Pythagorean Neutrosophic Bonferroni Mean Based Machine Learning Model for Data Analytics and Sentiment Classification of Product Reviews

Donia Badawood[1,*]

[1] Department of Data Science, School of Computers, Umm Al-Qura University, Saudi Arabia

Email: dybadawood@uqu.edu.sa

Abstract

To handle incomplete and indeterminate data, neutrosophic logic/set/probability was recognized. The neutrosophic falsehood, truth, and indeterminacy modules show symmetry as the truth and the falsehood appear the similar and perform in a symmetrical method with esteem to the indeterminacy module which aids as a line of the symmetry. Soft set is a general mathematical device to deal with uncertainty. Sentiment analysis (SA) is the foremost task of natural language processing (NLP), where judgments, opinions, thoughts, or attitudes toward an exact subject are removed. Web is a rich foundation of information and unstructured covering numerous text documents with reviews and opinions. The detection of sentiment will be useful for governments, discrete business groups, and decision-makers. With this motivation, this study develops a Data Analytics Framework for Sentiment Classification Using Pythagorean Neutrosophic Bonferroni Mean (DAFSC-PNBM) technique on Product Reviews. The presented DAFSC-PNBM technique mainly aims to determine the nature of sentiments based on product reviews. Primarily, data preprocessing is performed to increase the product review qualities. For the word embedding process, word2vec model is used. Besides, the DAFSC-PNBM model uses the Pythagorean Neutrosophic Bonferroni Mean (PNBM) technique for classification. To enhance the SA performance of the PNBM model, the grey wolf optimizer (GWO) model has been applied as a hyperparameter tune process. The experimentation outcome analysis of the DAFSC-PNBM method occurs and the outcomes are investigated under several features. The experimental study indicated the improvement of the DAFSC-PNBM method across the modern techniques

Keywords: Machine Learning; Neutrosophic Set; Pythagorean Neutrosophic; Sentiment analysis; Neutrosophic Logic

1. Introduction

Neutrosophic Logic is a neonate investigation region in which every proposal has been predicted to have the percentage (proportion) of truth in subgroup T, the indeterminacy proportion in subgroup I, and the falsity proportion in subgroup F [1]. Neutrosophic set (NS) is effectively used for unclassified processing of information and validates benefits to handle the data indeterminacy information and is even a model stimulated for classification application and data analysis [2]. NS offers an accurate and efficient manner to describe imbalanced information based on the data attributes [3]. Social networking channels such as Instagram, Twitter, WhatsApp, and Facebook have tempestuous environmental contact, it's authoritative to pass on sensitive understanding about the public's reactions, feelings, and moods on some product, policy, or concept over these social networking channels. To both suppliers and customers, these data can be valued [4].

In some online shopping, customers generally check other people's views about the product [5]. According to the customer's sentiment, the manufacturer is learning about its product's advantages and disadvantages. Even though either business organizations or individuals can obtain benefits from this review, the sheer number of these

DOI: https://doi.org/10.54216/IJNS.250321
Received: June 09, 2024 Revised: August 18, 2024 Accepted: October 14, 2024

sentiments on text data is overwhelming for the users [6]. These new study areas are termed Opinion Mining or Sentiment Analysis (SA). Owing to the huge variety of services and products these days, it has become problematic for users to choose their target product [7]. Product analyses prove to be an extremely helpful reference. Sentiment Analysis plays an important role nowadays, and it is the Natural Language Processing (NLP) approach. SA is a machine learning (ML) technology that searches for text polarity, ranging from positive to negative. ML tools aid in learning how to identify sentiment without human input by training them with emotional samples in text [8]. The sentimental-based assessments expected by the consumers in various forms like audio, text, emoji video, and so on., are classified by using classification techniques in ML models and Lexicon-based methods [9]. The SA approaches are mostly utilized for identifying the emotions that are communicated by the consumer on some type of social media website. An ML model utilizes a particular data type to reply to more questions via the patterns unseen in that data [10].

This study develops a Data Analytics Framework for Sentiment Classification Using the Pythagorean Neutrosophic Bonferroni Mean (DAFSC-PNBM) technique on Product Reviews. Primarily, data preprocessing is performed to increase the product review qualities. For word process of embedding, word2vec model is used. Besides, the DAFSC-PNBM model uses Pythagorean Neutrosophic Bonferroni Mean (PNBM) method for classification. To enhance the SA performance of the PNBM technique, the grey wolf optimizer (GWO) model has been applied as a hyperparameter tune process. The experimentation outcome analysis of the DAFSC-PNBM method takes place and the outcomes are examined under various features.

2. Related Works

El Mrabti et al. [11] introduced an easy-to-use ML approach depending on an intuitive geometrical model for SA of online analyses. Among the expansion of EABLC, dual novel techniques are additionally presented, such as binary classification and binary polyhedral classification, to prevent anomalies and to handle linear non-separable data. Asuquo et al. [12] present a hybrid architecture that depends on ML and lexicon algorithms to train before-seen tweets for predicting the opinions of various novel input tweetings into polarities of neutral, negative, and positive. Tweeting libraries have been utilized to remove tweets on Laptop probes to detect several features and classify sentiments near them into particular polarities. Next data preprocessing, the application in Python employed a Natural Language Processing (NLP) set named TextBlob for assigning polarity and subjectivity text scores. The scores are applied using the ML methods for classifying and analyzing sentiments. Li et al. [13] propose an SA model that associates the e-commerce analysis keyword-made image with a hybrid ML-based approach, in which the Word2VecTextRank can be applied to remove keywords that perform as the inputs for making the connected images by generating Artificial Intelligence (AI).

