

Indian journal of Engineering

To Cite:

Vinoth R, Sudha K, Lakshmi Priya C, Udhayashankar S. A Polymorphic Graph Attention Bayesian Asymmetric Quantized Neural Network - Based Approach for Sentiment Analysis of Movie Reviews. *Indian Journal of Engineering*, 2026, 23, e12ije1729
doi: <https://doi.org/10.54905/diss.v23i60.e12ije1729>

Author Affiliation:

¹Dept of CSE, RMK College of Engineering and Technology, India, Email: vinothcse@rmkcet.ac.in

²Dept of CSBS, RMD Engineering College, India, Email: sudhak14@gmail.com

³Dept of CSE, SA Engineering College, India, Email: lakshmi priya@saec.ac.in

⁴Dept of AI&DS, Vel Tech Multi Tech Dr. Rangarajan Dr. Sakunthala Engineering College, India, Email: shankarinfotech07@gmail.com

* Corresponding author:

Vinoth R,
Dept of CSE, RMK College of Engineering and Technology, India, Email: vinothcse@rmkcet.ac.in

Peer-Review History

Received: 29 December 2025

Reviewed & Revised: 18/January/2026 to 12/August/2026

Accepted: 27 August 2026

Published: 15 September 2026

Peer-Review Model

External peer-review was done through double-blind method.

Indian Journal of Engineering
pISSN 2319-7757; eISSN 2319-7765



© The Author(s) 2026. Open Access. This article is licensed under a [Creative Commons Attribution License 4.0 \(CC BY 4.0\)](https://creativecommons.org/licenses/by/4.0/), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

A Polymorphic Graph Attention Bayesian Asymmetric Quantized Neural Network-Based Approach for Sentiment Analysis of Movie Reviews

Vinoth R^{1*}, Sudha K², Lakshmi priya C³, Udhayashankar S⁴

ABSTRACT

Analyzing user sentiment from movie reviews requires deep learning techniques because the large volume of reviews makes manual analysis impractical. Sentiment analysis is now essential for understanding public opinion because people use online platforms to share vast amounts of data and opinions. Traditional approaches are capable of dealing with limited information, while big and complex data need sophisticated techniques. In this study, a novel approach is introduced for movie review classification based on sentiments using Polymorphic Graph Attention Bayesian Asymmetric Quantized Neural Network with Emperor Penguin Optimization (PGABAQNNet-EPO). Performance analysis of the proposed model is done based on three commonly used databases, which are MovieLens, IMDb, and Rotten Tomatoes. The results show a high accuracy rate of 99.1%, 99.4%, and 99.3% for MovieLens, IMDb, and Rotten Tomatoes, respectively, and an F1-score of more than 99%, the results demonstrate outstanding performance. The model achieves rapid computation time of 0.02 second along with low error values. The algorithm shows strong capabilities for sentiment analysis and provides a reliable approach to classify movie reviews with high speed.

Keywords: Bayesian asymmetric quantized neural networks, Dual-domain Cswin transformer, Emperor Penguin Optimization, natural language processing, Polymorphic graph attention.

1. INTRODUCTION

Many people use their spare time to watch movies and create reviews during the digital era. Sentiment analysis helps in natural language processing by understanding which emotion the texts convey. There are many subjective data streams that have to be analyzed for obtaining information about the products, firms, individuals and events (Pavitha et al., 2022). The emotional state of the text has to be determined to see whether it is neutral, negative, or positive. Before

watching the movie, it is important to know some critical information (Tan et al., 2022). Opinion mining and sentiment analysis are often used interchangeably. However, they are different and cannot be interchanged even though they are closely related (Mutinda et al., 2023). It is common practice for opinion mining to establish the opinion holder and the subject of that opinion. Irrespective of the owner or motive behind the opinion, sentiment analysis, on the other hand, involves the determination of the sentiment or emotion conveyed through the text (Roy and Dutta, 2022). Sentiment analysis is at times used to refer to opinion mining due to practical reasons.

There are various approaches to dealing with sentiment analysis tasks (Sahu et al., 2022). They utilize different techniques in which machine learning is used along with word definition. Approaches that use dictionaries and existing bodies of texts determine the sentiment of the text. Statistical text analysis deals with large amounts of text, but dictionary-based methods rely on tools such as SentiWordNet and WordNet for operations (Wankhade et al., 2022). However, ML relies on learning processes such as supervised and unsupervised learning to identify patterns (Zhang and Zhang, 2022). Sentiment analysis has made substantial use of traditional machine learning methods, including Random Forest (RF), Decision Tree, Naïve Bayes (NB), and Support Vector Machine (SVM) (Mazhar et al., 2022). However, machine learning algorithms fail to cope with the sheer volume and complexity of big data due to its exponential growth (Shaddeli et al., 2022).

Big data has led to many alterations in the methods of sentiment analysis. Because there is so much user-generated data accessible, sentiment analysis requires realistic methods (Lee et al., 2022). Conventional machine learning techniques work well with datasets that are modest to moderate in size, but not with enormous volumes of data. This restriction has led to the rise in popularity of deep learning, a branch of machine learning that excels at handling massive datasets (Selvakumar & Lakshmanan, 2022). A deep learning model consists of a number of layers, including DNNs, compared to conventional neural networks where the number of layers is finite (Bhuvaneshwari et al., 2022). Deep learning is a good choice of algorithm for dealing with big and complex datasets through its systematic design that allows the automatic learning of features from data at various levels (Zulqarnain et al., 2024). Deep learning is unique in sentiment analysis and opinion mining because it can learn features from data. Deep learning models assist professionals in interpreting huge collections of texts while detecting intricate patterns and making accurate predictions on sentiments (Mu and Wu, 2023). Data on client sentiment influence product development and client relationship management processes in order to make better business decisions (Dashtipour et al., 2022). The goal of this study is to employ deep learning technology to classify movie reviews into favorable and adverse groups in order to address the issues with sentiment analysis.

Key contribution

- A new Polymorphic Graph Attention Bayesian Asymmetric Quantized Neural Network with Emperor Penguin Optimization (PGABAQNNNet-EPO) is designed to handle sentiment analysis in movie review classification.
- The movie review data are initially gathered from the Rotten Tomatoes (RT), IMDb, and MovieLens datasets. NLP techniques are then used to preprocess the data.
- After the data is prepared, feature extraction is performed using a Dual-Domain CSWin Transformer.
- The sentiment categorization process for movie reviews uses the Polymorphic Graph Attention Bayesian Asymmetric Quantized Neural Networks (PGABAQNNNet).
- Optimization of the loss function of PGABAQNNNet model through the application of Emperor Penguin Optimization (EPO) technique improves classification performance.

The study is organized as follows: A survey of the body of research on sentiment analysis is provided in Section 2. The methodology for carrying out the research is explained in Section 3. Section 4 presents the outcomes of the proposed method. Section 5 provides the rationale for the study.

2. LITERATURE SURVEY

Bazargani et al., (2026) suggested a deep neural network model that suggests films based on their semantic significance in online networks. The goal is to use customer loyalty research to overcome the cold-start issue. The three steps of the process include: This study relies on three core aspects to assist in the identification of recommended items for users including grouping users via a deep neural network and discovery of new user neighbors and predicting product ratings using adjacency matrices. The suggested recommendation technique performs more effectively than the existing methods at making recommendations for movies using the Movielens dataset.

Cui et al., (2024) presented the Sentiment-Aware Knowledge Graph Convolutional Neural Network (RAKCR), a new architecture that improves recommendation performance using user ratings and knowledge graphs. RAKCR differs from conventional approaches in its use of sentiment perception in knowledge graph representation for the purpose of representing consumer decisions regarding product features. It constructs improved user and product feature representations through sentiment-aware neighborhood data. The recommendation process can be improved using RAKCR since it generates more effective item and user profiles.

Latrech et al., (2024) created Twit-CoFiD to provide product recommendations using sentiment analysis of tweets. This recommendation system uses the technology of Deep Neural Networks to cluster users depending on their demographic information. The algorithm takes several sets of population data and analyzes sentiments on Twitter through collaborative filtering techniques. The Twit-CoFiD framework utilizes customized product choices depending on the emotions expressed by users through their interactions on Twitter. The system enhances the personalized recommendation task by concurrently analyzing the age range and emotions of the users on Twitter.

Shrivastava and Kumar (2025) published their new movie recommendation system in 2024. The suggested approach makes use of Hesitant Fuzzy Clustering and Convolutional Spiking Neural Networks. Two benchmark datasets have been used by the proposed system as the input. The system employs Hesitant Fuzzy Bi-objective Clustering using Ternary Patterns and Discrete Wavelet Transform for the conversion of data from benchmark datasets into attribute-based clusters. During movie selection, the deep neural network uses processed user data from Deep Convolutional Spiking Neural Networks to make preference predictions. The proposed method shows better results in personalized movie recommendations existing to current methods GHRS-MRS and MT-ML-MRS.

Nagy et al., (2023) presented a System for securing federated learning algorithms which implement LDP using bitwise coding and feature hashing techniques. Updates of local machine learning models are studied using randomized response mechanism to develop a secure collaborative learning framework. The use of natural language data turns out to be the most useful way of achieving correct results among other data sources. The client application may ensure the security of user data by applying text input encoding using the rolling-hash method without decreasing the efficiency of the operation.

Ibrahim et al., (2023) revealed Collaborative Hybrid Neural Filtering (HNCF). This strategy uses interaction modeling and deep learning algorithms to address the initial experiences of customers. HNCF creates an integrated ranking by using deep multivariate ratings approach through combination of data gathered from a variety of external sources, including votes, likes, stars, and sentiment analysis. Hierarchical User and Product Attention using BiLSTM (HUAPA), Neural Sentiment Classifier (NSC), Deep Collaborative Filtering (DCF), and DMR constitute the four main elements of HNCF. The attitude of users is recorded by NSC, DMR makes use of rating data on a massive scale, HUAPA encodes the preferences of users and the features of products, while DCF encodes interactions between the users and products.

