International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Sentiment Analysis on Twitter Data using Modified Neuro Fuzzy Algorithm

[1]Prabakaran N. S., [2]Dr. S. Karthik

Submitted: 27/05/2023 **Revised**: 06/07/2023 **Accepted**: 24/07/2023

Abstract: The utilization of Social Media (SM) has been widely adopted by individuals as a convenient and formal means of communication. Individuals engage in the act of composing or disseminating textual content, as well as appending visual media such as images and videos, on various social networking platforms such as Twitter, Facebook, and other analogous digital platforms. Sentiment analysis (SA), also known as Opinion Mining (OM), is a prevalent task in dialogue preparation that seeks to uncover the underlying sentiments expressed in texts pertaining to diverse topics. The collection of data from social media platforms can be effectively leveraged to address various objectives, including the marketplace prediction, product suggestion, and analysis of reviewer sentiment. Managing unstructured data found on social media poses a significant challenge. Deep learning algorithms are a suitable solution for addressing the challenges associated with handling this type of data. Therefore, this study aims to conduct SA on a dataset collected from Twitter. In the proposed study, the initial step involves preprocessing the input Twitter data. Following the preprocessing stage, the words are organized in a structured format utilizing HDFS (Hadoop Distributed File System). The process of Map Reducing is then applied to eliminate duplicate words and establish the structured format. The features are derived from the Proposed Modified Principal Component Analysis (MPCA) method. In the final stage, the features are classified utilizing the Modified Fuzzy Neural Algorithm (MFNA) that has been proposed. The simulation results of the proposed method demonstrate superior performance in comparison to existing methods. Ultimately, the outcome was evaluated through the utilization of the K Fold Cross Validation technique. The aforementioned procedures are implemented in twitter data set publicly available in the Kaggle. The highest level of accuracy achieved is 96.984%, which is correlated with the sentiments of three classes positive, negative and neutral.

Keywords*: Twitter data, MPCA, K-fold validation, MFNA, kaggle*

1. Introduction

Individuals utilize SM platforms such as Twitter and Facebook as a means of expressing their viewpoints, presenting arguments, and articulating their emotions pertaining to their everyday experiences. Twitter is a popular micro-blogging site where users may publish and read short updates (or "Tweets") from other users all over the world. One of the sources of noise in Twitter text is the presence of incorrect grammar, freestyle writing, abbreviations, and typographical errors. Sentiment analysis, which involves analyzing the sentiment expressed in a tweet shared by a consumer, and opinion mining, which involves analyzing client testimonials, are both well-known research topics. The texts were collected from individual tweets using Opinion Mining and automatic Sentiment Analysis, which categorizes them into three ternary categories: Positive, Negative, and Neutral. Researchers face significant challenges when attempting to analyze thoughts expressed in Twitter data due to various factors such as limited dimensions, misspellings, abbreviations, and the use of slang. In order to comprehend the advancements in Tree-based sentiment classification, an investigation is conducted on the task of sentiment analysis and its classification in the context of Twitter analysis. They are employed to mitigate substantial and pivotal workloads. Moreover, sentiment analysis encounters significant challenges primarily related to selecting and categorizing features when applied to Twitter data. In order to conclude this study, a classification framework is proposed, which utilizes a Modified Principal Component Analysis (PCA) approach for Twitter data. This framework aims to identify oppressive feature subsets by employing gradient boosted decision normalization techniques.

SA within the field of NLP, pertains to the assessment and interpretation of opinions or attitudes expressed by individuals towards a particular objective. Social analysis (SA) is a systematic procedure that involves the gathering and examination of data derived from the personal opinions, perspectives, and thoughts of individuals. Sentiment analysis (SA) is conducted through the utilization of Natural Language Processing (NLP), Statistical models, and various machine learning (ML) algorithms to extract features from extensive datasets. The pursuit of comprehending a prevalent methodology for making sound decisions can enhance one's knowledge. Initially, the pre-processing of Twitter

[1]*Agile Coach, TATA Consultancy Services, Bangalore-560 066.*
Prabakaran.natrajan@tcs.com
[2]*Prof & Head, SNS College of Technology, Coimbatore-641 035*
profskarthik@gmail.com

data is expanded. Next, the pre-processed data is subjected to processing using the HDFS MapReduce framework. The extraction of features is performed on the processed information, followed by the selection of effective features using the MPCA method. The classification process is ultimately completed through the utilization of the Modified Neuro Fuzzy Algorithm (MNFA), and the resulting outcomes are subsequently verified using the K-fold cross validation technique.

2. Literature Survey

Erick Odhiambo Omuya et al. developed a machine learning model for SA on SM data and other datasets. They used Naive Bayes, SVM, and K-nearest neighbor to evaluate the model's efficiency and compare it to other contemporary models. The study found that utilizing diverse speech parts, training on preprocessed datasets, and reducing dimensions significantly enhances SA models' performance.

The study found that the model demonstrated superior accuracy compared to other models using the same data set. It showed stability and consistency in its performance. Sentiment analysis was enhanced through dimensionality reduction techniques, diverse speech parts, appropriate model training, and noise-free data. The model successfully incorporated these concepts, achieving the study's objective.

The study assessed the model's efficacy and compared it to other approaches. Results showed that using different speech segments, training on preprocessed datasets, and lowering dimensions significantly improved SA models' performance. The data was visualized using LDA and PCA. The method, which employed five classifiers, achieved accuracies of 82.14%, 81.42%, 77.85%, 76.42%, and 74.28%. This method holds potential for use in Pakistan's business community, aiding in investment decisions and forecasting future exchange rates.

Abdullah Alsaeedi and Mohammad Zubair Khan's study explores various SA methodologies, including ML, ensemble approaches, and DBA. They found that ML algorithms, like SVM and MNB, showed the highest precision when multiple features were incorporated. Dictionary-based techniques, such as Naive Bayes, Maximum Entropy, and SVM, were highly effective and require minimal human annotation. Ensemble and hybrid-based algorithms for Twitter SA were superior to supervised machine learning techniques, with classification accuracy of around 85%. Hybrid techniques also demonstrated strong performance and satisfactory classification accuracy scores, leveraging the strengths of both ML classifiers and lexicon-based approaches.

The challenge of identifying the most effective method for sentiment detection in Twitter data is complex due to the absence of established benchmarks. A previous study found that using datasets used in microblogging sentiment competitions could help alleviate this problem.

Abdullah Alsaeedi and Mohammad Zubair Khan's study explores various SA methodologies, including ML, ensemble approaches, and DBA. They found that ML algorithms, such as SVM and MNB, showed the highest precision when multiple features were incorporated. Dictionary-based techniques, such as Naive Bayes, Maximum Entropy, and SVM, were highly effective and require minimal human annotation. Ensemble and hybrid-based algorithms for Twitter SA were superior to supervised machine learning techniques, with classification accuracy of around 85%. Hybrid techniques also demonstrated strong performance and satisfactory classification accuracy scores, leveraging the strengths of both ML classifiers and lexicon-based approaches.