In [14], a unique model called Systemize Polarity Shift has been introduced in which a flow search-based SVM through the BoW method classifies pre-trained probe comments as neutral, negative, and positive opinions. Furthermore, the recognition of particular products is not concentrated on SA. Gupta and Noliya [15] developed an SA process for product analyses with URLs of specific items as databases. It uses web scrap models for collecting textual information from websites. NLP methods have been used for data preprocessing, with stop word removal and tokenization. Abbas et al. [16] recommended a sophisticated method that uses active learning (AL) based ML models for Flipkart's smartphone consumers' reviewed SA. The suggested investigation accepts difficult data balancing, data cleaning, and TF-IDF feature extraction algorithms, accompanied by 5 AL-based ML techniques, for classifying SA based on product reviews.

3. Methodology

In this work, we have introduced a unique DAFSC-PNBM technique for Product Reviews. The presented DAFSC-PNBM technique mainly aims to determine the nature of sentiments based on product reviews. Fig. 1 represents the complete method of DAFSC-PNBM technique.

DOI: https://doi.org/10.54216/IJNS.250321
Received: June 09, 2024 Revised: August 18, 2024 Accepted: October 14, 2024

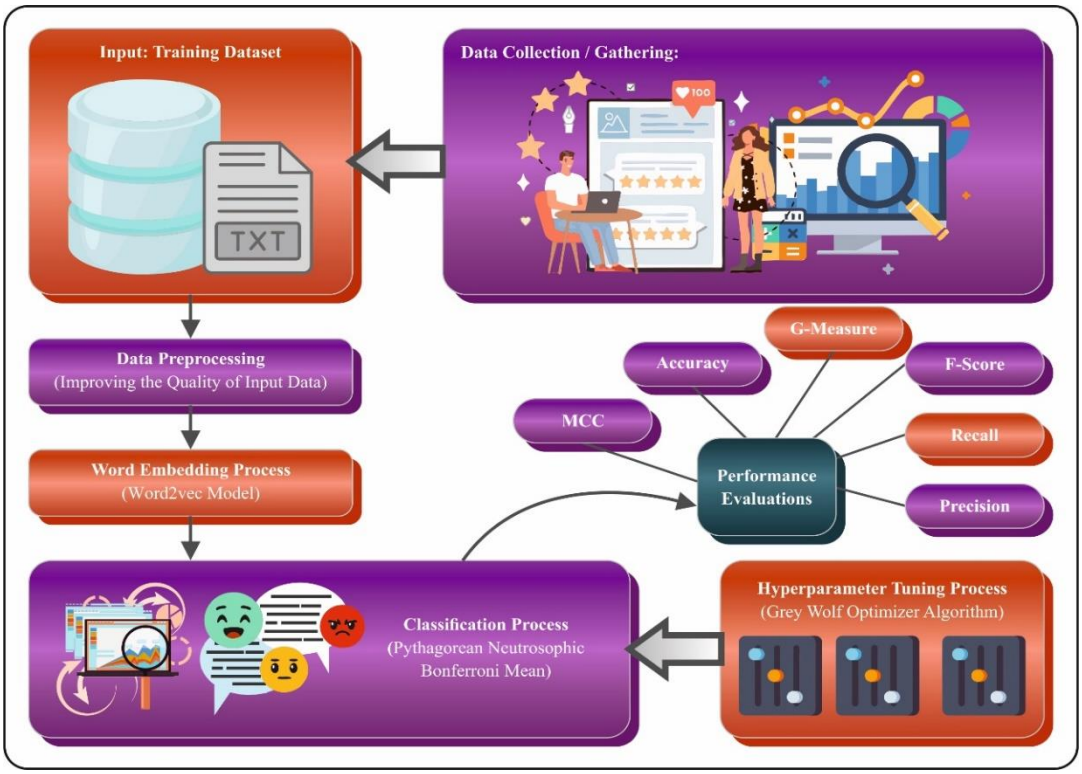

Figure 1. Overall process of DAFSC-PNBM technique

A. Data Preprocessing

Initially, data preprocessing is performed to increase the product review qualities. Data investigation needs data pre-processing to remove redundancies and increase the learning technique of the classification model and accuracy [17]. Redundant data characterize some information that gives minimum or none to forecasting a targeted class, but it improves the feature vector size and offerings redundant computational complexities. Then, the precision of a classifier method has been degraded when irregular or no pre-processing can be performed before encoding. These studies utilized Python's NLP set of tools to pre-process the data. Initially, the text was converted into lowercase, and after that punctuation, links, and HTML tags, are eliminated. Formerly, stemming and lemmatization techniques were applied to clean up the stop words, and lastly, the text was separate.

- Transformation to lowercase: the text is changed to lowercase. These models consider lower upper-case words as dissimilar, manipulating the performance classification and the process of training.
- Elimination of punctuation, URL links, tags, and numbers: They should not contribute to the classification accuracy then they offer no additional meaning for the learning method and improve the space of feature. Hence, removing them helps in reducing the space of features.
- Stemming and Lemmatization: These stages aim to minimize inflectional forms and sometimes derivative-related forms of the word to the usual baseline.
- Elimination of stop words: Stop words are regularly applied words that present no valued data.