Liu & Miyazaki (2023) introduced movie recommendations using Knowledge-Aware Attentional Neural Networks (KANN). When evaluating films, KANN finds knowledge entities that show important user-film interactions at the knowledge level. In contrast to conventional approaches using word embeddings, KANN utilizes knowledge graph representations to make it possible to give explanations at the knowledge level and prevent phrase-splitting misinterpretation. The values for inner and outer attention between review-user/movie pairs are stored before generating the latent representations. KANN outperformed other cutting-edge models in its assessments utilizing the IMDb and Amazon datasets when it came to forecasting the ratings and click-through rates in certain situations when there were limited reviews.

Danyal et al., (2024a) investigated sentiment analysis of movie reviews by extracting subjective information such as the perspective of the reviewer, pros and cons of the film, and recommendations based on advanced language models such as XLNet and BERT. These models improve performance in many NLP tasks by learning the bidirectional relations between words. XLNet uses permutation language modeling while BERT uses masked language modeling. Preprocessing methodologies such as data cleaning, de-duplication, and lemmatization enable the comparison of XLNet and BERT models with well-known baseline methodologies. Sentiment analysis, through general sentiment, would prove helpful in predicting movie performance, as seen by the results, where XLNet had the highest accuracy. The literature review is summarized in Table 1.

Table 1. Summary of literature review

Reference	Method	Advantage	Limitation
(Bazargani et al., 2024)	GDNN	Improves suggestion accuracy by grouping devoted users.	Depends mostly on data from devoted users, which can restrict generalization to new users.
(Cui et al., 2024)	RAKCR	Makes precise, individualized suggestions by utilizing sentiment analysis and knowledge graphs.	Computational expenses may rise due to the complexity of merging knowledge graphs with sentiment data.
(Latrech et al., 2024)	Twit-CoFiD	Improves suggestions by combining sentiment analysis from Twitter with demographic filters.	Applicability in settings with little social media data may be hampered by reliance on Twitter data.
(Shrivastava and Kumar, 2024)	DCSNN	Improves recommendation accuracy by using fuzzy clustering and deep learning.	A complex model structure could need a lot of processing power.
(Nagy et al., 2024)	FL	Allows for effective model training using natural language data while protecting user privacy.	May struggle to balance model accuracy and privacy, particularly for small datasets.
(Ibrahim et al., 2023)	HUAPA-DCF	Integrates information from several sources to provide a thorough rating and increase the precision of recommendations.	The complexity of integrating multiple data sources could result in higher computational demands.
(Liu & Miyazaki, 2023)	KANN	Avoids phrase splitting and offers recommendations that are easy to understand through the use of knowledge graph representation.	Performance is restricted when knowledge graphs are thin or lacking.
(Danyal et al., 2024a)	XLNet	High sentiment analysis accuracy using sophisticated language models yields insightful information.	Using complex models and extensive preprocessing can be computationally costly.

Problem statement

Sentiment evaluation is difficult because people have different emotional experiences in their movie reviews since they have different emotional states and backgrounds. Sentiment analysis methods are not efficient when it comes to scalability and understanding language context. When sentiment analysis fails, it decreases the usefulness of the data in terms of decision making for businesses. In this study, an attempt will be made to design a deep learning approach that identifies several emotional states in movie reviews more effectively than existing methods.

3. METHODOLOGY

The proposed system to classifying movie reviews follows multiple steps that combine deep learning methods with Natural Language Processing (NLP) technologies. NLP technology processes information from movie reviews in the Rotten Tomatoes, IMDb, and MovieLens datasets. The DCswinT transformer model starts feature extraction on normalized data. This model is able to understand the two dimensions of textual information by applying both the SAM module and the FAM module within its structure. The CSWin Transformer also enhances features by employing both vertical and horizontal attention mechanisms when processing input data. In the classification process, the Polymorphic Graph Attention Bayesian Asymmetric Quantized Neural Networks (PGABAQNNNet) is used for sentiment classification of movie reviews. The Emperor Penguin Optimization (EPO) technique helps the PGABAQNNNet model achieve optimal results through loss function tuning. The design proposed here is presented in Figure 1.

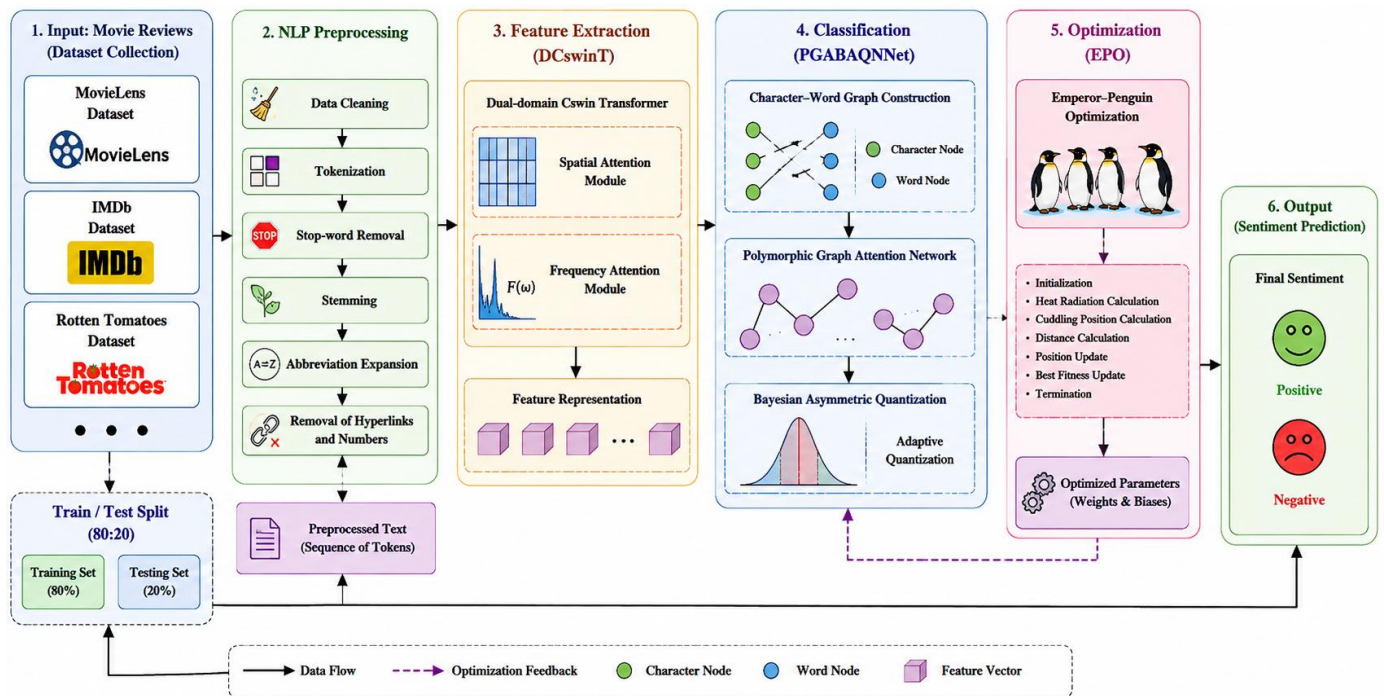


Figure 1. Overall architecture of the proposed PGABAQNNet-EPO framework

Data collection

This research utilizes reviews from the Rotten Tomatoes (RT), IMDb, and MovieLens datasets. These reviews, which include a variety of texts, were gathered from the Kaggle website. The data is cleaned and filtered using preprocessing procedures to guarantee higher quality for analysis. The details of these preprocessing techniques are covered in the next section.

NLP-based preprocessing

After collecting all available movie reviews, the next task is to preprocess this data. Natural Language Processing (NLP) is used at this stage to ensure that well-prepared data is available for analysis. Generally, processed data aids in eliminating any unnecessary information or features. There are several crucial steps involved in gathering preparing the for-film reviews:

Data cleaning: In the process of preparing data, the first stage entails the removal of all unwanted text components from the review texts. Data cleaning ensures that there are no errors to ensure that the texts are error-free.

Tokenization: The process of tokenization is performed after cleaning of the data. In this technique, the reviews are segmented into individual words and phrases. The segmentation of the words into tokens makes the analysis of the data easy.

Stop word removal: After that, the stop words are removed. Words that don't add anything to the evaluation's meaning are removed. This ensures that the analysis is done with important words, which help to understand the core of the review.

Stemming: Following this, stemming is applied to convert words into their base/root form. In other words, "enjoyed," "enjoying" and "enjoys" will be all abbreviated to "enjoy." Normalization of text along with feature space reduction makes the analysis more effective.

Abbreviation expansion: The process continues by expanding the informal evaluation words from abbreviations like "can't" or "won't" to their full form, such as "cannot" or "will not." This is important for performing effective sentiment analysis.

Removal of hyperlinks and numbers: Hyperlinks and numbers should be removed from the dataset because these features appear in customer feedback without contributing the emotions of the feedback. These features help to ensure that only the content of the reviews is highlighted. The above-mentioned preprocessing procedure allows for transformation of the input movie reviews dataset into the clean and structured form, increasing the efficiency of the feature analysis process.

Dataset Harmonization: Though these three datasets are in different formats and contained varied information, they are converted to one common format that included the text of the review and its sentiment. Data relevant to the dataset but unnecessary for sentiment analysis was ignored. All three datasets are subjected to the identical natural language processing preprocessing pipeline, which includes data cleaning, tokenization, stop-word removal, stemming, abbreviation expansion, and hyperlink and number removal, as previously described. This ensures that the representations of the text and the sentiment labels are consistent before feature extraction and classification.

