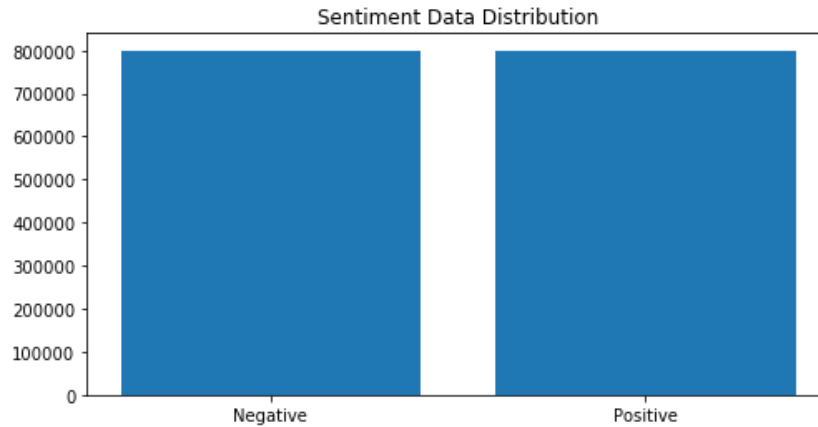


Figure 4. Data Distribution (Positive and Negative Tweets)

In Figure 4, the data distribution of positive and negative tweets is depicted, illustrating the relative proportions of each sentiment class within the dataset. This visualization provides insights into the balance or imbalance of sentiment labels, which is crucial for understanding the composition of the dataset and its potential impact on model performance. Figure 5 presents the correlation matrix, a graphical representation of the correlations between different features or variables in the dataset. In the context of sentiment analysis, this matrix could reveal correlations between specific words or features and sentiment labels, aiding in feature selection and interpretation of the model's decision-making process.