B. Word Embedding

For word embedding process, word2vec model is used. This model works on the important principle that the context word usage condenses its intrinsic meaning [18]. Essentially, this method creates a multi-dimensional vector demonstration for every word, preserving either semantic or syntactic associations between words. The vector distances among individual words imitate the human insight of word relationships. This model has presented two main architectures: Skip-gram and Continuous Bag-of-Words (CBOW). CBOW concentrates on forecasting a targeted word depending on the nearby contextual words inside a particular window, while Skip-gram concentrates on forecasting contextual words assumed a targeted word in an assumed contextual window. This architecture is agreed as simple neural networks including input, output, and hidden layers (HL). However, an activation function of Softmax has been utilized for the output layer, Word2Vec also includes algorithm improvements such as hierarchical Softmax and negative sampling for reducing computation complexities.

DOI: https://doi.org/10.54216/IJNS.250321
Received: June 09, 2024 Revised: August 18, 2024 Accepted: October 14, 2024

In this work, we utilized the Word2Vec method application offered by the Gensim library in the programming language of Python. The parameter values are chosen according to our preceding investigation, which intended to recognize suitable settings for optimum outcomes. Using both the Skip-gram structure, we advanced a unique method for the new sentences and 3 further methods for sentences exposed to particular modifications like synonym replacement back translation, and function word deletion. For training the method based on these clear techniques, we gained a vector of words for every obtainable word. We deposited the individual vector of words and their corresponding terms in a binary file format with the Gensim library and the Key Vector class, generating a complete dictionary-like framework. Fig. 2 represents the structure of word2vec model.

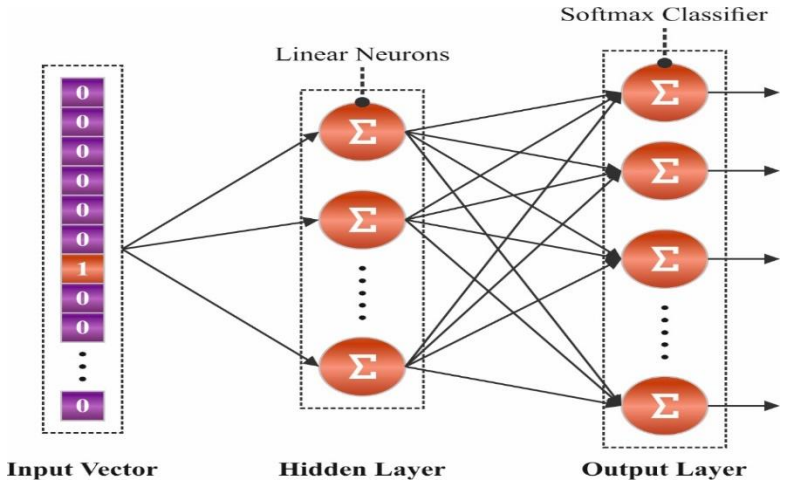

Figure 2. Structure of word2vec

C. Classification using PNBM Model

Besides, the DAFSC-PNBM model uses PNBM model for classification. The simple concept related to the Pythagorean Neutrosophic set (PNS) is explained below [19]. Let X be a non- universal or empty set. The PNS with ψ and κ as dependent membership

$$A = \{\langle x, \psi_A(x), \varsigma_A(x), k_A(x)\rangle | x \epsilon X\}\{\langle \tag{1}$$

Here, $\varsigma_A, \kappa_A, and\ \psi_A$ are indeterminacy, false, and truth memberships respectively.

$$0 \leq \psi^2 + \kappa^2 \leq 1 \tag{2}$$

$$0 \leq \psi^2 + \varsigma^2 + \kappa^2 \leq 2 \tag{3}$$

Assume $\chi_1 = \left(\psi_{x_1}, \varsigma_{x_1}, \kappa_{x_1}\right), x_2 = \left(\psi_{x_2}, \varsigma_{\chi_2}, \kappa_{x_2}\right)$ and $x = (\psi_x, \varsigma_x, \kappa_x)$ as a PNSs, then the operational rules are expressed as follows:

$$x_1 \oplus x_2 = \left(\sqrt{\psi_{x_1}^2 + \psi_{x_2}^2 - \psi_{x_1}^2 \psi_{x_2}^2}, \varsigma_{x_1}\varsigma_{\chi_2}\kappa_{x_1}\kappa_{x_2}\right) \tag{4}$$

$$x_1 \otimes x_2 = \left(\psi_{\chi_1}\psi_{\chi_2}, \varsigma_{\chi_1} + \varsigma_{\chi_2} - \varsigma_{\chi_1}\varsigma_{\chi_2}\sqrt{\kappa_{\chi_1}^2 + \kappa_{\chi_2}^2 - \kappa_{\chi_1}^2\kappa_{\chi_2}^2}\right) \tag{5}$$

$$\mu x = \left(\sqrt{1 - (1 - \psi_\chi^2)^\mu}\sigma^\mu \kappa_\chi^\mu\right) where \mu \in \Re\ and\ \mu \geq 0 \tag{6}$$

$$\chi^\mu = \left(\psi_x^\mu, 1 - \left(1 - \varsigma_\chi\right)^{\mu_t}\sqrt{1 - (1 - \kappa_\chi^2)^\mu}\right)\ where\ \mu \in \Re\ and\ \mu \geq 0. \tag{7}$$

Consider $p, q \geq 0$ with $\chi_i = (\psi_i(x), \sigma_i\kappa_i$ whereas $(i = 1,2,3, \dots, n)$ is a PNS, where PNBM is formulated as follows:

$$PNBM\ (x_1, x_2, \dots, x_n)^{p,q} = \left(\frac{1}{n(n-1)} \oplus_{\substack{i,j=1 \\ i \neq j}}^{n} \left(x_i^p \otimes x_j^p\right)\right)^{\frac{1}{p+q}} \tag{8}$$

DOI: https://doi.org/10.54216/IJNS.250321
Received: June 09, 2024 Revised: August 18, 2024 Accepted: October 14, 2024

Step1: Generate the Direct-Relation Matrix, X^k

The matrix has been prepared by PNS to describe the straight relationship depending upon the preferences of creating decisions and evaluates the preference as a non-negative matrix, $X^k = [x_{ij}^k]_{n\times n}$, whereas $1 \le k \le m$.