Dual-domain Cswin transformer (DCswinT) based feature extraction

The Dual-Domain CSWin Transformer (Lu et al., 2023) combines SAM, which stands for the Spatial Attention Module and FAM, which is Frequency Attention Module, as two important components in the feature extraction process. The dual domain technique improves feature extraction process through the utilization of both domains. Whereas the spatial domain of texts can be considered as the structure or organization of words (sentence or paragraph structure), the frequency domain represents the distribution of various properties of words such as TF-IDF, which quantify the significance of terms inside the review dataset. The SAM considers every word of the phrase as if each is in a grid, with its emphasis being on the spatial aspects of the data. SAM captures word dependencies based on spatial correlations by using attention mechanisms akin to the Multi-Head Self-Attention (MHSA). The initial MHSA formulation is presented in equation (1).

$$MHSA(R, L, W) = \text{concat}(\text{head}_1, \dots, \text{head}_i) X^p \quad (1).$$

Here, R is the current word (query), L is the surrounding words (keys), and W is the context or relationships between those words (values). The scaled dot-product attention is computed using equation (2).

$$\text{attention}(R, L, W) = \text{soft max} \left(\frac{RL^U}{\sqrt{e_i}} \right) W \quad (2).$$

The weight assigned to the surrounding words with respect to the current word will enables the algorithm to concentrate on the important terms in the review sentiment analysis. Frequency Attention Module (FAM) is used to extract multiple frequencies from the text. The frequency domain is segmented into low and high frequency in FAM. It takes place through two steps:

Global Frequency Modulation (GFM): The extraction is done for both global low-frequency (words used frequently in the reviews) and high-frequency (less frequent words that may have sentiments attached to them) components, as shown by equation (3).

$$Y_h^m = GAP(Y); Y_h^i = Y - Y_h^m \quad (3).$$

Y_h^m and Y_h^i represent the global low-frequency and high-frequency components, respectively, that were extracted from the entire review text Y .

Local Frequency Modulation (LFM) concerns itself with obtaining local information about smaller text fragments (e.g., sentences or phrases) and adjusting the frequency components locally, as seen in equation (4).

$$Y_m^m = AP(Y); Y_m^i = Y - Y_m^m \quad (4).$$

In this case, Y_m^m and Y_m^i are the local low- and high-frequency components obtained by applying average pooling (AP) to local review segments.

The FAM module applies attention weights to regulate the extracted low- and high-frequency components, as shown in equation (5).

$$FAM(Y) = X_h^m Y_h^m + X_h^i Y_h^i + X_m^m Y_m^m + X_m^i Y_m^i \quad (5).$$

The model's performance in sentiment classification will be improved by using the high and low-frequency keywords from the movie review. The CSWin Transformer plays a very important role during the feature extraction stage. This hierarchical transformer increases the feature complexity while progressively decreasing the spatial dimensions of the input data. The CSWin Transformer is employed to process the data. The multi-head self-attention processing of the input involves arranging tokens (words) into horizontal and vertical stripes. From left to right in the review, the horizontal stripe handles word groups. The vertical stripe handles word clusters from top to bottom. To calculate the horizontal and vertical attention, use equation (6).

$$H - attention_l(Y) = attention(Y^j X_l^R, Y^j X_k^L, Y^j X_l^W) \quad (6).$$

Where, Y^j represents each word in the horizontal stripe and X_l^R , X_k^L , and X_l^W are the learned weight matrices for queries, keys, and values. Lastly, a linear transformation is applied to the concatenated outputs of the both horizontal and vertical attention, as indicated by equation (7).

$$CSWin - attention(Y) = concat(head_1, head_2 \dots head_L) X^P \quad (7).$$

Here, attention is calculated individually by each head, and X^P , a learnable weight matrix, linearly transforms the concatenated output to capture diverse information from various subspaces. The CSWin Transformer block's processing can finally be effectively represented by equation (8).

$$Y^m = MLP(LN(\hat{Y}^m)) + \hat{Y}^m \quad (8).$$

Where, the output of the m^{th} CSWin Transformer block is denoted by Y^m .

Review classification using Polymorphic graph attention Bayesian asymmetric quantized neural networks (PGABAQNNet)

Review classification is performed using Polymorphic Graph Attention Bayesian Asymmetric Quantized Neural Networks (PGABAQNNet). The Polymorphic Graph Attention Transform (Wang et al., 2022) focuses on matching words and characters to create a polymorphic graph that represents character-word interactions in a review using a graph attention network. In the given character sequence $D = [d_1, d_2 \dots d_m]$ and the matching words $X = \{x_1, x_2, \dots x_n\}$, directed edges between the word vertices and character vertices are found. To define edge types, use equation (9).

$$edge(j, k) = \begin{cases} 1, & x_k \in C(d_j) \\ 2, & x_k \in N(d_j) \\ 3, & x_k \in F(d_j) \\ 4, & x_k \in T(d_j) \end{cases} \quad (9).$$

Where, $C(d_j)$, $N(d_j)$, $F(d_j)$ and $T(d_j)$ stand for the word x_k boundaries in the character sequence. Every letter and word is contained in a dense vector space. The embeddings for the words i_{d_j} and the letters i_{x_k} are determined by equation (10).

$$\begin{aligned} i_{d_j} &= f^x(d_j), d_j \in D \\ i_{x_k} &= f^x(x_k), x_k \in X \end{aligned} \quad (10).$$

In this instance, the embedding retrieval function is represented by f^x . Softmax attention is used to compute the attention between letters and matching words by the formula (11).

$$att(R, L, H) = \text{soft max} \left(\frac{RL^U \bullet H}{\sqrt{e_n}} \right) \quad (11).$$

The query and key matrices for character and word embeddings are R and L , respectively, and graph's adjacency matrix is H . For each of the four subgraphs that constitute the graph, and correspond to the edge types ($C, N, F, \text{and } T$), attention is calculated independently, as shown by equation (12).

$$\alpha_\gamma = att(i_d X_R, i_x X_L, H_\gamma) \quad (12).$$

This attention mechanism α_γ is used on the word embeddings to extract information from the embeddings for each edge type. Another improvement done on the attention model is the use of multi-head attention. The usage of a Bayesian Asymmetric Quantized Neural Network (BAQNN) is suggested in this research (Chien and Chang, 2023), which employs weight quantization and weight uncertainty in the neural network, to classify the sentiments. The objective is to classify the input sequence through minimization of the negative log-likelihood between the actual class and the probability prediction. The loss function is described by Equation (13).

$$L_{nll}(\hat{z}, z) = -\log q(z|y) = -\sum_{m=1}^M \sum_{l=1}^L z_{ml} \log \hat{z}_{ml} \quad (13).$$

Where, z_{ml} is the actual class label, $\hat{z}_{ml}$ is the projected probability for class ml , and L is the total number of classes. A Bayesian framework is applied to control the degree of uncertainty in the model parameters. The probability of the data z given the input y and weights x is given by equation (14)

$$q(E) = q(z|y) = \int q_\theta(z|y, x) q(x) dx \quad (14).$$

Where, z is the target label for that data point and q is the x^{th} input data point. It is further assumed that the weights are Gaussian priors as described by equation (15).

$$q(x_{jk}) \propto \frac{1}{|x_{jk}|} \quad (15).$$

The variational distribution is employed in approximating the q posterior distribution. The variational parameter α is denoted as and is modified during the process of training. This can be described through Equation (16) below, which improves the Evidence Lower Bound (ELBO).

$$L(\theta, \phi) = F_{r_\phi(x)} [\log q_\theta(z|y, x)] - LM(r_\phi(x) \| q(x)) \quad (16).$$

Where θ and ϕ are model parameters, L represents the objective function. With input y and latent variables x , the prior distribution of x is $q(x)$, and the likelihood of the target z is $(z|y, x)$. A finite number of levels are created by quantizing the

weights using an M-ary asymmetric quantization function $R(x)$. To find the final class prediction $\hat{z}$, an input y is sent through a network with quantized weights and bias c , as shown in equation (17).

$$\hat{z} = b^{(L)} \quad (17).$$

The activation function applied to the model's output is denoted as $b^{(L)}$, where L is the final layer output, and z is the classification output.

Emperor penguin optimizer

Emperor Penguin Optimization (EPO) (Dhiman et al., 2021) uses the social behavior of emperor penguins to develop an optimization method. This method mimics the behavior of penguins as they form groups and moving around in order to optimize certain objectives. A number of steps are involved in finding the optimal contact points and then adapting them based on the movements of penguins and environmental temperature. The process can be explained as follows to modify this approach for optimizing a loss function:

Step 1: Initialization

To start the optimization process, researchers need to set the raft size, which determines the no of penguins. Every penguin serves as a possible solution for the optimization search space. A raft size determines how well the algorithm can search its space.

Step 2: Calculation of Penguin's Heat Radiation

In EPO, the penguin's temperature V is calculated using equation (18).

$$V' = \left(V - \frac{\max_{iter}}{v - \max_{iter}} \right) \quad (18).$$

Where, V' is the penguin's temperature at iteration, $\max_{iter}$ is the total number of iterations, and v is the number of iterations that are currently occurring. Additionally, a random number T is used to adjust the heat radiation, as described in equation (19).

$$V = \begin{cases} \text{if } T > 0.51 & 0 \\ \text{if } T < 0.5 & 1 \end{cases} \quad (19).$$

The random number $T \in [0,1]$ that is used to make this decision aids in determining how much the temperature affects the penguins' movement.

Step 3: Cuddling Position Calculation

The movement vectors C and E are used by penguins to avoid collisions and to place themselves in the best possible position for cuddling. To calculate these vectors, use equation (20).

$$\begin{aligned} \bar{C} &= \left(O \times \left(V' + Cbc \left(\bar{R} - \bar{R}_{ep} \times (accuracy) \right) \times rand() \right) \right) - V' \\ \bar{E} &= rand() \end{aligned} \quad (20).$$

where $\bar{R}$ is the penguin's current position, $\bar{R}_{ep}$ is the best solution or fitness discovered thus far, and V is the temperature. $rand$ ensures that the solution space is explored by producing a random number.

Step 4: Calculating Distance among Penguins

The distance between the penguins within the raft then influences their movement relative to one another. The distance between two penguins is calculated using the control parameters h , w and an exponential function g , as described in equation (21).

$$\begin{aligned} \vec{F}_{ep} &= Cbc \left(U \left(\vec{C} \bullet \vec{Q}(z) - \vec{E} \bullet \vec{R}_{ep}(z) \right) \right) \\ U(\vec{C}) &= \left(\sqrt{h \cdot g^{-w/x} - g^{-w}} \right)^2 \end{aligned} \quad (21).$$

where, g is the exponential function, $R(z)$ is the penguin's current position, h and x are control factors that affect the strength of social forces, F is the distance between penguins, and $U(C)$ is the social forces acting between penguins. The penguins will be steered towards better solutions because of the social aspects, which ensure that they do not clash and position themselves according to the best solutions.