The challenge of identifying the most effective method for sentiment detection in Twitter data is complex due to the absence of established benchmarks. A previous study found that using datasets used in microblogging sentiment competitions could help alleviate this problem.

A comprehensive assessment of machine learning classifiers revealed Naive Bayes (NB) classifiers have the highest accuracy rate of 80% on Twitter and reviews datasets. IB-k classifier outperformed all other classifiers in geopolitical datasets with a 95% accuracy rate. The proposed technique's scalability was assessed through execution times, showing a linear increase in speed as dataset size increased. Genetic algorithm-based feature sets resulted in a significant acceleration in modeling classifiers, with an observed speed increase of up to 55%.

Dangi et al. developed an algorithm that integrates Convolutional Neural Networks (CNNs) with Genetic Algorithms (GA) for effective generalization across different CNN architectures. This comprehensive autonomous training approach identifies optimal hyperparameter setups for facial sentiment analysis. The methodology achieved an accuracy rate of up to 96.984% when compared to leading methodologies. It is automated, making it user-friendly, even for those with limited knowledge of CNNs or GAs.

The assessment of polarity in a document is a crucial element within the field of text mining. The topic of utilizing tree kernels in future engineering has been previously addressed [8]. This particular technique demonstrates superior outcomes compared to alternative methodologies. The technique in question was introduced by the authors, who provided definitions for

two distinct classification models: a two-way classification model and a three-way classification model. In the context of sentiment analysis, sentiments can be categorized into two distinct classes: positive or negative. However, in the case of 3-way classification, sentiments are further classified into three distinct categories: positive, negative, or neutral.

In this study authors have developed a model for movie reviews that is based on domain-specific features [10]. In this context, the utilization of an aspect-based technique is employed to examine textual reviews on movie data and allocate a sentiment label to each review based on the specific aspect being evaluated. The various components are subsequently consolidated from numerous critiques, and the sentiment rating of a particular film is ascertained. The researchers employed a methodology based on sentiment WordNet to extract features and perform analysis on sentiment data at the document level. The obtained results were compared with those acquired through the utilization of Alchemy API. The feature-based model yielded superior outcomes compared to the model based on the Alchemy API.

3. Proposed Method

In the proposed study, the processing of natural language is conducted in a series of stages. Initially, the input Twitter data undergoes preprocessing techniques such as tokenization, removal of URLs and hash tags, handling of abbreviations and acronyms, capitalization of subject words, and removal of usernames. These preprocessing steps aim to transform unstructured data into a structured format. After preprocessing, the words are organized in a structured format using HDFS (Hadoop Distributed File System). Map reducing is then applied to eliminate duplicate words and further refine the structured format. Following this step, the features are extracted. Thirdly, the features are extracted from the resulting word after the HDFS process. The features utilized in this study were derived from two existing papers and were subsequently merged with additional features including Bag-of-Words (BOW), Part-of-Speech (POS), and Internet slang. In the fourth step, the selection of features is carried out utilizing the Modified Principal Component Analysis (PCA) algorithm. Next, the selected featured words will be ranked. Subsequently, the Modified Neuro Fuzzy Algorithm is employed to classify the data by utilizing the ranking value of the extracted features, categorizing them as positive, negative, or neutral for the purpose of SA. The proposed algorithm combines elements of both neural networks and fuzzy systems, resulting in a Modified Neuro Fuzzy Algorithm. In order to optimize the execution time, the features are initially clustered using the K-Medoid Algorithm prior to classification. The simulation results of the proposed method demonstrate superior performance in comparison to existing methods. Ultimately, the outcome was evaluated through the utilization of the K Fold Cross Validation technique. The dataset was obtained through the utilization of the Ruby Twitter API.

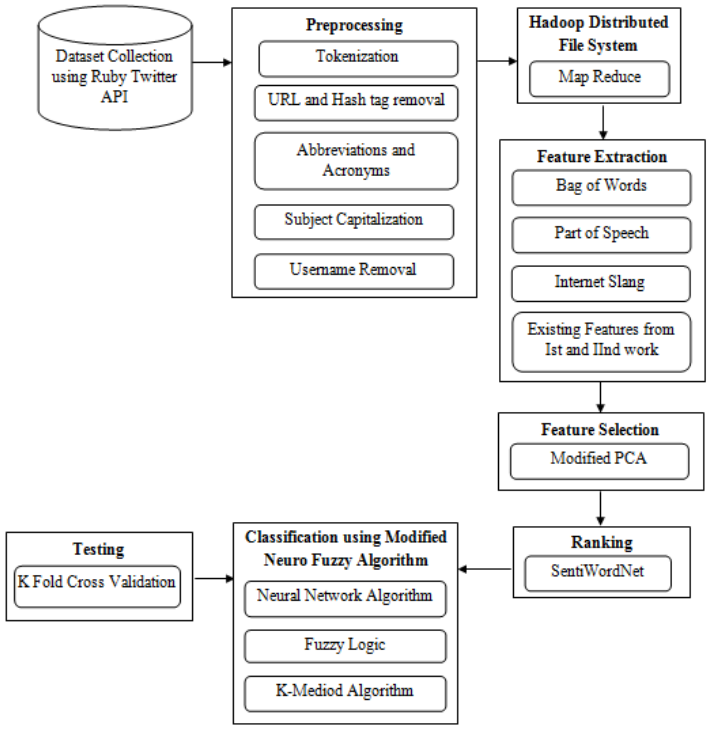

Fig 1: Proposed PCCA-FNN model architecture

3.1 Data Set

The dataset utilized in this study is the Twitter Sentiment Analysis Training Corpus (Dataset), which primarily consists of Twitter data. The dataset consists of a substantial number of tweets that have been previously classified based on their sentiment. The dataset is based on data obtained from two sources. The main source of information for this study is the SA competition hosted by the University of Michigan on the Kaggle platform. Each entry in the data file represents a sentence that has been extracted from SM platforms, specifically blogs. The secondary source utilized in this study is the 'Twitter Sentiment Corpus' authored by Niek Sanders. The dataset comprises a total of 5513 tweets that have been manually classified by human annotators. The tweets were categorized based on one of the four distinct topics. The dataset comprises data instances that have been labeled as positive, negative, or neutral.

The Twitter SA Dataset consists of 896,886 tweets that have been categorized. Each row in the dataset includes the ItemID, Sentiment, SentimentSource, and SentimentText. The Sentiment variable is assigned a value of '1' to indicate positive sentiment and '0' to indicate negative sentiment. In this scenario, a fraction of 1/10 of the corpus is allocated for the purpose of testing, while the remaining majority is allocated for training in order to facilitate sentiment classification.

3.2 Preprocessing

The proposed system preprocesses Twitter datasets to identify emotions and non-emotional symbols. It performs data analysis to avoid misleading outcomes. The dataset includes diverse tweets, including textual words and symbolic characters. Preprocessing procedures include tokenization, hashtag removal, abbreviation, acronym identification, subject capitalization, and user name removal to ensure anonymity and privacy. The system aims to differentiate between emotions-conveying and non-emotional symbols

a. Tokenization

Tokenization is the process of segmenting strings into discrete units such as symbols, words, phrases, key phrases, and other specialized elements, referred to as tokens. Tokens have the ability to serve as terms, individual words, and even entire sentences. Certain characters, like as punctuation marks, are removed from the input before it is tokenized. There are several methods available for dividing an array of tweet tags into individual keywords that correspond to emojis, keys, words, or phrases.