Step2: Get the Collective Direct-Relation Matrix A.

Depending upon Eq. (8), PN-NWBM unites straight relationship matrix $X^k = \{X^1, X^2, \dots, X^m\}$ into a joined decision matrix $A =$ while $a_{ij} = (\psi_{ij}, \varsigma_{ij}, \kappa_{ij})$.

Step3: Deneutrosophication into crisp matrix B.

The aggregated matrix $A = [a_{ij}]_{m\times n}$ into matrix of crisp, B has been deneutrosophicated in Eq. (9).

$$B = \frac{\psi_A(x) + \varsigma_A(x) + \kappa_A(x)}{3} \tag{9}$$

Step4: Regularize the matrix to consistent matrix Z. Whereas, Eq. (10) was employed to regularize the procedure of matrix.

$$Z = \frac{B}{s} \tag{10}$$

While $s = \max \sum_{j=1}^{n} b_{ij}$ and every element in matrix Z fulfills with $0 \le z_{ij} < 1$.

Step5: Build the Total-Influence Matrix T. Make the matrix of influence utilizing the below-mentioned formulation.

$$T = Z(l - Z)^{-1} \tag{11}$$

Here, l denotes the matrix of identity.

Step6: Calculate the Amount of Rows and Columns

The R and C vectors are the number of rows and columns. Where the complete influence matrix T was well-defined.

$$R = [\widetilde{r_l}]_{n\times 1} = [\sum_{j=1}^{n} t_{ij}]_{n\times 1} \tag{12}$$

$$C = [\widetilde{c_l}]_{1\times n} = [\sum_{i=1}^{n} t_{ij}]_{1\times n}^{T} \tag{13}$$

Step7: Network Relationship Map (NRM), the Value of Threshold

The dataset graph was $(R + C_t R - C)$. $R + C$ and $R - C$ denotes vertical and horizontal axis.

D. Hyperparameter Tuning

To increase the SA performance of the PNBM technique, the GWO model has been applied as a hyperparameter tune process. Mirjalili et al. proposed GWO algorithm based on the social intelligence of grey wolves living in groups containing 5 to 12 individuals [20]. The algorithm integrates four different levels: α, β, δ, and ω to replicate the observed leadership hierarchy in GWO. In this work, the hunting process is composed of α, β, and δ wolves, with ω wolves following the commands. These behaviors are expressed in the following:

$$\vec{\mathrm{X}}(t+1) = \vec{\mathrm{X}}_{\mathrm{p}}(t) + \vec{A}.\vec{D} \tag{14}$$

In Eq. (14), $\vec{\mathrm{A}}$ and $\vec{\mathrm{D}}$ are the coefficient vector, the vector of prey's location is represented as $\vec{\mathrm{X}}_{\mathrm{p}}$, and the wolf location in a d-dimension space is indicated as X, D signifies the number of variables, and the variable (t) shows the iteration count:

$$\vec{D} = \left|\vec{C}.\vec{\mathrm{X}}_{\mathrm{p}}(t) - \overline{X}(t)\right| \tag{15}$$

$$\vec{A} = 2\vec{a}.\overrightarrow{r_1} - \vec{a} \tag{16}$$

$$\vec{C} = 2.\overrightarrow{r_2} \tag{17}$$

DOI: https://doi.org/10.54216/IJNS.250321
Received: June 09, 2024 Revised: August 18, 2024 Accepted: October 14, 2024

Where the above equation is used to characterize $\vec{A}$ and $\vec{C}$, where r_1, r_2 are randomly selected in the interval of [0,1]. During the iteration process, the component of a is linearly dropped from 2 to 0. By randomly changing its position around the prey, a grey wolf gets closer to the prey using Eq. (15).

The x_1, x_2, and x_3 values are computed and defined by:

$$\overrightarrow{x_1} = \overrightarrow{X_\alpha} - A_1.(\overrightarrow{D_\alpha}),$$
$$\overrightarrow{x_2} = \overrightarrow{X_\beta} - A_2.(\overrightarrow{D_\beta}),$$
$$\overrightarrow{x_3} = \overrightarrow{X_\delta} - A_3.(\overrightarrow{D_\delta}) \quad (18)$$

In the swarm, the vectors $\overrightarrow{x_1}, \overrightarrow{x_2}$, and $\overrightarrow{x_3}$ are the three leading wolves (solution) at the given iteration t. These values are defined by following the calculations for A_1, A_2, and A_3. Furthermore, the D_α, D_β, and D_δ vectors are calculated by the following expression:

$$\overrightarrow{D_\alpha} =\rightarrow |\overrightarrow{C_1}.\overrightarrow{X_\alpha} - \vec{X}|,$$
$$\overrightarrow{D_\beta} = |\overrightarrow{C_2}.\overrightarrow{X_\beta} - \vec{X}|,$$
$$\overrightarrow{D_\delta} = |\overrightarrow{C_3}.\overrightarrow{X_\delta} - \vec{X}| \quad (19)$$

The vectors $\overrightarrow{C_1}, \overrightarrow{C_2}$, and $\overrightarrow{C_3}$ are defined by performing calculations.