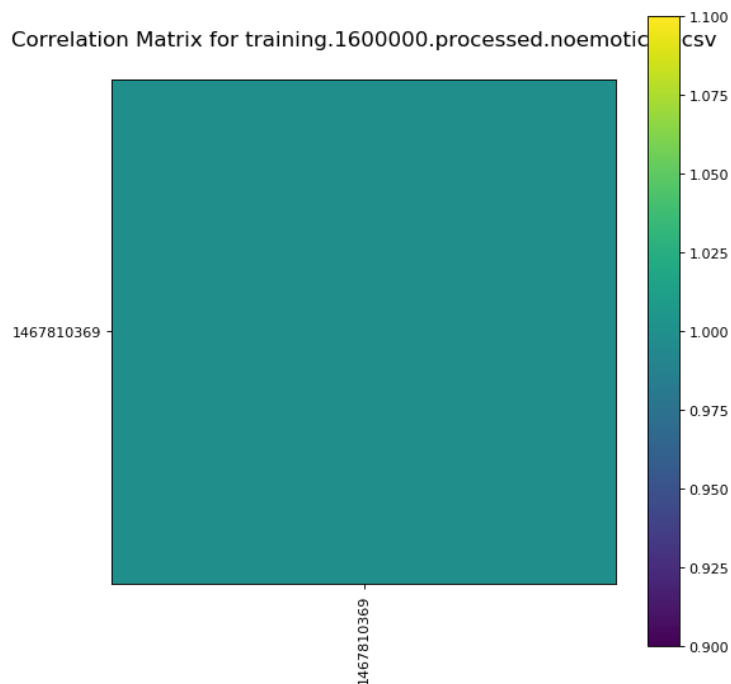


Figure 5. Correlation Matrix

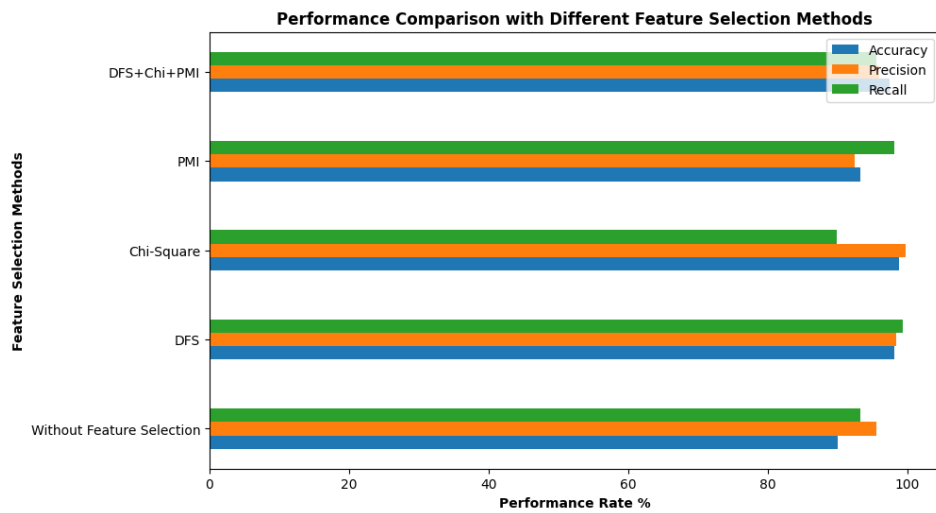


Figure 7. Performance Comparison of With and Without Feature Selection

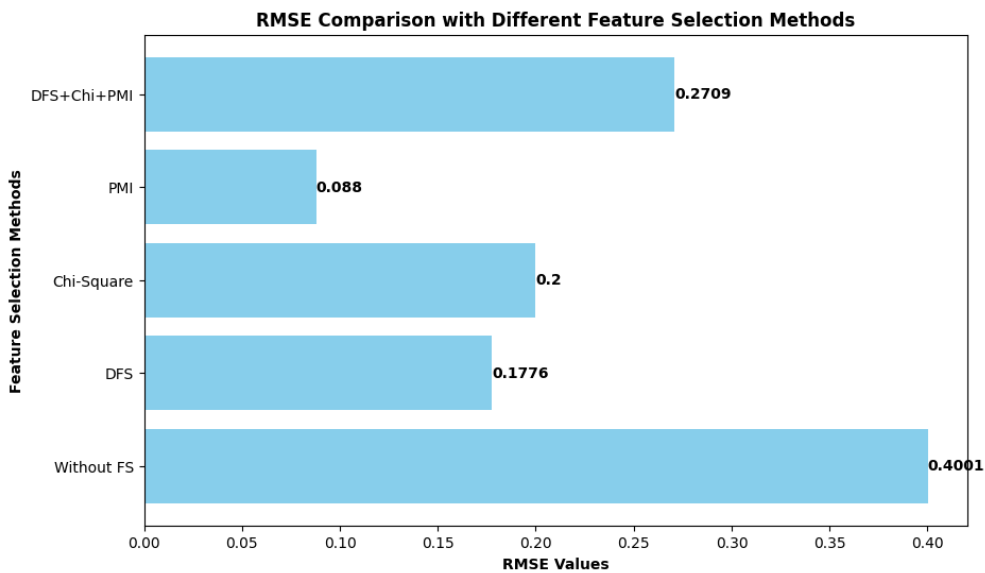


Figure 8. Performance Comparison of RMSE

Table 2. Comparison of Classification

Methods	Accuracy	Precision	Recall	RMSE
LDA and NMF	93.2	92.45	93.56	0.967
SVM	97.832	94.34	86	0.8967
CNN-LSTM	98.372	95.34	89.56	0.782
KELM	98.61	98.56	98.14	0.187