Step 5: Position Update

Following this computation of distances and forces, the following formula (22) is used for updating the position of the penguins.

$$\vec{R}_{exp}(z+1) = \vec{R}(z) - \vec{C} \bullet \vec{F}_{ep} \quad (22).$$

Where, C represents the movement vector from Step 3, D represents the distance from Step 4, and $R_{exp}(z+1)$ represents the penguin's updated position. This makes it sure that the penguins are moving in a direction which is bringing them closer to the best possible solution.

Step 6: Storing and Comparing Best Fitness

In every iteration, the optimal solution obtained from the earlier iteration is compared with the penguin having the maximum fitness in that iteration. Global best solution is updated so that the algorithm approaches the optimal solution. Fitness function is analyzed to get the best possible response. The process of tracking the global best fitness $U(C)$ is described in equation (23).

$$U(C) = \text{Global optima}(\text{best solution}) \quad (23).$$

Step 7: Termination

The best solution stops improving significantly, or the loss function attains the desired optimal value after a certain number of maximum iterations. When these conditions are met, the algorithm halts and outputs the best answer available at that moment (R_{best}). The provision of an ideal or almost ideal response ensures that the result will be effective. Figure 2 shows the EPO flowchart.

Rationale for EPO Selection: EPO algorithm was chosen because it has an optimal search strategy that combines during the optimization process, including exploration and exploitation. Temperature control, huddling, and socialization provide the ability to conduct efficient searching of the solution space and guide the candidate solutions to better regions. While Adam uses gradient-based optimization, EPO uses population-based optimization, making it more appropriate for tuning the nonlinear objective function of the proposed PGABAQNNet model. When compared to PSO, GWO, and WOA, EPO offers an innovative population-based search technique that is characterized by the adaptive temperature control and interaction among social individuals. Thus, EPO was used for the effective optimization of the parameters of the proposed model.

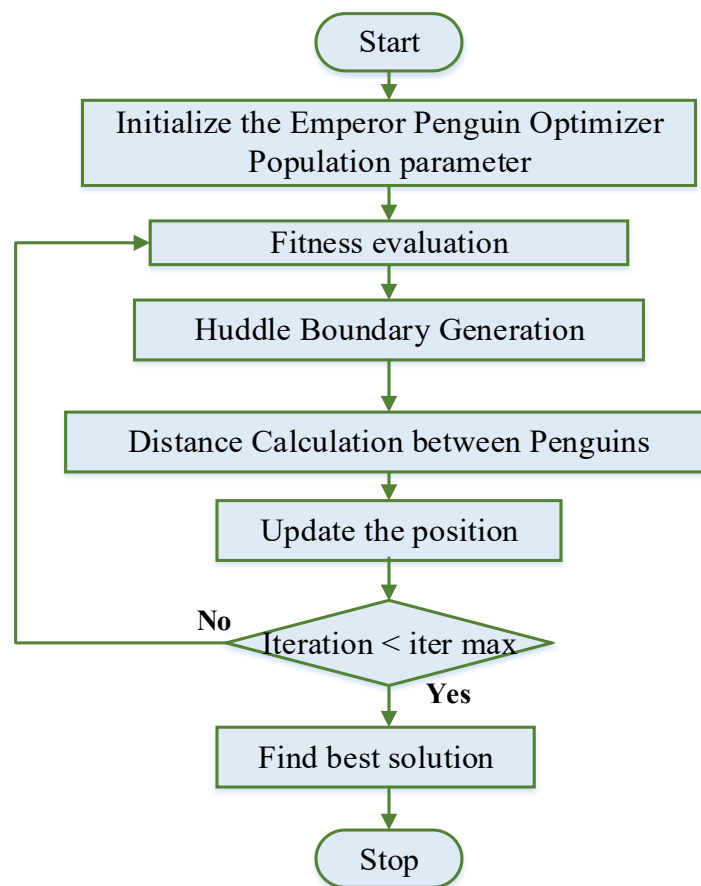


Figure 2. Flowchart of EPO

4. RESULTS AND DISCUSSION

The analysis and outcomes of the suggested PGABAQNNNet-EPO method will be covered in this part. The Python version 3.7.14 was employed in the implementation of the algorithm, while Windows 10 was used for testing. Table 2 displays the implementation parameters used in the analysis.

Table 2. implementation parameter

Parameter	Description
Programming language	Python
Version	3.7.14
Operating system	Windows 10
Dataset	IMDb, RT, MovieLens
Data type	Text
Hardware	Intel Core i7 CPU, 16 GB RAM
GPU	NVIDIA RTX 3060, 12 GB
Deep learning framework	PyTorch 1.10
Optimization	EPO
Number of epochs	100
Batch size	32

Learning rate	0.001
Optimizer	Adam
Weight decay	0.0001
Dropout rate	0.3
EPO population size	20
EPO maximum iterations	50
Train/test split	80% / 20%
Random seed	42

Table 2 summarizes the hardware, software, training, optimization, and hyperparameter settings used to implement the proposed PGABAQNNNet-EPO model and ensure reproducibility of the experiments.

Dataset description

Since sentiment analysis and movie suggestion are the goals of the proposed study, this dataset will be employed. The class label within this particular data set determines if the sentiments contained are either positive or negative.

IMDb dataset: The English movie reviews data in IMDb are comprised of 100,000 reviews. The plain text file does not have any information regarding rating. 20% of the dataset is used for testing, while the remaining 80% is used for training.

Rotten Tomatoes (RT) Dataset: A total of 1,130,017 film reviews in the English language from the Rotten Tomatoes website have been accessed through Kaggle. In order to save processing time, it was restricted to 45,746 reviews since 2020 onwards. The total number of 40,737 reviews was attained through classification of the data using two sentiment categories: "Fresh," which consisted of 28,986 reviews, and "Rotten," which had 11,751 reviews.

MovieLens Dataset: The number of users in the MovieLens dataset is 1M, who rated 4,000 movies, resulting in 1M records. This dataset had text data enhancement in the form of extraction of keywords from the IMDb description of movies for better rating prediction accuracy. There are 424,928 occurrences for negatives and 575,281 occurrences for positives; hence, the data is relatively balanced. Table 3 displays the dataset description.

Table 3. Description of the dataset

Dataset	Class	Total	Training (80%)	Testing (20%)
MovieLens	Positive	575,281	460,225	115,056
	Negative	424,928	339,942	84,986
	Total	1,000,209	800,167	200,042
IMDb	Positive	25,000	20,000	5,000
	Negative	25,000	20,000	5,000
	Total	50,000	40,000	10,000
Rotten Tomatoes	Positive	28,986	23,189	5,797
	Negative	11,751	9,401	2,350
	Total	40,737	32,590	8,147

Performance metrics

Several crucial metrics, including accuracy, precision, recall, F1-score, specificity, and error rate, are used to assess PGABAQNNNet-EPO's performance. Other popular techniques for sentiment analysis of film reviews are contrasted with these metrics. Formulas for calculating each of the metrics are provided below in Table 4:

True Positive (A): Correctly detects the presence of positive sentiment.

True Negative (B): Correctly predicts the negative sentiment.

False Positive (C): Incorrectly identifies a negative sentiment as positive.

False Negative (D): Incorrectly identifies a positive sentiment as negative.

Table 4. Performance metrics

Metrics	Equation
Accuracy	$\frac{A + B}{A + C + B + D}$
Precision	$\frac{A}{A + C}$
Recall	$\frac{A}{A + D}$
F1-score	$\frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$
Specificity	$\frac{B}{B + C}$

Performance analysis of proposed model

This part proves the success of the proposed PGABAQNNNet-EPO model by analyzing sentiment in movie reviews across different datasets from IMDb, RT, and MovieLens.

Table 5. Testing outcomes

Original review	Preprocessing	Feature extraction	Review classification	Rating
A dream journey that combines Greek mythology with modern American locations and ideals. Anybody are...	fantasy adventure Greek mythology contemporary American places values	Fantasy, Adventure, Mythology.	Positive	5
There are no more blatant and careless Harry Potter knockoffs than this would-be franchise star...	Harry Potter knockoffs transparent slapdash wannabe franchise jumpstart.	Harry Potter, Knockoffs, Transparent.	Negative	1
Halfway thru, the protago says one of the most important lines in the whole movie...	significant quotes halfway film protagonist	Significant, Quotes, Film	Positive	4.5
This movie came in a box of five pretend movies, so I didn't have high expectations when I saw it...	Watched film expecting pack of films	Film	Negative	
I didn't have great hopes when I viewed this movie because it came in a box with five phony movies...	crammed dragons set-destroying fights things exploding Columbus squeezes	Fights, Exploding	Positive	4
It's an indifferent film, but it's not awful for what it is or for whom it is meant...	bad movie indifferent	Indifferent, Intended	Negative	2

Table 5 shows how the system handles movie review preparation, feature extraction, and review categorization. The Preprocessing column contains simplified versions of original reviews with the main points highlighted. Feature extraction highlights keywords or

themes, such as genres or sentiments, extracted from the text. The rating column shows the reviewer's score in equivalence, and the review classification shows if the review is positive or negative.