The $\vec{a}$ vector, the key element for balancing exploration and exploitation in the GWO algorithm. As the number of iterations increases, the vector value is linearly decreased from 2 to 0 for all the dimensions:

$$\vec{\alpha} = 2 - t.\frac{2}{\max_i ter} \quad (20)$$

Here, the t variable denotes the existing iteration count, "ter" shows the overall amount of iterations for the optimization method. The GWO approach obtains a fitness function (FF) to gain higher classifier performance. It defines a positive integer to symbolize the greater performance of the candidate solutions. In this section, the minimization of the classifier rate of error can be determined for the FF, as specified in equation. (21).

$$fitness(x_i) = ClassifierErrorRate(x_i)$$
$$= \frac{no\ of\ misclassified\ samples}{Total\ no\ of\ samples} * 100 \quad (21)$$

4. Result Analysis and Discussion

The performance assessment of the DAFSC-PNBM model is investigated using the Amazon fine food reviews dataset [21]. The dataset has 35000 reviews under five ratings as indicated in Table 1. Table 2 represents samples of the dataset.

Table 1: Details on Dataset

Ratings	Number Reviews
Rating-1	7000
Rating-2	7000
Rating-3	7000
Rating-4	7000
Rating-5	7000
Total reviews	35000

Table 2: Samples of the Dataset

DOI: https://doi.org/10.54216/IJNS.250321
Received: June 09, 2024 Revised: August 18, 2024 Accepted: October 14, 2024

Ratings	Text Summary
Rating-5	Good Quality Dog Food
Rating-1	Not as Advertised
Rating-4	"Delight" says it all
Rating-2	Cough Medicine
Rating-5	Great taffy

Fig. 3 provides the classifier results of the DAFSC-PNBM technique on the test database. Figs. 3a-3b demonstrates the confusion matrix with correct identification and classification of each 5 number of classes under a 70:30 TRAP/TESP. Fig. 3c illustrates the study of PR, which stated higher values through every class. Finally, Fig. 3d depicts the investigation of ROC, illustrating efficient performances with better values of ROC for various numbers of classes.

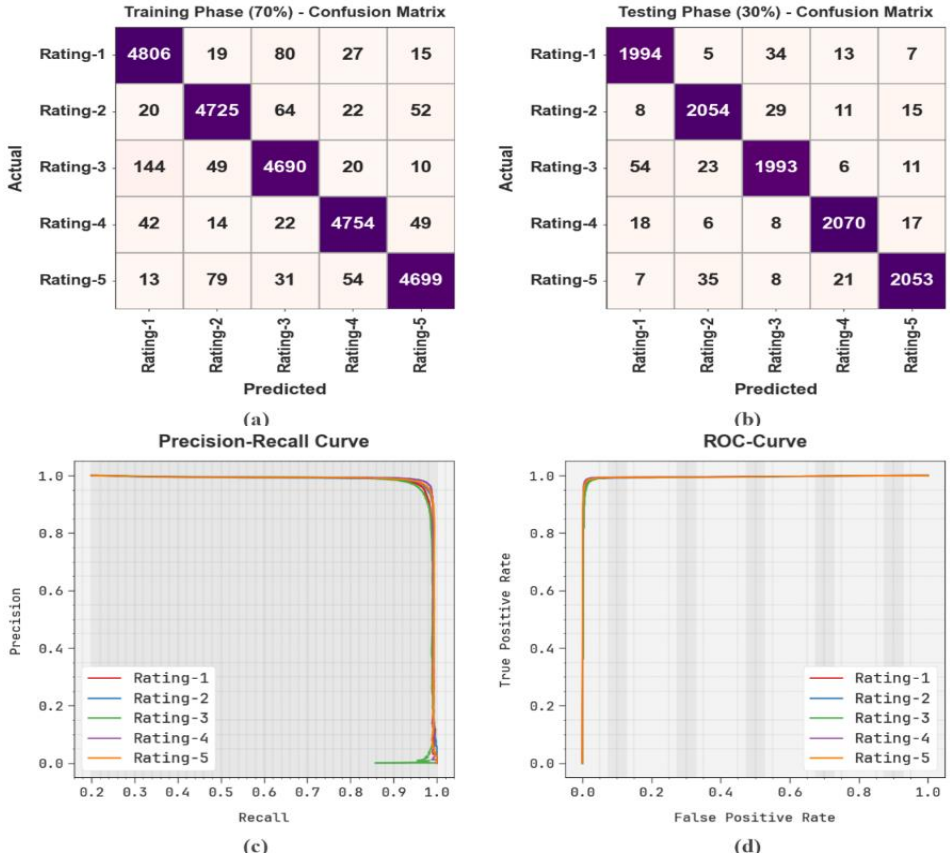

Figure 3. Classifier outcome of (a-b) 70% and 30% confusion matrices and (c-d) PR and ROC curves

Table 3 portrays the classifier outcomes of the DAFSC-PNBM method under 70%TRAP and 30%TESP. The results displayed the better achievement of the DAFSC-PNBM method on product reviews.

In Fig. 4, the average consequences provided by the DAFSC-PNBM approach under 70% of TRAP are underlined. On 70%TRAP, the DAFSC-PNBM approach achieves an average $accu_y$ of 98.65%, $prec_n$ of 96.64%, $reca_l$ of 96.63%, $F1_{score}$ of 96.63%, MCC of 95.79% and $G_{Measure}$ of 96.63%.