Algorithm for Tokenization

Input: Selected Tweets

Output: Tokenized Tweets

For all words in **Processed Tweets**

Tokenize the word passing to Tweet Tokenizer Method and append Tokenize Sentence

Return Tokenized Sentence

b. URL and Hash tag removal

In the provided text, all hashtags that are denoted by a number sign "#" preceding the words or unspaced phrases are removed.

c. Abbreviation and Acronyms

- In order to facilitate efficient communication and comprehension, it is advisable to construct a comprehensive compendium or reference table encompassing prevalent abbreviations and their respective expanded forms. This can encompass widely recognized initialisms, colloquial expressions, or field-specific abbreviations.
- The Twitter data should be tokenized into discrete words or tokens. The process of tokenization can be achieved by utilizing a tokenizer that is designed for the particular programming language or natural language processing (NLP) library being employed.
- Perform an iterative process on every individual token present in the tweet.
- Verify the presence of the token in the abbreviation lookup table
- If the token is located within the lookup table, it should be substituted with the related expanded form
- The aforementioned procedure should be iteratively applied to each individual token present within the tweet.
- The tokens are reassembled to form a processed tweet

d. Subject Capitalization

- The subject should be tokenized into discrete words. The task can be accomplished by employing a tokenizer that is tailored to the programming language or natural language processing (NLP) library being utilized
- A list is to be created for the purpose of storing words that have been capitalized.
- Perform an iteration process on each individual word within the subject
- When capitalizing the first letter of each word, one can utilize the suitable function or method

offered by the language of programming or library being employed

- The subject should be capitalized if it fulfills the requirements for capitalization. This process generally entails converting the initial character of the text to uppercase
- The subject should be capitalized if it fulfills the requirements for capitalization. This process generally entails converting the initial character of the text to uppercase
- In order to enhance the accuracy and comprehensiveness of the capitalization rules, it is possible to incorporate supplementary examinations or regulations to address specific scenarios such as proper nouns, acronyms, or instances that deviate from the established capitalization guidelines.

e. User name Removal

To eliminate user names or mentions from Twitter data, one can adhere to the following procedures:

- The objective is to determine the structure or pattern of user names or mentions within the Twitter data. In the majority of instances, user names are typically preceded by the "@" symbol.
- The tweet should be tokenized into particular phrases or tokens using a tokenizer that is specific to the programming language or NLP library being utilized.
- Perform an iteration process on every individual token present in the tweet
- To determine whether a token is a user name or mention, it is necessary to verify if the token commences with the "@" symbol.
- If the token corresponds to a user name or mention, there is the option to either eliminate it completely or substitute it with a placeholder like "USERNAME" in order to sustain the original structure of the tweet.
- The aforementioned procedure should be iteratively applied to each individual token present within the tweet.
- The tokens are reassembled to form a processed tweet.

3.3 Hadoop Distributed File System

Apache Hadoop MapReduce and Hadoop Distributed File System (HDFS) are based on Google's MapReduce and Google File System (GFS). Hadoop Distributed File System (HDFS) provides a reliable approach for managing large volumes of Big Data and enhances the efficiency of data transfer between nodes. MapReduce proposes a structured framework for the processing of information. In the present study, the utilization of MapReduce is employed to effectively eliminate redundant sensor information with respect to temporal considerations.

The proposed study aims to minimize the collection of superfluous sensor data that is generated and recorded over a period of time. Figure 2 depicts the HDFS MapReduce Framework. The HDFS diagram is depicted utilizing a generalized model that employs a minimization structure.

a. Map Reduce

The design of Hadoop's map/reduce framework revolves around the organization of large data collections into smaller, manageable units, enabling efficient processing of vast amounts of information. In order to effectively process large-scale datasets, the map or reduce framework utilize broadcast strategies across a cluster of systems within a group. Significant scalability across numerous Hadoop clusters on commodity hardware is made possible by a programming capability in Apache Hadoop. In a Hadoop cluster, the MapReduce framework is employed to process vast unstructured datasets by leveraging a distributed algorithm. Hadoop applications are responsible for the execution of such jobs. The term "MapReduce" refers to a computational framework that encompasses two distinct tasks, known as the Map Job and the Reduce Job. In order to establish associations between keys and values, the project employs a mapping technique to link procedural steps with corresponding datasets. Key-value pairs are entered as input and used to produce the desired results in job reduction. Both the input signal and the output signal are stored in the HDFS.

The Map Reduce framework consists of two fundamental functions.

- Map () Function
- Reduce() Function

i. Mapping Function

In the Map () function, The Master Node (MN) partitions its input data into smaller sub blocks in order to distribute themselves to the worker nodes. The moment processes are carried out by this worker node, which also notifies the master node of its receipts. The sub-operations are executed collectively to accommodate diverse endeavors. The input data is divided in order to distribute the work among all of the mapping nodes. All information is gathered and sent to the first node. As a result, we get new sets of two people, or tuples. All of the nodes continue to use the same mapping function. In addition, the tuples are sent to the reduction nodes.

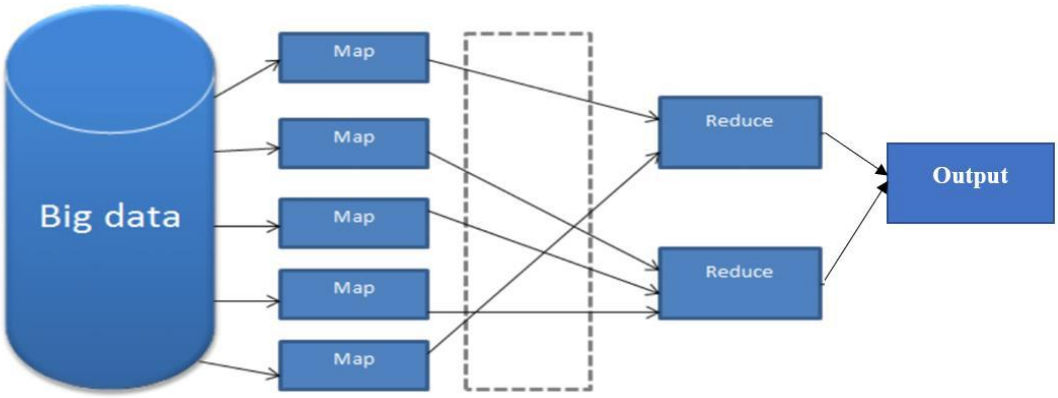

Fig 2: Map Reduce framework of HDFS

ii. Reduce Function

The results of the extensive sub-operations are constructed by the Reduce() function. The findings are integrated to generate consolidated decision-focused outcomes provided as an Acknowledgment of the initial substantial requirements. The reducer nodes are responsible for processing the entirety of the tuples. Therefore, all pairs possessing the identical key are aggregated. Additionally, the value of said key has been enhanced.