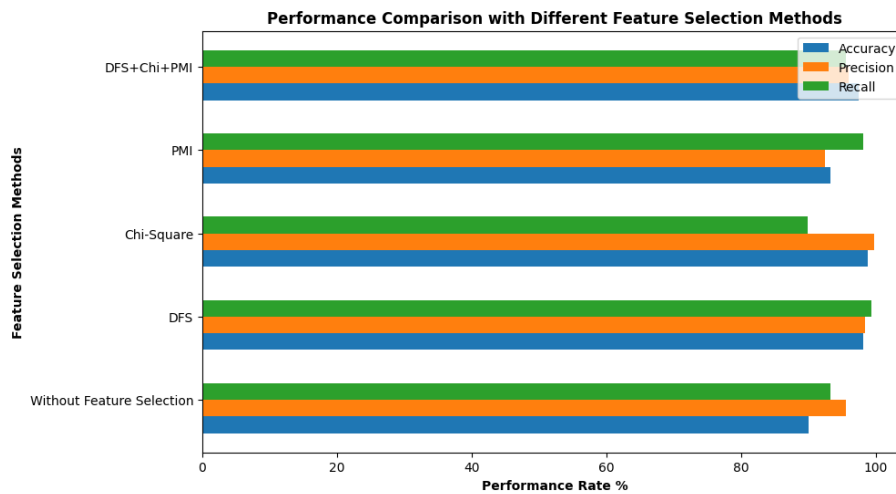


Figure 9. Comparison of Classification

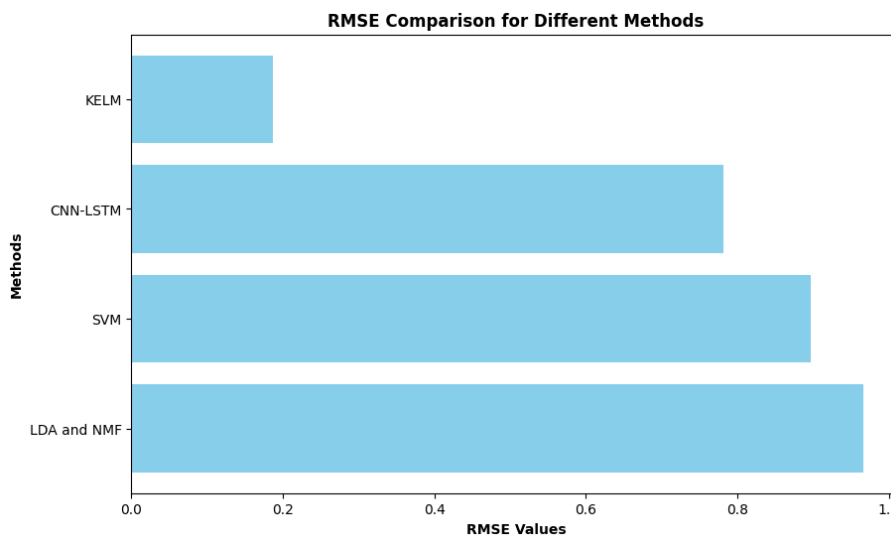


Figure 10. Comparison of RMSE

Table 1 presents a comparative analysis of sentiment analysis performance with and without feature selection using various metrics. The results highlight the impact of different feature selection techniques, including Document Frequency Subset (DFS), Chi-Square, Pointwise Mutual Information (PMI), and their combination (DFS+Chi+PMI), on sentiment analysis accuracy, precision, recall, and Root Mean Squared Error (RMSE). Notably, incorporating feature selection techniques significantly improves performance across all metrics compared to the baseline without feature selection. Specifically, DFS, Chi-Square, and PMI individually enhance accuracy, precision, and recall, with notable improvements observed in precision and recall metrics. However, the combined approach (DFS+Chi+PMI) achieves the highest accuracy, precision, and recall, demonstrating the complementary benefits of integrating multiple feature selection methods. Nevertheless, RMSE values indicate a slight increase compared to the baseline, suggesting a trade-off between classification accuracy and regression error.

Figure 7 provides a visual representation of the performance comparison with and without feature selection, reinforcing the trends observed in Table 1 and highlighting the substantial performance gains

achieved through feature selection techniques. Table 2 extends the comparative analysis by evaluating the performance of different classification methods, including Latent Dirichlet Allocation (LDA) and Non-Negative Matrix Factorization (NMF), Support Vector Machine (SVM), Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM), and Kernel Extreme Learning Machine (KELM). The results demonstrate varying degrees of performance across classification methods, with CNN-LSTM and KELM outperforming LDA and NMF as well as SVM in terms of accuracy, precision, recall, and RMSE. Particularly, CNN-LSTM and KELM exhibit superior performance across all metrics, indicating their efficacy in sentiment analysis tasks.

Figure 8 visually compares the classification methods, corroborating the findings presented in Table 2 and emphasizing the relative performance of each method in terms of accuracy, precision, recall, and RMSE. Overall, the comparative analysis underscores the importance of feature selection techniques and the effectiveness of advanced classification methods in improving sentiment analysis performance.

5. Conclusion

The sentiments from the classified messages are determined using HHO based FRNN classifier. Before that, the messages need to be preprocessed because pre-processing may reduce the error rate of the classification process. The features that are presented in preprocessed data are extracted using sentiment-based and classification-based feature extraction techniques. Finally, the dimension of extracted features is reduced using the three most efficient feature selection techniques. Then, the selected features are provided to the classification algorithm.