In Fig. 5, the average performances given by the DAFSC-PNBM system on 30% of TESP are underscored. On 30%TESP, the DAFSC-PNBM process reaches an average $accu_y$ of 98.72%, $prec_n$ of 96.79%, $reca_l$ of 96.80%, $F1_{score}$ of 96.79%, MCC of 96.00% and $G_{Measure}$ of 96.80%.

Table 3: Classifier outcome of DAFSC-PNBM technique under 70%TRAP and 30%TESP

Classes	$Accu_y$	$Prec_n$	$Reca_l$	F_{score}	MCC	$G_{Measure}$

DOI: https://doi.org/10.54216/IJNS.250321
Received: June 09, 2024 Revised: August 18, 2024 Accepted: October 14, 2024

TRAP (70%)						
Rating-1	98.53	95.64	97.15	96.39	95.47	96.39
Rating-2	98.70	96.70	96.76	96.73	95.92	96.73
Rating-3	98.29	95.97	95.46	95.71	94.64	95.71
Rating-4	98.98	97.48	97.40	97.44	96.80	97.44
Rating-5	98.76	97.39	96.37	96.88	96.11	96.88
Average	98.65	96.64	96.63	96.63	95.79	96.63
TESP (30%)						
Rating-1	98.61	95.82	97.13	96.47	95.61	96.47
Rating-2	98.74	96.75	97.02	96.89	96.10	96.89
Rating-3	98.35	96.19	95.50	95.84	94.81	95.84
Rating-4	99.05	97.60	97.69	97.64	97.04	97.64
Rating-5	98.85	97.62	96.66	97.14	96.42	97.14
Average	98.72	96.79	96.80	96.79	96.00	96.80

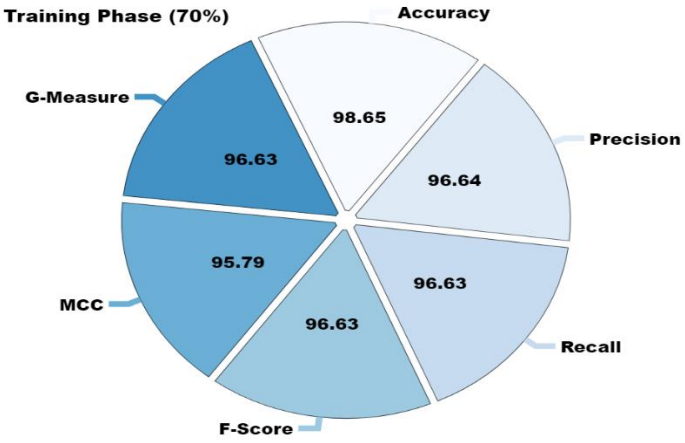

Figure 4. Average of DAFSC-PNBM technique under 70%TRAP

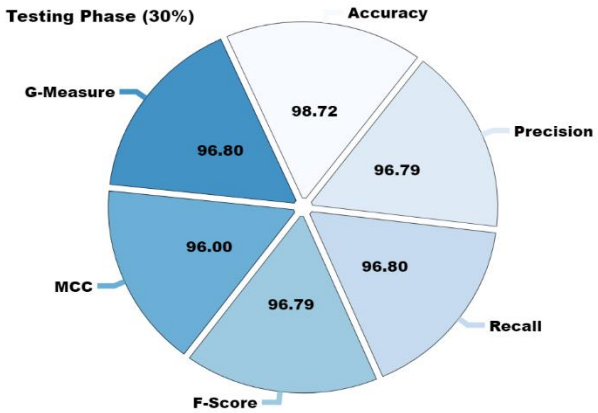

Figure 5. Average of DAFSC-PNBM technique under 30%TESP

DOI: https://doi.org/10.54216/IJNS.250321
Received: June 09, 2024 Revised: August 18, 2024 Accepted: October 14, 2024

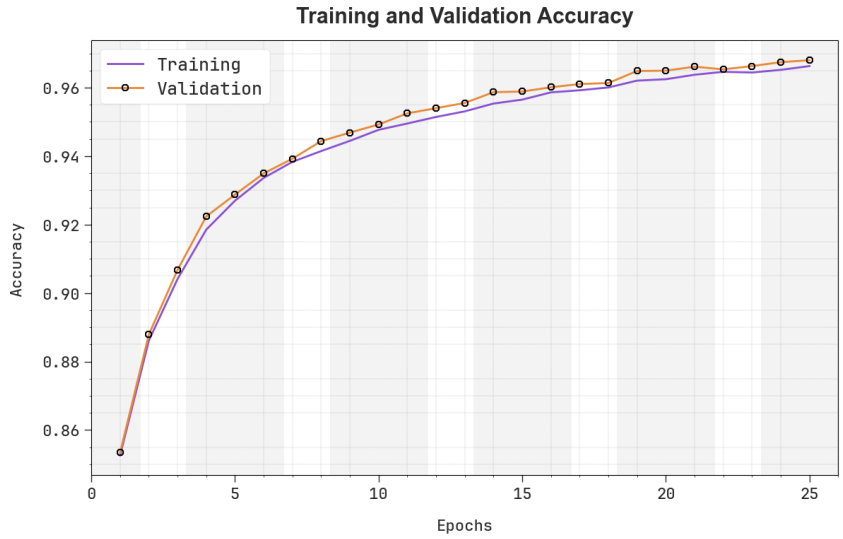

Figure 6. $Accu_y$ Curve of DAFSC-PNBM technique

In Fig. 6, the training (TRA) and validation (VLA) accuracy results of the DAFSC-PNBM approach are described. The correct consequences are calculated for 0-25 number of epochs. The figure underlines that the TRA and VLA accuracy outcomes determine an enhanced tendency that informed the proficiency of the DAFSC-PNBM technique with superior performance across different numbers of iterations. Furthermore, the TRA and VLA accuracy remains nearer over the epoch counts, which illustrates lower minimal overfitting and displays the better results of the DAFSC-PNBM methodology, assuring steady prediction on hidden instances.