3.4 Feature Extraction and Selection

3.4.1 Feature Extraction

The feature part of the Information can serve as a useful tool in addressing forecasting challenges. The significance of both the quality and quantity of features cannot be understated in relation to the outcomes produced by the selected model. This study aims to identify the specific characteristics of the datasets that are particularly advantageous in the detection and analysis of sentiments. The primary objective is to identify distinctive characteristics that can categorize the data into positive, neutral, or negative classes, thereby enhancing the precision of Sentimental Classification. The utilization of diverse models such as bag of words, part of speech, internet slang, and other features from previous works aids in the assessment of individuals' opinions [11][12].

3.4.2 Feature Selection

In previous studies, Principle Component Analysis (PCA) has been employed to derive feature vectors from the dataset obtained from Twitter. One limitation of employing Principal Component Analysis (PCA) is its tendency to disregard features that exhibit minor variations in correlations, potentially overlooking valuable feature information. The present study addresses this concern by implementing a modification to the PCA algorithm through the utilization of the proposed Modified Principle Component Analysis (MPCA) methodology.

The eigenvector values in MPCA are mitigated through the process of vector normalization. Let us consider $A_{x,y}$ as the y^{th} feature vector. In this case, the application of the Standard Deviation $\sqrt{\lambda_y}$ on the feature vector is possible. The feature vector A't can be write in the form of Equation (1).

$$A'_t = \left[\frac{A_{i0}}{\lambda_0}, \frac{A_{i1}}{\lambda_1} \dots\dots\dots\dots\dots \frac{A_{i(n-1)}}{\lambda_{n-1}}\right] \quad (1)$$

The normalization of feature vectors leads to the creation of a novel feature subspace. In this study, the feature vector values are normalized by dividing them by the square root of their corresponding eigen values. Subsequently, the training and testing feature distances are calculated. The linear transformation of PCA is illustrated in Equation (2).

$$A = TI \quad (2)$$

In this context, we consider a T-transform matrix, denoted as T, which operates on a set of feature vectors, represented by I, resulting in a set of transformed feature vectors, denoted as A. The matrix T is subjected to transformation using Equation (3).

$$(\lambda I - M)S = 0 \quad (3)$$

The square matrices can be defined as follows: Matrix I is characterized by having identity values along its diagonal. Matrix M represents the covariance matrix of the original data. Matrices U and λ denote the eigen vectors. The eigen vectors Si and λ_i, where i ranges from 1 to n, are computed using equation (2) in a descending order, such that $\lambda_1 \geq \lambda_2 \dots\dots \geq \lambda_n$. The eigenvectors S can be represented as a column vector S $S = [S_1, S_2 \dots\dots S_n]$.

The Twitter dataset was transformed using the proposed MPCA method. The resulting transformed matrix was applied to the training instances and represented by Equations (4), (5), and (6)

$$A = T'I \quad (4)$$

$$B_N = v_1u_1 + v_2u_2 + \cdots . + v_Nu_N \quad (5)$$

$$M = \sum_{i=0}^{1} b_i u_i \ \ where \ \ 1 < N \quad (6)$$

When examining (3) and (5), it is observed that the transformed matrices are derived from the matrix of covariance and a comprehensive dataset of hate speech.

The primary benefit of MPCA lies in its ability to reduce dimensionality and minimize information loss. The procedure under consideration is rooted in PCA, which is a mathematical practice employed to transform high-dimensional data into lower-dimensional representations. This transformation is achieved through a linear mapping, wherein the lower dimensions are determined by the eigenvectors of the variance matrix. The proposed method, known as MPCA, effectively extracts sentiment features while minimizing errors.

Algorithm 1: MPCA

1. Start
2. Find the Mean value X' for the input data X
3. Subtract the Mean value X' from eqn (6)
4. Obtain the new Matrix N
5. Obtain the new covariance matrix $C = NN^T$
6. Obtain the eigen values for new covariance matrix $E_1, E_2, E_3 \ldots \ldots E_N$
7. Calculate eigen vectors from eigen values
8. It is possible to express any vector S as a linear combination of Eigen vectors by employing equation (5)
9. The lower-dimensional dataset is constructed by retaining only the largest eigenvalues
10. Match the combination of words in the given tweets
11. Extract the more informative features
12. Stop

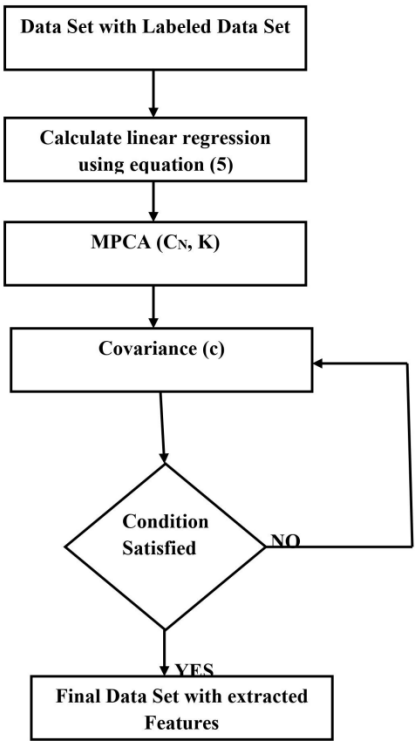

Fig 3: Flow chart of MPCA

3.5 Ranking

SentiWordNet is a lexical resource derived from WordNet, designed to encompass subjective information pertaining to individual words. SentiWordNet gives each phrase numerical values that represent its emotional significance and gives a meaning for every term, which gives further details about the word. SentiWordNet is a highly recommended choice for English speakers seeking a specialized lexical resource. Every definition contained within this dictionary comprises a collection of lexemes that share both the identical grammatical category and meaning, irrespective of their assigned part-of-speech label. Every group is associated with three opinion number ratings that indicate the extent to which the words it comprises are positive, negative, or neutral. The values in question span a range from 0 to 100. The scores under consideration demonstrate a spectrum ranging from 0.0 to 1.0, and the combined sum of these scores within each group equals 1.0. Based on the data presented in Table 1, the term "excellent" is categorically identified as an adjective, with a positive score of 1.0 and a negative score of 0.0. When the term "cold" is used as an adjective to describe a low or insufficient temperature, it exhibits a negative sentiment score of 0.75. Furthermore, when utilized as a noun to denote a

mild viral infection, it exhibits a sentiment score of 0.125 that conveys a negative connotation.

3.6 Classification

The proposed architecture of a Fully Convolutional Neural Network (FCNN) for text sentiment analysis is depicted in Figure 4. The initial step involves passing the input sentence through an embedding layer, resulting in a matrix of real-valued representations. The fuzzification layer is responsible for converting the input matrix into the fuzzy domain.