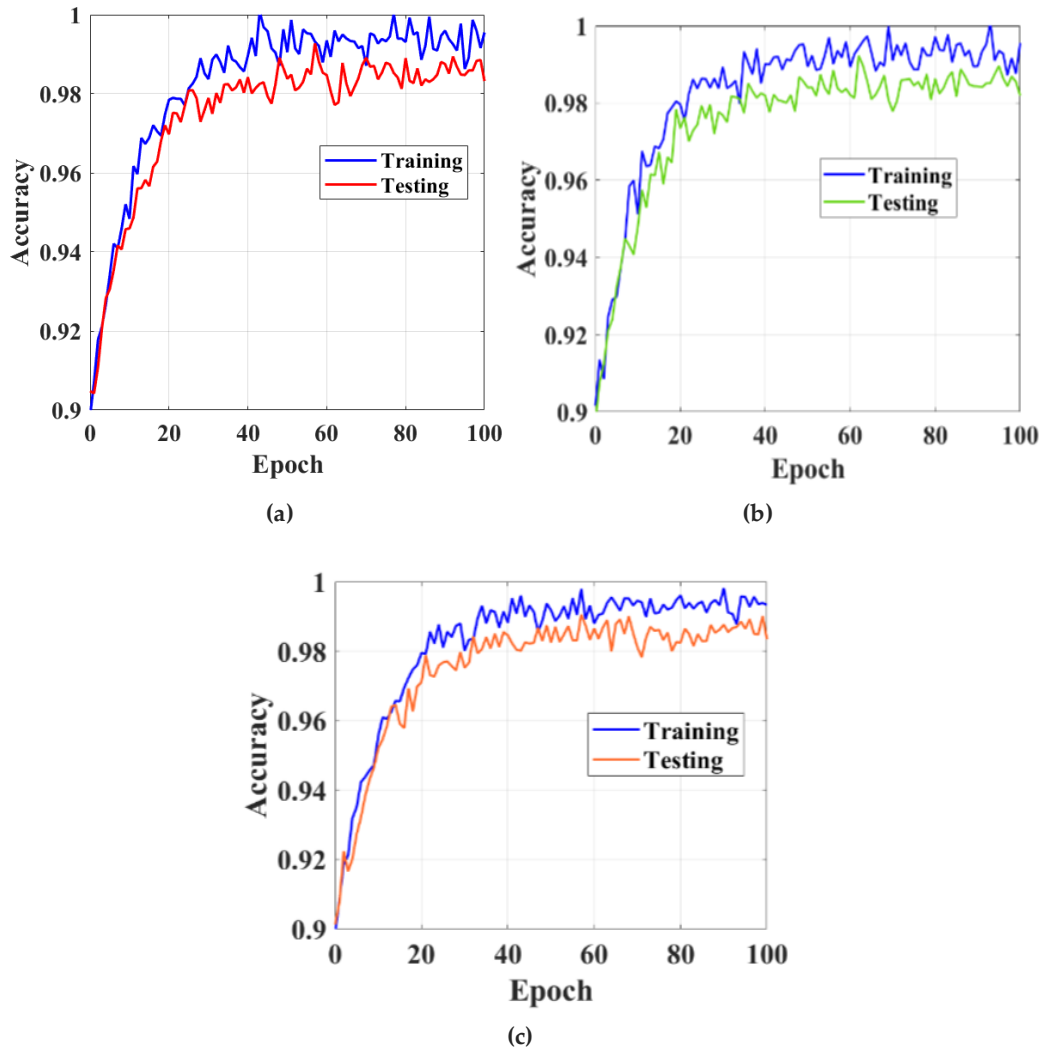


Figure 3. Proposed loss graph on (a) RT, (b) IMDB, (c) MovieLens datasets

Figure 3 presents the training (blue) and testing (red/green/orange) accuracy plot of the PGABAQNNNet-EPO model over the 100 epochs for RT (a), IMDB (b), and MovieLens (c) datasets. The accuracy reaches around 0.98 in the case of the training dataset, and 0.96 for the testing. Accuracy while training for IMDB is about 0.99 while testing accuracy is close to 0.98. Accuracy while training for MovieLens is about 0.98 while testing accuracy converges to about 0.97. In all datasets, the graphs clearly show convergence.

The loss performance of PGABAQNNNet-EPO model for the three datasets is illustrated in figure 4. The loss function values of RT (a) decrease from 0.68 to 0.57. The loss value of IMDB (b) is 0.56. The testing loss of MovieLens (c) becomes around 0.57. These results indicate that loss minimization is quite accurate and effective in all data sets.

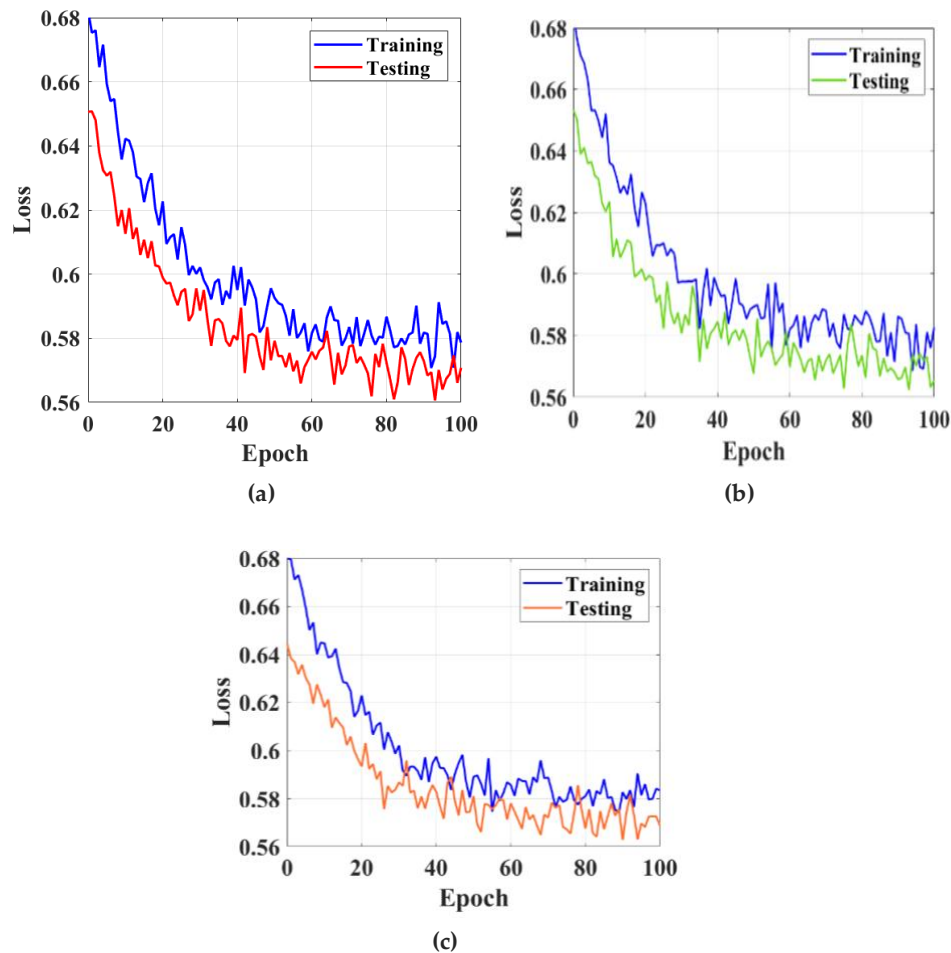
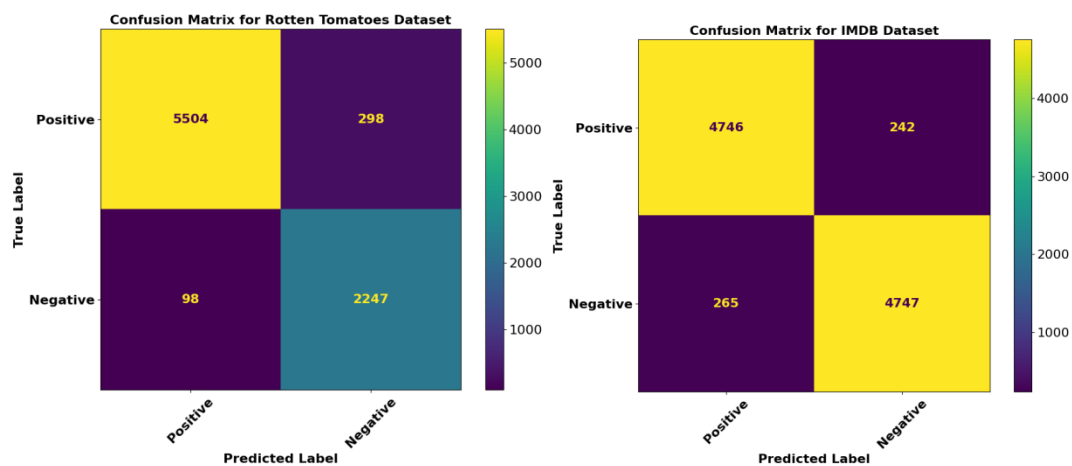


Figure 4. Proposed loss graph on (a) RT, (b) IMDB (c) MovieLens datasets

Confusion matrices for the proposed PGABAQNNNet-EPO model across the three datasets are shown in the Figure 5. In Figure (a), the classifier correctly classifies 5504 positives and 2247 negatives, and incorrectly classifies 298 positives and 98 negatives. In Figure (b), the classifier correctly classifies 4746 positives and 4747 negatives, and incorrectly classifies 242 positives and 265 negatives. Moreover, the model accurately classify 109,251 positive and 80,862 negative examples in Figure (c), as well as misclassifies 5805 positive and 4124 negative examples. The above results illustrate the effectiveness of the model in classification tasks.