In Fig. 7, the TRA and VLA loss graphs of the DAFSC-PNBM process are demonstrated. The loss performance is calculated for 0-25 number of epochs. It is portrayed that the TRA and VLA accuracy values indicate a reducing tendency, stating the proficiency of the DAFSC-PNBM approach in balancing a trade-off between data fitting and generalization. The steady decline in loss outcomes additionally assurances the superior results of the DAFSC-PNBM technique and tuning the prediction performances on time.

Figure 7. Loss curve of DAFSC-PNBM technique

The comparison study of DAFSC-PNBM model with recent techniques can be portrayed in Table 4 and Fig. 8 [22]. The experimental outcomes depicted that the DAFSC-PNBM method outshined improved performances. According to $accu_y$, the DAFSC-PNBM system has greater $accu_y$ of 98.72% whereas the SAPR-EGODL methodology, LSTM approach, LSTM-ReLU system, Deep LSTM algorithm, Ensemble classifier, GoogleNet model, and Custom DLL methodologies have minimum $accu_y$ of 97.90%, 92.55%, 89.65%, 93.13%, 93.24%, 91.88%, and 88.65%, correspondingly. Similarly, depends on $prec_n$, the DAFSC-PNBM process has better $prec_n$ of 96.79% but the SAPR-EGODL methodology, LSTM approach, LSTM-ReLU system, Deep LSTM algorithm,

DOI: https://doi.org/10.54216/IJNS.250321
Received: June 09, 2024 Revised: August 18, 2024 Accepted: October 14, 2024

Ensemble classifier, GoogleNet model, and Custom DLL systems have the least $prec_n$ of 95.96%, 91.23%, 92.02%, 94.14%, 91.52%, 88.83%, and 90.39%, individually. Lastly, based upon F_{score}, the DAFSC-PNBM methodology has a greater F_{score} of 96.79% however, the SAPR-EGODL methodology, LSTM approach, LSTM-ReLU system, Deep LSTM algorithm, Ensemble classifier, GoogleNet model, and Custom DLL methodologies have the least F_{score} of 96.09%, 90.16%, 93.02%, 92.38%, 90.43%, 90.18%, and 92.09%, appropriately.

Table 4: Comparative analysis of DAFSC-PNBM technique with existing models

Methods	$Accu_y$	$Prec_n$	$Reca_l$	F_{score}
DAFSC-PNBM	98.72	96.79	96.80	96.79
SAPR-EGODL	97.90	95.96	95.90	96.09
LSTM Classifier	92.55	91.23	93.50	90.16
LSTM-ReLU	89.65	92.02	93.32	93.02
Deep LSTM	93.13	94.14	90.80	92.38
Ensemble Model	93.24	91.52	94.27	90.43
GoogleNet Method	91.88	88.83	92.97	90.18
Custom DLL Model	88.65	90.39	94.42	92.09

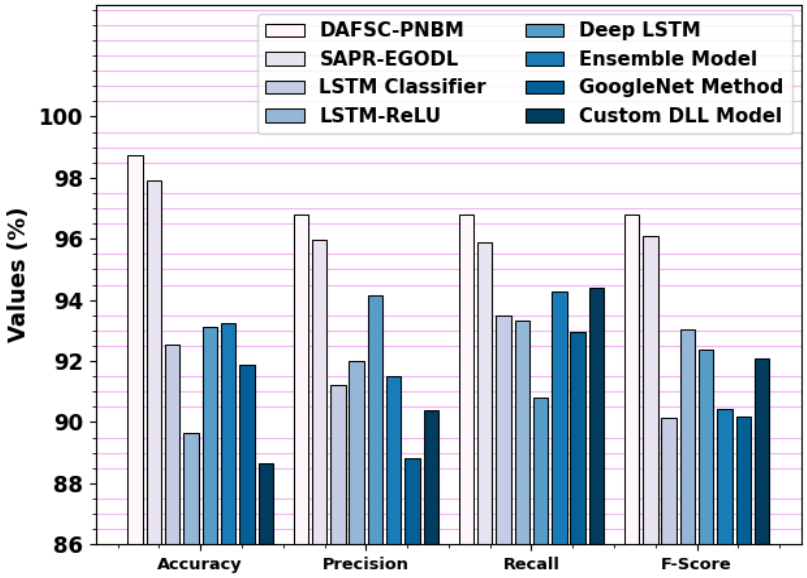

Figure 8. Comparative analysis of DAFSC-PNBM technique with existing models

5. Conclusion

In this article, we have developed a novel DAFSC-PNBM model for Product Reviews. The presented DAFSC-PNBM method mainly aims to determine the nature of sentiments based on product reviews. Initially, data preprocessing is performed to increase the product review qualities. For the word process of embedding, word2vec model is used. Besides, the DAFSC-PNBM model uses PNBM model for classification. To increase the SA performance of the PNBM technique, the GWO model has been applied as a hyperparameter tune process. The experimentation outcome analysis of the DAFSC-PNBM method occurs and the outcomes are studied under several features. The experimental study indicated the improvement of the DAFSC-PNBM method above the modern techniques.