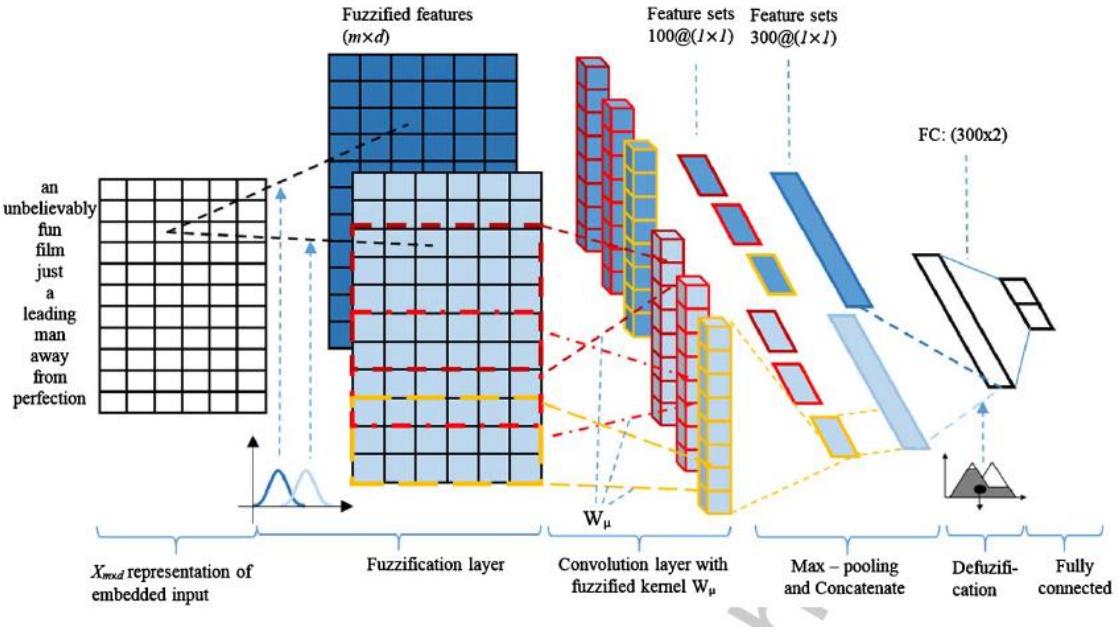

Fig 4: Architecture of FNN

3.6.1 Embedding Level

The sentiment classification (SC) of a given word is conducted by the fully connected neural network (FCNN), which calculates a numerical score corresponding to each emotional label. The model first accept the input as a list of words in a particular order in order to do this task, and then it moves through the model's many levels. At each subsequent layer, there is a progression in the extraction of higher-level features, which are then transmitted to the subsequent layer. The model then proceeds to extract features that span from the vector representation of individual words to the sentiment analysis of entire sentences.

In order to perform accurate computations using a Fully Connected Neural Network (FCNN), it is necessary to represent the words within a sentence as numerical values. The initial stage in the process involves the creation of word-level embeddings, which entails assigning a d-dimensional vector to each word in a given sentence. As a result, the sentence is converted into a matrix with dimensions i×j, where i denotes the length of the sentence (i.e., the number of words) and j denotes the number of dimensions of the embedded vector. In general, there is a variation in sentence length within the dataset. To ensure convenience, a special word is appended at the conclusion of sentences in order to maintain uniform length across all sentences.

For every N-word sentence, we have $W_1, W_2 \dots \dots W_M$. A vector $V_m = [u_1, u_2 \dots \dots u_D]$ will be generated from the W_M words in the phrase. Since our dictionary size is always going to be S, our embedding matrix will always be D ∈ R(i×|S|). By solving for V in terms of WM, we obtain from (7)

$$V_m = Dv^w \tag{7}$$

The symbol v^w represents the vector dimension |V|, which is defined as a vector where the value is 1 at the index w and 0 everywhere else. In the dictionary S, w stands for the index of the word w^m. The values of matrix D are initially set at random and then changed as part of the model's training phase.

3.6.2 FNN Classifier level

After the inputs have been embedded in matrix A, numerous linguistic labels are assigned to each input element based on their membership functions. The membership function of fuzzy determines the grade for the input node's membership in a specific fuzzy collection.

The fuzzy sets A' in Equation (8) are derived using Equation (9), which employs the max product operation

for computation. In the realm of discourse, the input and output data may fall inside the range indicated by the fuzzy reference number $FM_{i,j}$.

$$A` = fuzzification(a_{i,j} | ca_{i,j}) \quad (8)$$

where i and j are indexes of a cell in the input matrix A, and ca is the null set of the fuzzy membership function.

$$a_{i,j} = possibility(a_{i,j} | FM_{i,j}) = \max\ a_{\in} A(FM_{i,j}\, \delta(a - a_{i_j})) \quad (9)$$

Where $\delta(a - a_{ij})$ is the Kronecker delta function.

There are three main processing steps that make up the fuzzy convolution layer: the fuzzy convolution stage, the nonlinearity stage, and the pooling stage. The fuzzy convolutional neural networks (FNN) stage involves the application of fuzzy convolutional algorithms to two-dimensional data, as described in Equation (10). The fuzzy convolutional filters, denoted as Fμ, are computed according to Equation (12), where W represents the original convolution filter.

$$a_{i,j} = \sum_{m=0}^{a-1} \sum_{n=0}^{b-1} W_{\mu} a(i+m)(j+n) \quad (10)$$

$$W_{\mu} = fuzzification(W) \quad (11)$$

The fully connected layer of the fully connected neural network (FCNN) functions as a classifier. The input features of this layer are the crisp values Xi, which are derived through the deffuzzification process using the center of gravity method described in Equation (12). In this equation, C_X represents the center of the membership function used in the defuzzification process. The output of the algorithm is denoted as $Y_i^{`}$, while the weight matrix of the fully connected layer is represented as W_{fc}.

$$X_i = defuzz(a_i) = \frac{\sum C_y a_i}{\sum a_i} \quad (12)$$

$$Y_i' = W_{fc} X_i \quad (13)$$

3.6.3 FNN Training

In Equation (14), where y is the target, y' is the classifier's output, and I is the number of samples, cross entropy is the loss function used to evaluate the output error.

$$E = -\frac{1}{I} \sum_{n=1}^{I} [y_n \log(y_n') + (1 - y_n) \log(1 - y_n')] \quad (14)$$

Using the standard back-propagation learning process and the cross-entropy loss function, the model's parameters are trained. Equation (15) represents the revised weighting system.

$$W_{fc}(p+1) = W_{fc}(p) - \sigma_{fc} \frac{\partial E}{\partial W_{fc}} \quad (15)$$

The equation (15) is used to update the centers $C_y(p)$ of the defuzzification membership functions. The learning rate for updating the center is denoted by lCy in this equation, while y(p+1) and y'(p+1) denote the desired and actual outputs, respectively.

$$C_y(p+1) = C_y(p) + l_{cy} \nabla_{cy} \quad (16)$$

The concept of "center value" refers to a statistical measure that represents the central tendency of a dataset. With Equations (17) and the learning rate α_{yw}, you can figure out the values of Cy and for the fuzzification membership function of the convolution layer's weight.

$$C_y(p+1) = C_y(p) + \alpha_{yw} \nabla W_{\mu} \quad (17)$$

Equations (18) are employed to update the variance and mean of the membership function in the fuzzification layer, with αca representing the learning rate.

$$C_a(p+1) = C_a(p) + \alpha_{ca} \nabla C_a \quad (18)$$

$$\alpha_{ca}(p+1) = \sigma_{ca}(p) + \alpha_{ca} \nabla C_a \quad (19)$$

Algorithm 2: FNN

Step 1training Samples A and the names that go with them B, hyper parameter

Step 2: Trained Parameter of FNN

Step 3: Parameter initialization: Set the weight W and the middle of the membership function to a random value. Cx, Ca, Cy.