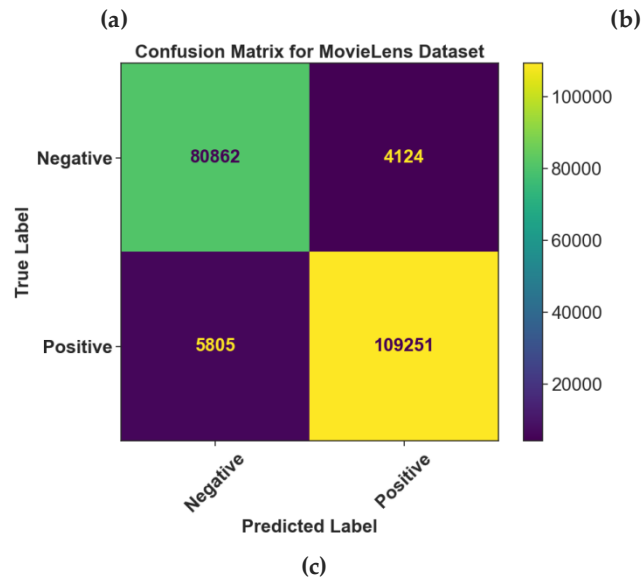


Figure 5. Proposed confusion matrix on (a) RT (b) IMDB (c) MovieLens datasets

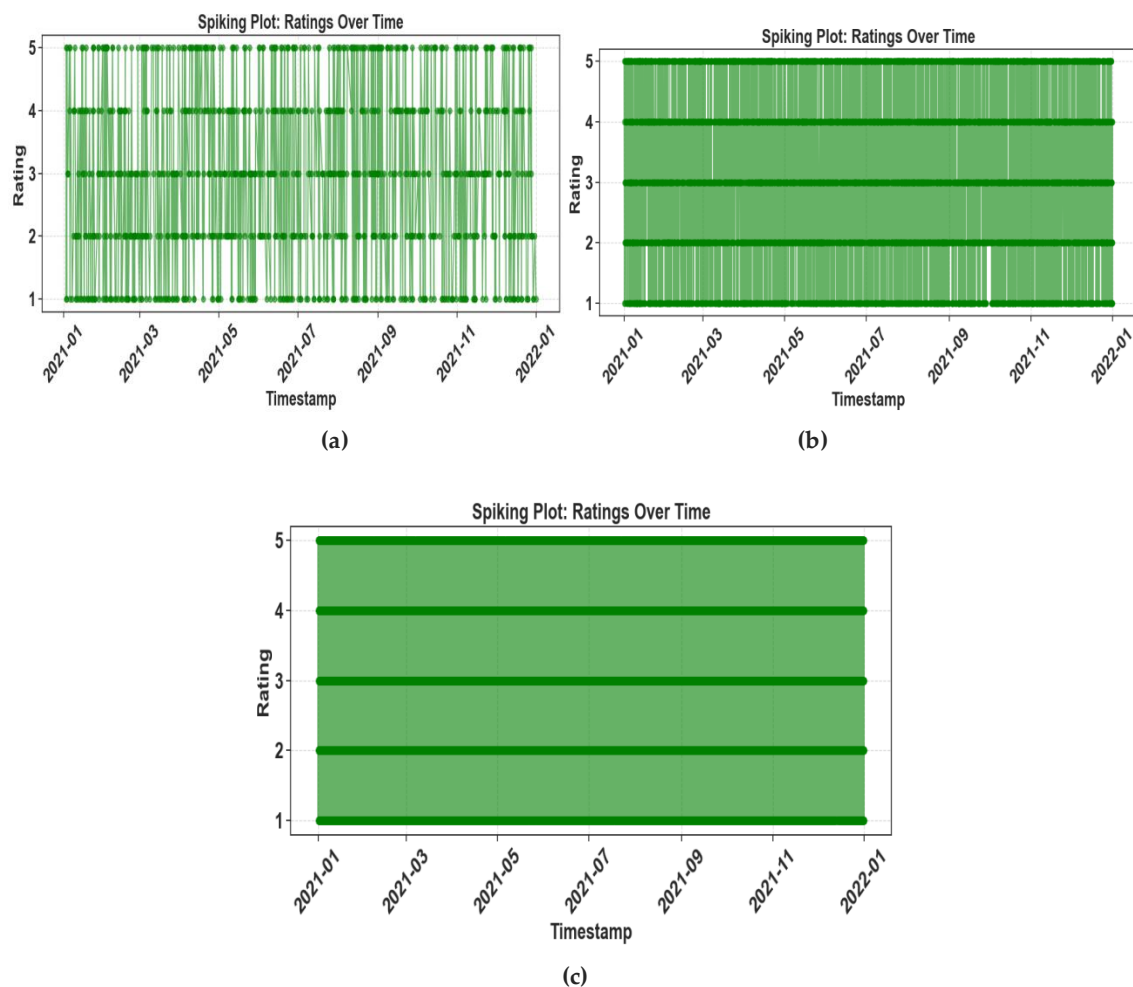


Figure 6. Proposed spiking plots of ratings over time on (a) RT, (b) IMDB (c) MovieLens datasets

Figure 6 displays user rating spike charts across time for three datasets: MovieLens, IMDb, and Rotten Tomatoes (RT). Ratings (1–5) on the y-axis are plotted against timestamps on the x-axis for the entire year 2021 in each subplot. Ratings are sparse in subplot (a), denser in subplot (b), and more uniformly distributed in subplot (c). These graphs illustrate how the user participation and rating patterns vary among the datasets.

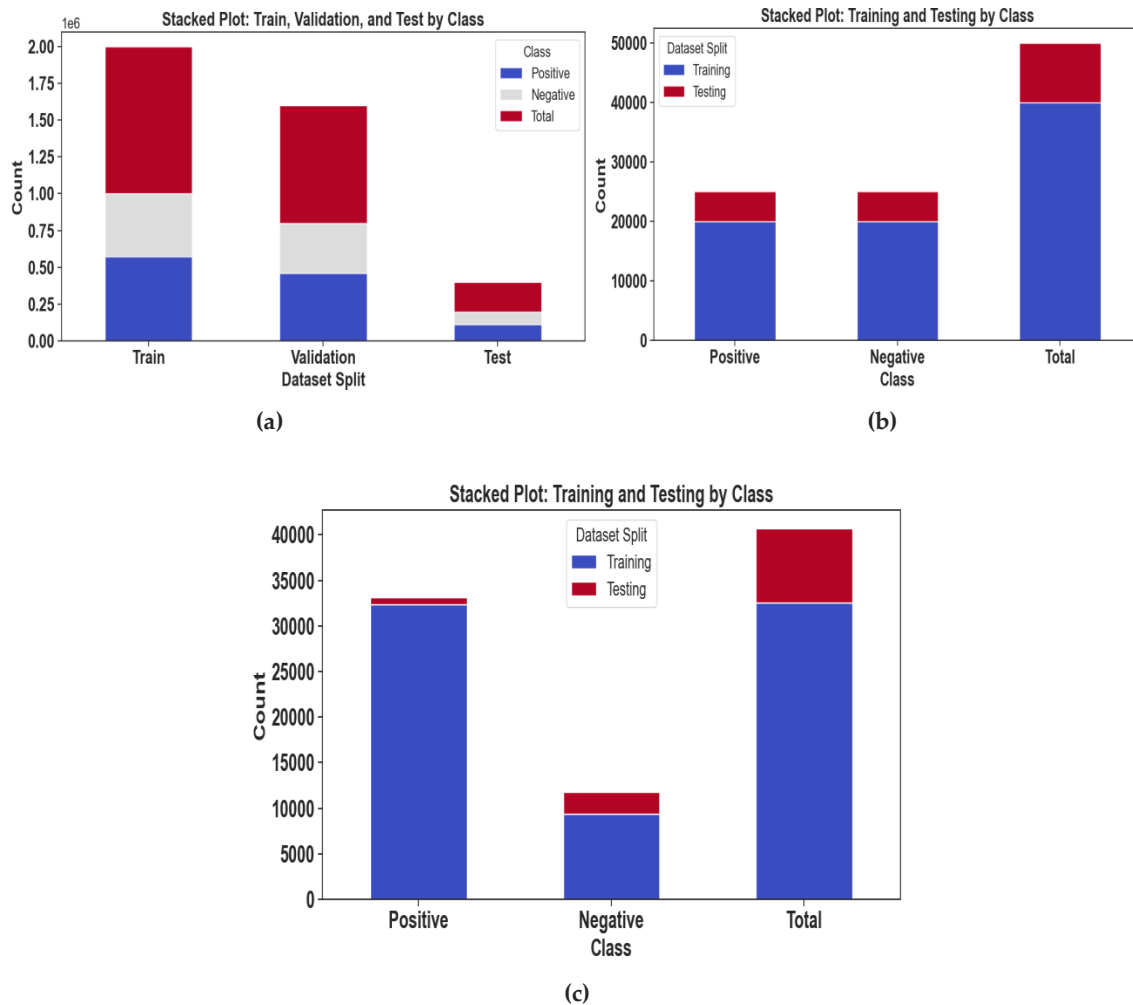


Figure 7: Stacked plot on (a) RT (b) IMDb (c) MovieLens datasets

Figure 7 shows stacked bar graphs representing class distribution (positive, negative) and data split (training, validation, test) for RT (Rotten Tomatoes), IMDb, and MovieLens datasets. Training over a larger set of data is depicted in Subplot (a) that depicts data distribution among the splits. The subplots (b) and (c) are shown for testing and training of positive and negative classes. Positive classes dominate in all datasets, whereas MovieLens exhibits more balanced splits for testing and training. The class distribution and the dataset splits are depicted in these diagrams.

The distributions of the positive, negative and total number of classes of the RT, IMDb, and MovieLens datasets are illustrated in Figure 8. The first subplot (a) emphasizes the larger training set by presenting the distribution across the training, validation, and test splits. Graphs (b) and (c) represent the training and test data distributions, respectively. Positive instances are more in number than negative instances for all the data sets. The graphs present the imbalance in class distribution and the percentage of each data set used for training and testing data, respectively.