Funding: "This research received no external funding"

Conflicts of Interest: "The authors declare no conflict of interest."

References

DOI: https://doi.org/10.54216/IJNS.250321
Received: June 09, 2024 Revised: August 18, 2024 Accepted: October 14, 2024

[1] Abobala, M., "A Study of AH-Substructures in n-Refined Neutrosophic Vector Spaces", International Journal of Neutrosophic Science", Vol. 9, pp.74-85, 2020.
[2] Khalid, M., Khalid, N.A. and Iqbal, R., 2020. MBJ-neutrosophic T-ideal on B-algebra. International Journal of Neutrosophic Science, 1(1), pp.29-39.
[3] Saheb, A.H. and Buti, R.H., 2024. A Specific Category of Harmonic Functions Characterized By a Generalized Komatu Operator in Conjunction With The (RK) Integral Operator and Applications to Neutrosophic Complex Field. Full Length Article, 23(3), pp.44-4.
[4] Noaman, I.A.R., Hasan, A.H. and Ahmed, S.M., 2024. Optimizing Weibull Distribution Parameters for Improved Earthquake Modeling in Japan: A Comparative Approach. International Journal of Neutrosophic Science, 24(1), pp.65-5.
[5] Alhamido, R., and Abobala, M., "AH-Substructures in Neutrosophic Modules", International Journal of Neutrosophic Science, Vol. 7, pp. 79-86, 2020.
[6] Xu, F., Pan, Z. and Xia, R., 2020. E-commerce product review sentiment classification based on a naïve Bayes continuous learning framework. Information Processing & Management, 57(5), p.102221.
[7] Vijayaragavan, P., Ponnusamy, R. and Aramudhan, M., 2020. An optimal support vector machine based classification model for sentimental analysis of online product reviews. Future Generation Computer Systems, 111, pp.234-240.
[8] Taherdoost, H., & Madanchian, M. (2023). Artificial intelligence and sentiment analysis: A review in competitive research. Computers, 12(2), 37.
[9] Mutinda, J., Mwangi, W., & Okeyo, G. (2023). Sentiment analysis of text reviews using lexicon-enhanced bert embedding (LeBERT) model with convolutional neural network. Applied Sciences, 13(3), 1445.
[10] Patruni, M.R., Angadi, A., Gorripati, S.K., & Saraswathi, P. (2023). Artificial Intelligence Techniques in Text and Sentiment Analysis. In Handbook of Research on Artificial Intelligence Applications in Literary Works and Social Media (pp. 171-191).
[11] El Mrabti, S., Jaouad, E.M., Hachmoud, A. and Lazaar, M., 2024. An explainable machine learning model for sentiment analysis of online reviews. Knowledge-Based Systems, p.112348.
[12] Asuquo, D., Attai, K., Usip, P., George, U. and Osang, F., 2023, November. Aspect-Based Sentiment Classification of Online Product Reviews Using Hybrid Lexicon-Machine Learning Approach. In International Conference on Applied Machine Learning and Data Analytics (pp. 124-143). Cham: Springer Nature Switzerland.
[13] Li, J., Huang, Y., Lu, Y., Wang, L., Ren, Y. and Chen, R., 2024. Sentiment Analysis Using E-Commerce Review Keyword-Generated Image with a Hybrid Machine Learning-Based Model. Computers, Materials & Continua, 80(1).
[14] Purohit, A. and Patheja, P.S., 2023. Product review opinion based on sentiment analysis. Journal of Intelligent & Fuzzy Systems, 44(2), pp.3153-3169.
[15] Gupta, S. and Noliya, A., 2024. URL-Based Sentiment Analysis of Product Reviews Using LSTM and GRU. Procedia Computer Science, 235, pp.1814-1823.
[16] Abbas, S., Boulila, W., Driss, M., Sampedro, G.A., Abisado, M. and Almadhor, A., 2023. Active learning empowered sentiment analysis: An approach for optimizing smartphone customer's review sentiment classification. IEEE Transactions on Consumer Electronics.
[17] Elangovan, D. and Subedha, V., 2023. Adaptive Particle Grey Wolf optimizer with deep learning-based sentiment analysis on online product reviews. Engineering, Technology & Applied Science Research, 13(3), pp.10989-10993.
[18] Kapusta, J., Držík, D., Šteflovič, K. and Nagy, K.S., 2024. Text Data Augmentation Techniques for Word Embeddings in Fake News Classification. IEEE Access, 12, pp.31538-31550.
[19] Ahmed, A., Badawy, M. and Gubarah, G.F., 2024. Modelling of Green Human Resource Management using Pythagorean Neutrosophic Bonferroni Mean Approach. International Journal of Neutrosophic Science, 24(2), pp.258-58.
[20] Sumiea, E.H.H., Abdulkadir, S.J., Ragab, M.G., Al-Selwi, S.M., Fati, S.M., Alqushaibi, A. and Alhussian, H., 2023. Enhanced Deep Deterministic Policy Gradient Algorithm using Grey Wolf Optimizer for continuous Control Tasks. IEEE Access.
[21] https://www.kaggle.com/datasets/snap/amazon-fine-food-reviews
[22] Al-Fatlawy, R.R., 2023. Computational Intelligence-based Data Analytics for Sentiment Classification on Product Reviews. Journal of Smart Internet of Things, 2023(2), pp.84-104.

DOI: https://doi.org/10.54216/IJNS.250321
Received: June 09, 2024 Revised: August 18, 2024 Accepted: October 14, 2024