Step 4: for e=1 to I do

Step 5: for j=1 to J do

Step 6: A'←fuzzification (A)

$$W_{\mu} = fuzzification(W)$$

Step 7: for n=1 to N do

$A'^{n+1} \leftarrow conv(W^{n}_{\mu}, A'^{n})$

End

$P \leftarrow defuzz(A'^{n})$

$Q \leftarrow fulconnected(P)$

$R \leftarrow crossentrophy(Q, Q')$

$(W_{fc}, C_y, C_{w,} C_x) \leftarrow update\ (W_{fc}, C_y, C_{w,} C_x)$;

$(\sigma_x, \sigma_w) \leftarrow update\ (\sigma_x, \sigma_w)$

End

End

3.7 k-Fold Cross validation

Cross-validation is a method that is often used in the field of predictive modeling. This methodology involves partitioning the original dataset into two distinct subsets, namely a training and a test set, in order to enable their

independent utilization. In the k-fold cross-validation process, the main sample is randomly split into k groups of the same sizeOne subset has been chosen as the validation data to test how well the model works, and the other k-1 groups have been used to train the model. Subsequently, the cross-validation technique is iterated k times, where k represents the overall amount of folds. In each iteration, one of the subsets with k members is exclusively allocated for confirmation, while the remaining subsets are utilized for training. The final accuracy is obtained by calculating the average accuracy across all k-folds. The utilization of tenfold replication in conjunction with cross-validation is a commonly employed methodology in numerous research endeavors. All instances within the dataset are utilized for the purpose of conducting a 10-fold cross validation. Subsequently, these instances are partitioned into 10 distinct groups. There exists a collection of ten distinct groups, wherein nine of these groups serve the purpose of training, while the remaining group is exclusively designated for examination purposes. The method is iterated a total of 10 times, and the average accuracy achieved across all iterations is subsequently computed. The 10-fold cross-validation method yields a greater level of accuracy compared to the alternative method outlined in Table 1.

Table 1: Accuracy using K-fold validation

K-fold cross validation	FNN
1	75.2
2	74.3
3	75.8
4	76.7
5	79.5
6	77.5
7	75.6
8	73.4
AVG	79.3
STD	78.7

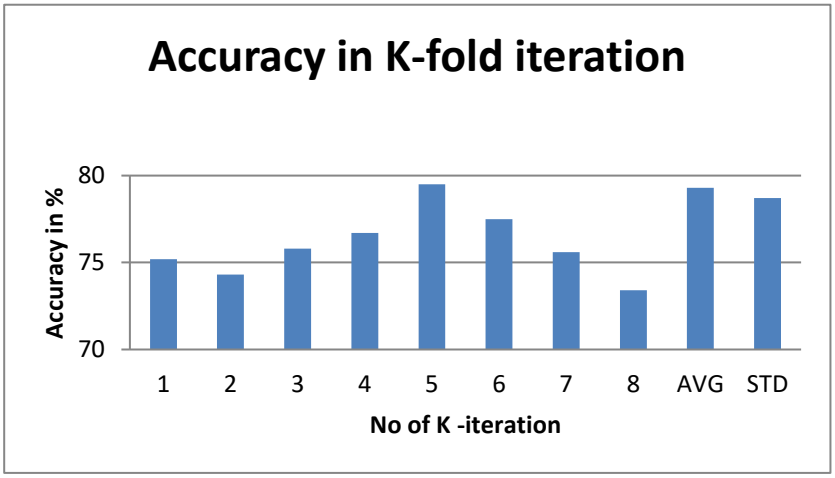

Fig 4: K-fold accuracy validation

4. Result Analysis

In this phase, all of the collected tweets undergo text preprocessing. The aforementioned procedure involves the utilization of the tokenization technique to separate each word into individual tokens. Subsequently, the extraction of salient features from the tweets will be conducted, which will be followed by the generation of scores and an assessment of the tweets' orientation. The

evaluation of the accuracy, precision, and recall metrics is used to assess the efficiency and effectiveness of the proposed classifier.

Table 2.Over all performance of proposed work

	Accuracy	Specificity	Sensitivity	F-Score
Positive	91.3	90.5	89.7	93.2
Negative	94.4	96.2	94.3	94.6
Neutral	95.2	93.6	93.9	92.4
Overall	97.2	91.4	93.9	95.1

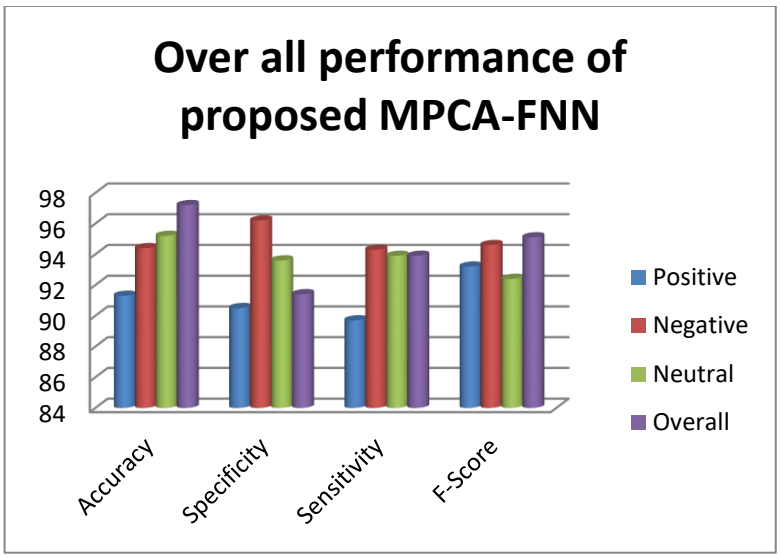

Fig 5: Overall performance comparison

Accuracy

Equation (20) calculates the accuracy rate of any algorithm by determining the percentage of test tuples that are correctly classified.

$$Acc = \frac{TP+TN}{TP+TN+FP+FN} \quad (20)$$

The proposed feedforward neural network (FNN) model demonstrates an overall accuracy of 97.2%. Our work, along with other existing works, has demonstrated higher accuracy compared to previous proposals. The comparison is depicted in Figure 5.