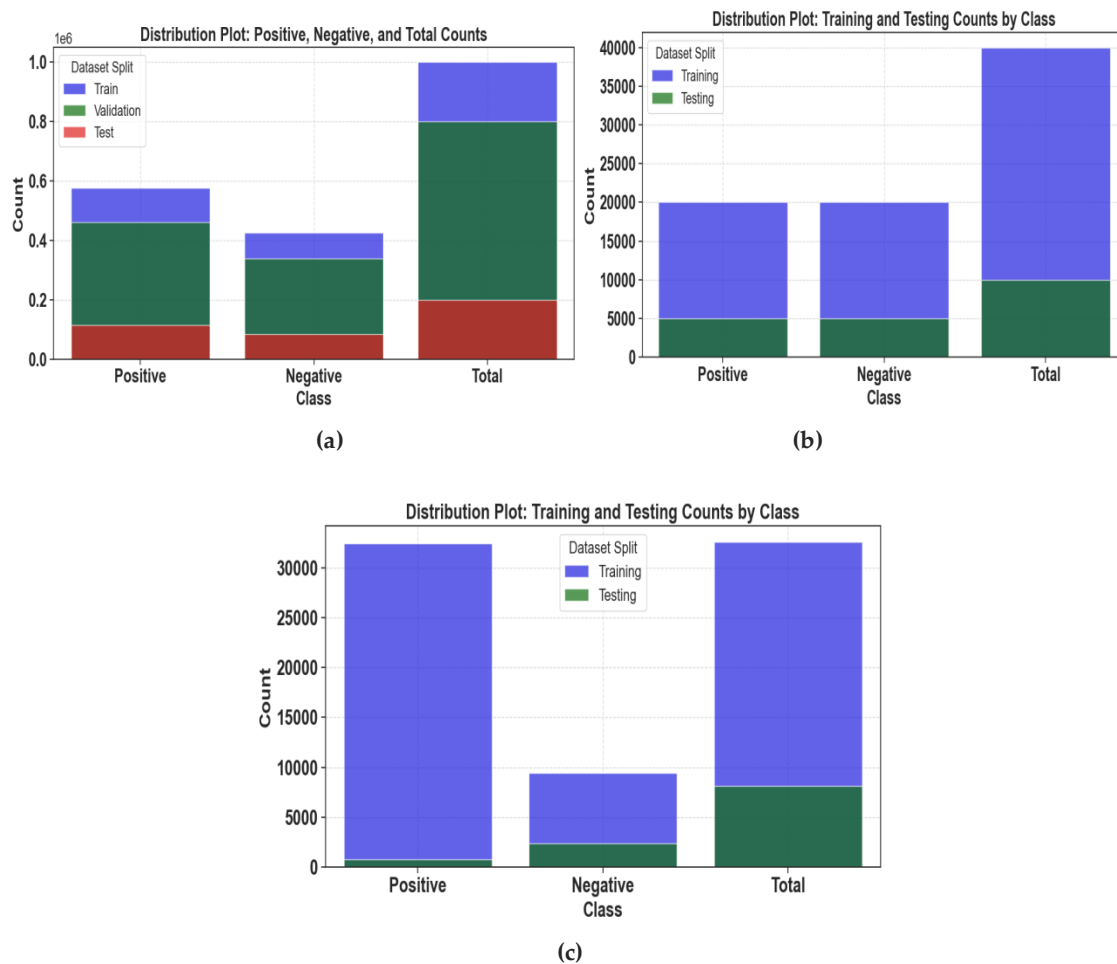


Figure 8. Proposed Distribution plots on (a) RT, (b) IMDb (c) MovieLens datasets

Correlations among the attributes like age groups, gender, and scores can be found in Figure 9 for three sets of data: Rotten Tomatoes (RT), IMDb, and MovieLens. In each graph, there are distributions along with the scatter points indicating the user demographics (age groups) and classes (positive/negative scores). Density graphs highlight the differences in the data distribution across the datasets and also indicate the trends in user activity and ratings. MovieLens indicates consistent patterns in review IDs and ratings, while IMDb highlights class labels, i.e., positive and negative. On the other hand, RT focuses on various age groups. This analysis helps improve our personalized recommendation model.

Figure 10 shows the rating distribution in the IMDb and RT dataset. With minor deviations, the frequency of ratings for RT on the scale from 1.0 to 5.0 is shown in subplot (a). The user rating preferences have been depicted in Subplot (b) above, which indicates that there is an increased number of ratings in the IMDb dataset at the mid-range level, with peaks at regular intervals. The overlaid graphs indicate the pattern of ratings and repeated rating behavior. Recommendation systems require these types of graphs for their functioning.

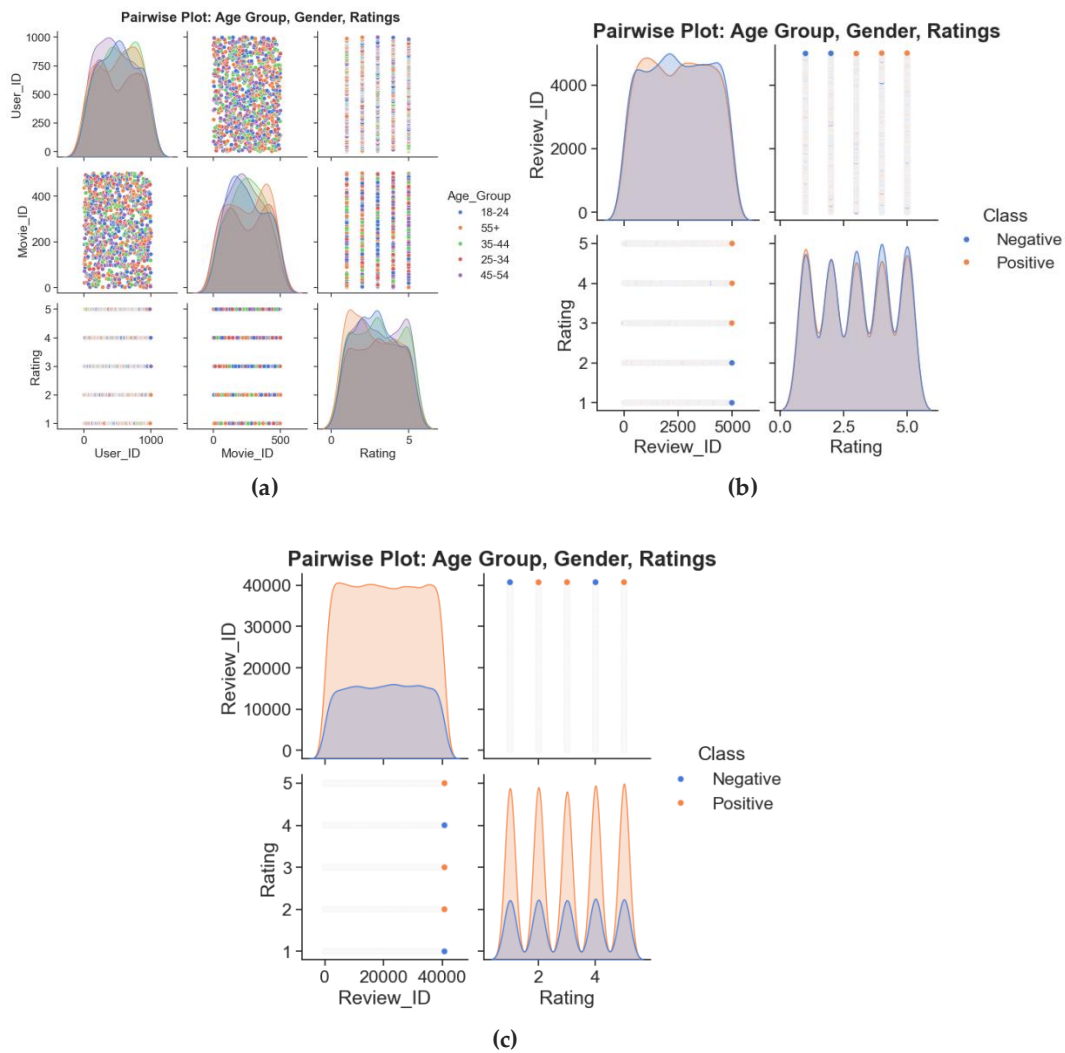


Figure 9. Pairwise Analysis of Age Group, Gender, and Ratings (a) RT (b) IMDb (c) MovieLens datasets

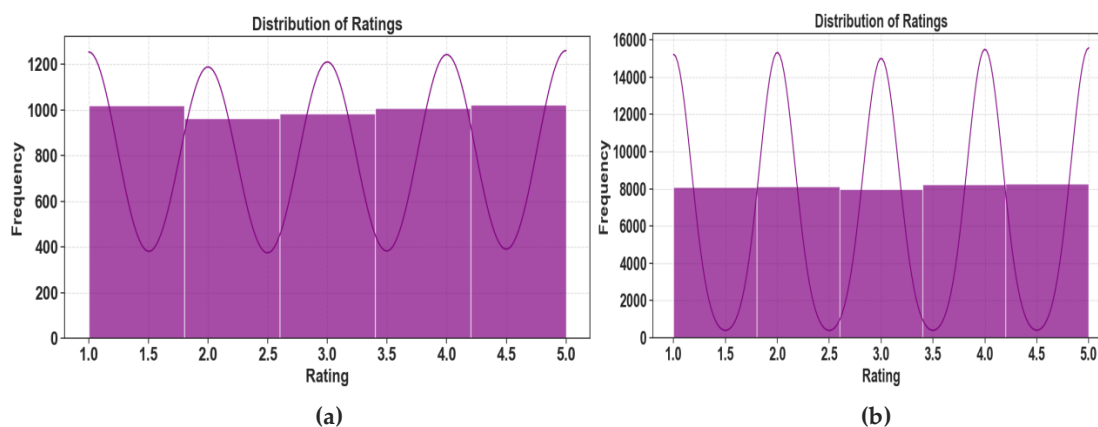


Figure 10. Ratings Distribution for the IMDb and RT Datasets

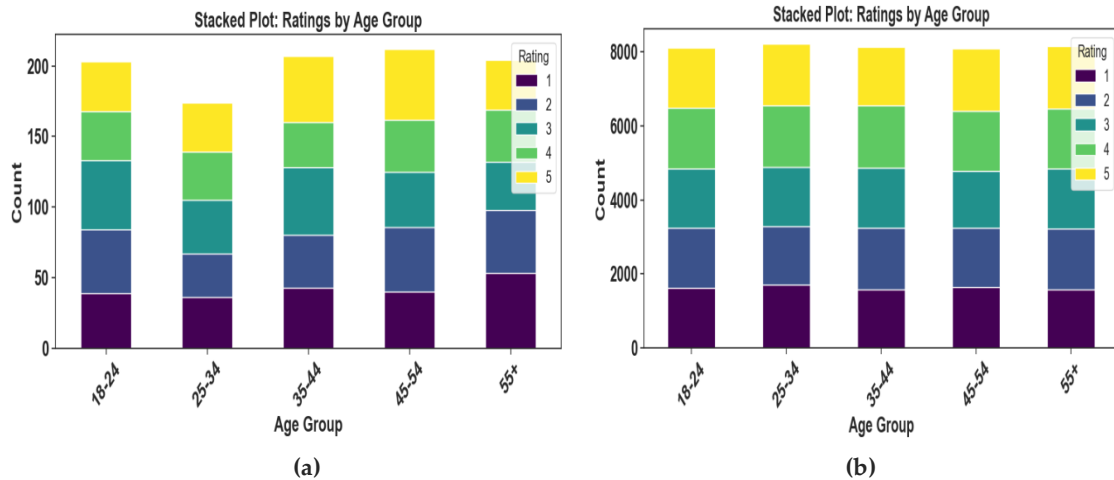


Figure 11. Ratings Distribution by Age Group for (a) RT and (b) IMDb

The distribution of user ratings for the RT and IMDb datasets in terms of age groups is represented in Figure 11. All age groups in RT have a relatively even distribution of ratings, with no preference for any rating in particular, as observed in subplot (a). Although there is an increase in the number of ratings in all categories, it is observed that the pattern in Subplot (b) for IMDb is quite the similar. Using the stacked bar method gives the information regarding the effects of demographic factors on ratings in terms of the proportional share of each rating (1-5) in each demographic segment.