Reference

- [1] Alameri, I., Komarkova, J., & Ramadhan, M. K. (2021, June). Conceptual analysis of single and multiple path routing in MANET network. In 2021 International Conference on Information and Digital Technologies (IDT) (pp. 235-244). IEEE.
- [2] Mankotia, V., Sunkaria, R. K., & Gurung, S. (2023). DT-AODV: A dynamic threshold protocol against black-hole attack in MANET. *Sādhanā*, 48(4), 190.
- [3] Ramalingam, R., Muniyan, R., Dumka, A., Singh, D. P., Mohamed, H. G., Singh, R., ... & Noya, I. D. (2022). Routing protocol for MANET based on QoS-aware service composition with dynamic secured broker selection. *Electronics*, 11(17), 2637.
- [4] Al Ogaili, R., Alsaedi, A. H., Alkafagi, S. S., & Alfoudi, A. S. D. (2022). A critical review of Optimization MANET routing protocols. *Wasit Journal of Computer and Mathematics Sciences*, 1(4), 70-86.
- [5] Ramphull, D., Mungur, A., Armoogum, S., & Pudaruth, S. (2021, May). A review of mobile ad hoc NETWORK (MANET) Protocols and their Applications. In 2021 5th international conference on intelligent computing and control systems (ICICCS) (pp. 204-211). IEEE.
- [6] Kaur, G. and Thakur, P., 2019, July. Routing protocols in manet: An overview. In 2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT) (Vol. 1, pp. 935-941). IEEE.
- [7] Chitkara, M., & Ahmad, M. W. (2014). Review on manet: characteristics, challenges, imperatives and routing protocols. *International journal of computer science and mobile computing*, 3(2), 432-437.
- [8] Abbas, T., Qamar, F., Hindia, M. N., Hassan, R., Ahmed, I., & Aslam, M. I. (2020, September). Performance analysis of ad hoc on-demand distance vector routing protocol for MANET. In 2020 IEEE student conference on research and development (SCOREd) (pp. 194-199). IEEE.
- [9] Saudi, N. A. M., Arshad, M. A., Buja, A. G., Fadzil, A. F. A., & Saidi, R. M. (2019). Mobile ad-hoc network (MANET) routing protocols: A performance assessment. In *Proceedings of the Third International Conference on Computing*,

Mathematics and Statistics (iCMS2017) Transcending Boundaries, Embracing Multidisciplinary Diversities (pp. 53-59). Springer Singapore.

- [10] Alameri, I., Komarkova, J., & Ramadhan, M. K. (2021, June). Conceptual analysis of single and multiple path routing in MANET network. In 2021 International Conference on Information and Digital Technologies (IDT) (pp. 235-244). IEEE.
- [11] Long, W., Jiao, J., Liang, X., & Tang, M. (2018). An exploration-enhanced grey wolf optimizer to solve high-dimensional numerical optimization. *Engineering Applications of Artificial Intelligence*, 68, 63-80.
- [12] Pu, X., Wu, G., & Yuan, C. (2019). Exploring overall opinions for document level sentiment classification with structural SVM. *Multimedia Systems*, 25, 21-33.
- [13] Al-Smadi, M., Qawasmeh, O., Al-Ayyoub, M., Jararweh, Y., & Gupta, B. (2018). Deep Recurrent neural network vs. support vector machine for aspect-based sentiment analysis of Arabic hotels' reviews. *Journal of computational science*, 27, 386-393.
- [14] Moraes, R., Valiati, J. F., & Neto, W. P. G. (2013). Document-level sentiment classification: An empirical comparison between SVM and ANN. *Expert Systems with Applications*, 40(2), 621-633.
- [15] Ali, S., Shaukat, Z., Azeem, M., Sakhawat, Z., Mahmood, T., & ur Rehman, K. (2019). An efficient and improved scheme for handwritten digit recognition based on convolutional neural network. *SN Applied Sciences*, 1, 1-9.
- [16] Kalarani, P., & Selva Brunda, S. (2019). Sentiment analysis by POS and joint sentiment topic features using SVM and ANN. *Soft computing*, 23, 7067-7079.
- [17] Kalaivani, P., & Shunmuganathan, K. L. (2013). Sentiment classification of movie reviews by supervised machine learning approaches. *Indian Journal of Computer Science and Engineering*, 4(4), 285-292.
- [18] <https://www.kaggle.com/datasets/kazanova/sentiment140>