Specificity/Precision

The concept of precision involves assessing the accuracy of a model by determining the proportion of predicted positive instances that are truly positive. Precision is a reliable metric for assessing situations where the costs associated with False Positive outcomes are substantial.

$$Spe = \frac{TP}{TP+FP} \quad (21)$$

Table 2 and Figure 5 illustrate the precision values corresponding to varying quantities of data. The performance of the Proposed FNN is evaluated across all three classes: positive, negative, and neutral. Table 3 and Figure 5 present a comparative analysis of the outcomes between the proposed Feedforward Neural Network (FNN) and previous research efforts. Upon analyzing the graph, it is evident that the proposed work demonstrates a higher level of precision in comparison to other approaches.

Table 3: Comparison of result Proposed Vs Existing

Work	Accuracy	Specificity	Sensitivity	F-Score
CS-MANN	84.2	77.6	79.4	82.8
PSOGA-MCNN	90.4	83.7	87.5	86.9
MPCA-FNN	97.2	91.4	93.9	95.1
KNN	67	70.5	69.3	67.9
SVM	68	69	68.1	68.7
KNN+SVM	76	68.5	68.1	77.5
GA	86	87.3	87.9	87.5
PSO	88	88.7	89.1	81.4
GA	90	90.1	90.3	75.3

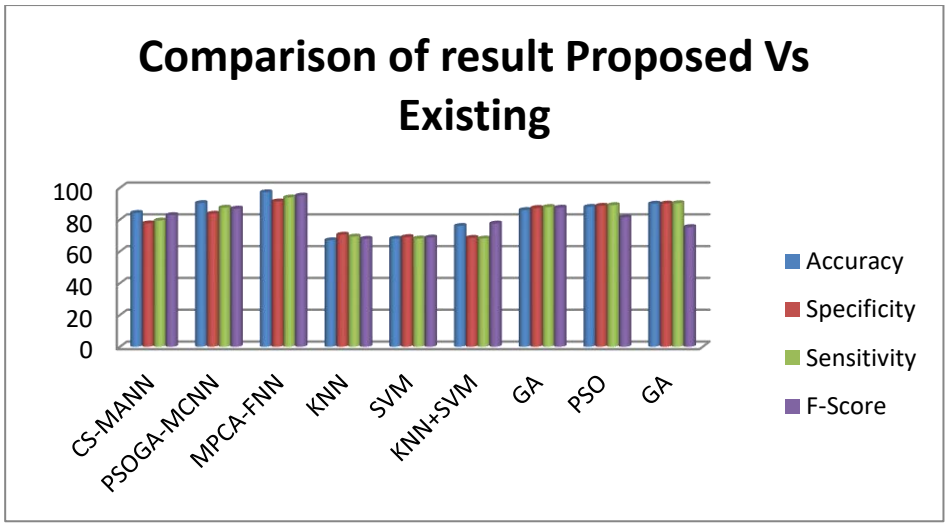

Fig 6: Result comparison Proposed Vs Existing

Sensitivity

Recall is widely recognized as the measure of the successful retrieval of relevant instances, or instead, it can be interpreted as the ratio of properly recognized items that are selected.

$$Sen = \frac{TP}{TP+FN} \quad (22)$$

Table 3, Table 4, and Figure 6 demonstrate that our proposed model accurately predicts a high recall value. This shows that the model effectively captures the majority of positive instances.

F-Score

The F-measure is a metric that quantifies the accuracy of a given test is calculated by using equation().

$$F - Score = 2.\frac{Sen.Spe}{Sen+Spe} \quad (23)$$

Table 5 presents the performance analysis of all three models proposed in this study. The initial work demonstrates an accuracy of 84.2%, while the subsequent contribution, PSOGA-MCNN, achieves a higher accuracy of 90.4%. Finally, the proposed MPCA with FNN achieves an even higher accuracy of 97.2%, as depicted in Figure 7.

Table 4: Proposed Vs Existing

Work	Accuracy	Specificity	Sensitivity	F-Score
CS-MANN	84.2	77.6	79.4	82.8
PSOGA-MCNN	90.4	83.7	87.5	86.9
MPCA-FNN	97.2	91.4	93.9	95.1
KNN	67	70.5	69.3	67.9
SVM	68	69	68.1	68.7
KNN+SVM	76	68.5	68.1	77.5
GA	86	87.3	87.9	87.5
PSO	88	88.7	89.1	81.4
GA	90	90.1	90.3	75.3

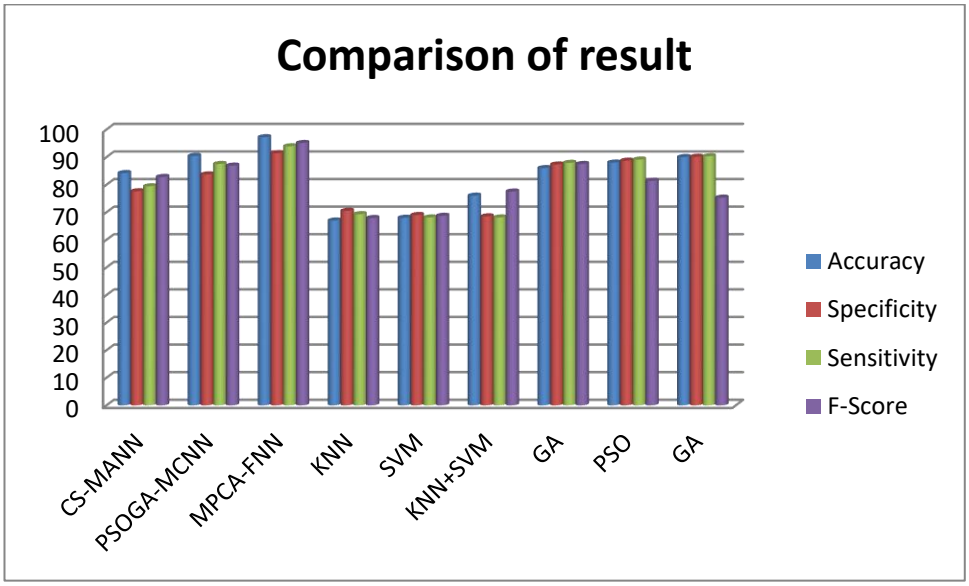

Fig 7: Existing Vs Proposed comparison

Table 5: Performance comparison of proposed contributions

Work	Name of contribution	Accuracy	Specificity	Sensitivity	F-Score
Work 1	CS-MANN	84.2	77.6	79.4	82.8
Work 2	PSOGA-MCNN	90.4	83.7	87.5	86.9
Work 3	**MPCA-FNN**	**97.2**	**91.4**	**93.9**	**95.1**

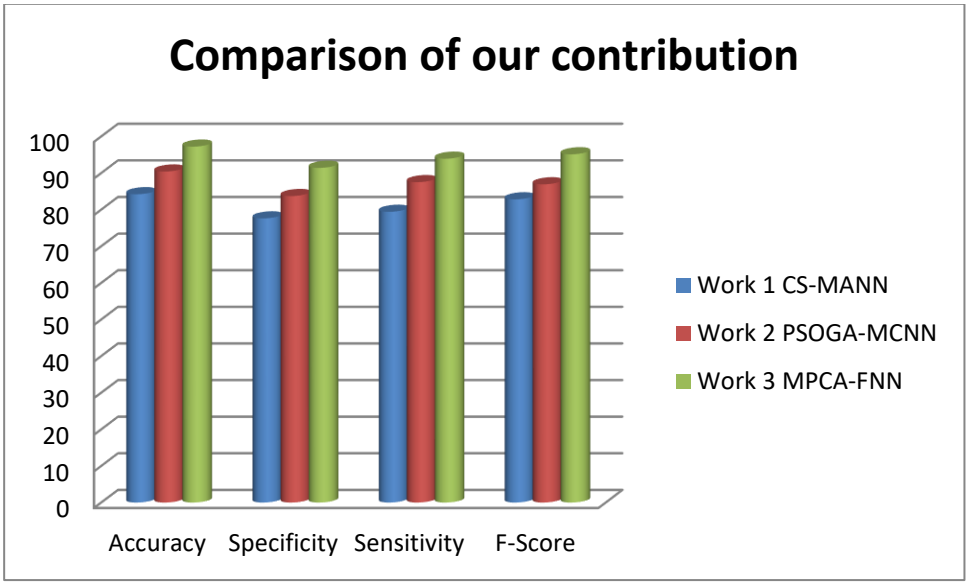

Fig 8: Performance comparison of our contribution

5. Conclusion

This study leverages the benefits of deep learning, fuzzy modeling, and NN to introduce a novel hybrid model that combines deep learning and fuzzy-neural techniques. The proposed approach involves the integration of fuzzy logic and Convolutional Neural Network (CNN) in order to develop a Fuzzy Convolutional Neural Network (FCNN) model for the task of text sentiment classification. The utilization of FCNN demonstrates the capability to produce more rational features, resulting in improved classification accuracies when applied to emotional data, in contrast to conventional methodologies like CNN.

The actual results show that the suggested sentiment analysis (SA) method, which makes use of data from Twitter, is a powerful tool for analyzing huge data. The classification methodology utilized in the study was the Gradient Boosting Decision Tree (GBDT) algorithm. Recall, precision, F-score, accuracy, calculation time, and average sentiment score were some of the performance indicators used in the evaluation and comparison to existing methods.

A proposed methodology for feature selection and classification is anticipated. The utilization of the Hadoop framework in conjunction with the MPCA algorithm is employed to estimate and select optimal features from a high-dimensional dataset obtained from Twitter. The arrangement is conducted using a FNN classifier with an effective feature selection approach classifier. The utilization of MPCA for feature selection showcases the superiority of the proposed grouping system over previous methodologies, as evidenced by its higher accuracy across five benchmark datasets. The lung dataset exhibits a reported accuracy rate of 97.2%. The results indicate that the FNN classifier-based system demonstrates superior accuracy, sensitivity, and specificity compared to existing methods for classifying the tweet dataset.

References

[1] J. Tao and X. Fang, “Toward multi-label sentiment analysis: A transfer learning based approach,” Journal of Big Data, vol. 7, no. 1, pp. 1–26, 2020.

[2] M. Alam, J. F. Wang, C. Guangpei, L. V. Yunrong and Y. Chen, “Convolutional neural network for the semantic segmentation of remote sensing images,” Mobile Networks and Applications, vol. 26, no. 1, pp. 200–215, 2021.

[3] Erick Odhiambo Omuya, George Okeyo and Michael Kimwele, “Sentiment analysis on social media tweets using dimensionality reduction and natural language processing”, Engineering Reports. 2023;5:e12579. wileyonlinelibrary.com/journal/eng2 1 of 14 https://doi.org/10.1002/eng2.12579.

[4] Samreen Naeem, Wali KhanMashwani, Aqib Ali, M. Irfan Uddin, MarwanMahmoud, Farrukh Jamal, and Christophe Chesneau, “Machine Learning-based USD/PKR Exchange Rate Forecasting Using Sentiment Analysis of Twitter Data”, Computers,Materials & Communication, DOI:10.32604/cmc.2021.015872, 2021, vol.67, no.3.

[5] Abdullah Alsaeedi and Mohammad Zubair Khan, “A Study on Sentiment Analysis Techniques of Twitter Data”, International Journal of Advanced Computer Science and Applications, Vol. 10, No. 2, 2019.

[6] Farkhund Iqbal, Jahanzeb Maqbool Hashmi, Benjamin C. M. Fung, Rabia Batool, Asad Masood Khattak, Saiqa Aleem and Patrick C. K. Hung, “A Hybrid Framework for Sentiment Analysis Using

Genetic Algorithm Based Feature Reduction", IEEE Access, February 8, 2019, Digital Object Identifier 10.1109/ACCESS.2019.2892852.

[7] Dharmendra Dang, Amit Bhagat and Dheeraj Kumar Dixit, "Sentiment Analysis on Social Media Using Genetic Algorithm with CNN", Computers,Materials & Continua, DOI:10.32604/cmc.2022.02043, vol.70, no.3, 2022.

[8] A. Insaf, A. Ouahabi, A. Benzaoui and A. T. Ahmed, "Past, present, and future of face recognition: A review," Electronics, vol. 9, no. 8, pp. 1188, 2020.

[9] G. Aaryan, V. Dengre, H. A. Kheruwala and M. Shah, "Comprehensive review of text-mining applications in finance," Financial Innovation, vol. 6, no. 1, pp. 1–25, 2020.

[10] Y. X. Yang, C. Wen, K. Xie, F. Q. Wen, G. Q. Sheng et al., "Face recognition using the SR-CNN model," Sensors, vol. 18, no. 12, pp. 4237, 2018.

[11] Prabakaran N S, "Efficient Natural Language Processing used for Twitter data based on Sentiment Analysis", International Journal of Science & Engineering Development Research, ISSn:2455-2631, vol-4, issue 6, pg.no-368-380, june 2019.

[12] Prabakaran N S, Dr. S.Karthick, " A Hybrid Deep Learning Method for Twitter Data based on Sentiment Analysis", ICTACT, 2023. (Paper under Review).

[13] Benjamin Jackson, Mark Johnson, Andrea Ricci, Piotr Wiśniewski, Laura Martínez. Ethical Considerations in Machine Learning Applications for Decision Science. Kuwait Journal of Machine Learning, 2(4). Retrieved from http://kuwaitjournals.com/index.php/kjml/article/view/221

[14] Dr. S. Praveen Chakkravarthy. (2020). Smart Monitoring of the Status of Driver Using the Dashboard Vehicle Camera. International Journal of New Practices in Management and Engineering, 9(01), 01 - 07. https://doi.org/10.17762/ijnpme.v9i01.81

[15] Dhabliya, D. (2021). Delay-tolerant sensor network (DTN) implementation in cloud computing. Paper presented at the Journal of Physics: Conference Series, , 1979(1) doi:10.1088/1742-6596/1979/1/012031 Retrieved from www.scopus.com