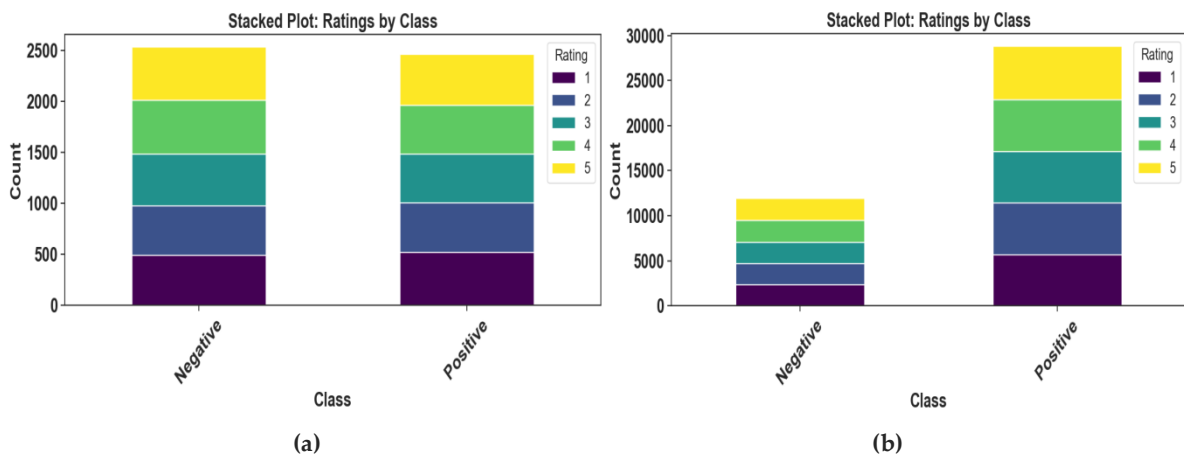


Figure 12. Comparison of Ratings Distribution for (a) RT and (b) IMDb

Stacked bar plots comparing the distribution of ratings (1–5) between the positive and negative classifications are shown in the Figure 12. There is no class imbalance in (a), as both classes have a balanced distribution with approximately equal counts for each rating. As there is significantly more of the positive instances as compared to the negative instances, the positive class is greater than the negative class in (b). These insights inform how to understand dataset properties and ensure appropriate preprocessing or sampling techniques are used before model training.

The distribution of male and female ratings (1–5) is shown visually in the figure 13. No discernible gender bias or imbalance is observed in the dataset, since both genders display a similar distribution pattern with comparable counts across all rating levels. The balanced results indicate gender plays no part in rating distribution, helping us achieve equal models free from bias. Revealing these data insights helps us trust that proposed models work faithfully and accurately.

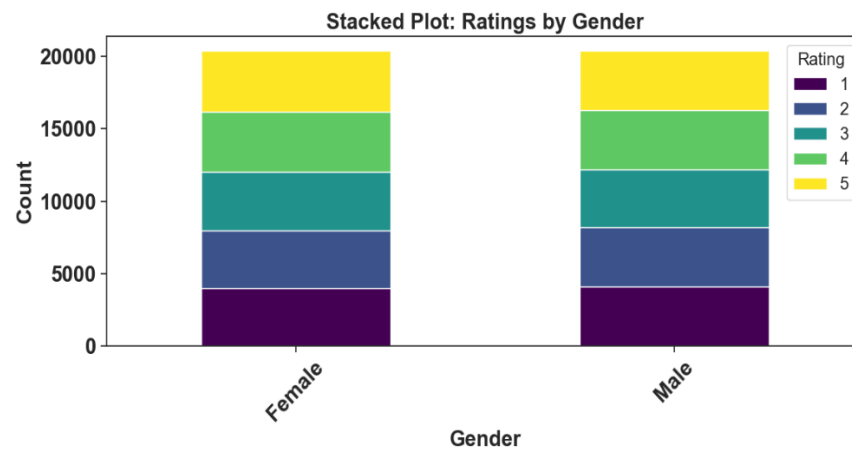


Figure 13. Distribution of Ratings by Gender on MovieLens datasets

Comparative analysis

Three datasets are used to assess the efficacy of the suggested PGABAQNNNet-EPO method: Rotten Tomatoes (RT), IMDB, and MovieLens. In case of MovieLens data set, the methods include GDNN (Bazargani et al., 2024), RAKCR (Cui et al., 2024), Twit-CoFiD (Latrech et al., 2024), DCSNN (Shrivastava and Kumar, 2024), FL (Nagy et al., 2024), and the Dynamic Graph Structural Framework (DGNN) (Zhao et al., 2024). The IMDB dataset is benchmarked against the following methods: FL (Nagy et al., 2024), HUAPA-DCF (Ibrahim et al., 2023), KANN (Liu & Miyazaki, 2023), XLNet (Danyal et al., 2024a), TF-IDF (Danyal et al., 2024b), and DeBERTa (Cao, 2025). For the RT data set, it is being compared against the following approaches: XLNet (Danyal et al., 2024a), TF-IDF (Danyal et al., 2024b), TF-IDF+ASR (Chao et al., 2023), WAC (Chien et al., 2024), DGNN (Zhao et al., 2024), and DeBERTa (Cao, 2025). These new GNNs and Transformer methods being incorporated have provided us with a much more thorough and current way of evaluating the proposed framework. PGABAQNNNet-EPO outperforms the state-of-the-art sentiment classification models in all datasets examined in terms of accuracy, precision, recall, F1 score, specificity, and error rate.

Table 6. Evaluation of the Proposed Algorithm's Performance Compared to Current Approaches on MovieLens, IMDB, and Rotten Tomatoes Datasets

Dataset	Methods	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
MovieLens	GDNN	91.5	92.1	90.3	91.2
	DCSNN	94.6	95.2	93.9	94.5
	DGNN	95.2	95.6	94.8	95.1
	DeBERTa	96.1	96.4	95.7	96.0
	Proposed PGABAQNNNet-EPO	99.1	99.3	99.2	99.5
IMDb	KANN	94.9	96.2	94.5	95.3
	XLNet	96.2	97.0	95.7	96.3
	RoBERTa-GRU	94.6	94.8	94.5	94.6
	BERT	98.0	98.1	97.9	98.0
	DeBERTa	98.2	98.4	98.0	98.2
	DGNN	97.1	97.3	96.8	97.0
	Proposed PGABAQNNNet-EPO	99.4	99.5	99.3	99.6
RT	XLNet	91.0	91.2	89.8	90.5
	WAC	94.3	95.1	93.2	94.1
	DGNN	95.4	95.7	95.0	95.3
	DeBERTa	97.2	97.4	97.0	97.2
	Proposed PGABAQNNNet-EPO	99.3	99.7	99.5	99.7

Table 6 indicates that the suggested PGABAQNNNet-EPO model has proven to be the best performer on the MovieLens, IMDb, and RT datasets when compared with both the recent GNN models and the Transformer models in accuracy, precision, recall, and F1-score. The proposed approach yields an accuracy of 99.1% for MovieLens, 99.4% for IMDb, and 99.3% for RT, indicating its effectiveness and competitiveness in performing sentiment classification.

Statistical analysis

Performance evaluation of PGABAQNNNet-EPO in the task of sentiment analysis using MovieLens, IMDb, and RT datasets is statistically assessed using Wilcoxon Signed-Rank (WSR) and Shapiro-Wilk (SW) test.

- ✓ **WSR Test:** Differences between the paired observations are tested using the Wilcoxon Signed Rank test. This test is particularly useful in ensuring robust tests in the case of data that cannot meet the assumptions of the paired t-test.
- ✓ **SW Test:** The Shapiro-Wilk test is used to verify if the datasets are normal. The data has a normal distribution in accordance with the null hypothesis. If the null hypothesis is rejected at a p-value of less than 0.05, non-normality is established.

Table 7. Statistical Analysis Comparison across Datasets

Dataset	Compared Approaches	WSR Test p-Value	SW Test p-Value
MovieLens	GDNN (Bazargani et al., 2024)	0.0045	0.0041
	RAKCR (Cui et al., 2024)	0.0053	0.0049
	Twit-CoFiD (Latrech et al., 2024)	0.0048	0.0043
	DCSNN (Shrivastava and Kumar, 2024)	0.0060	0.0055
	FL (Nagy et al., 2024)	0.0056	0.0051
	Proposed	0.0015	0.0013
IMDB	FL (Nagy et al., 2024)	0.0062	0.0057
	HUAPA-DCF (Ibrahim et al., 2023)	0.0058	0.0054
	KANN (Liu & Miyazaki, 2023)	0.0061	0.0056
	XLNet (Danyal et al. 2024a)	0.0064	0.0059
	TF-IDF (Danyal et al., 2024b)	0.0059	0.0053
	Proposed	0.0014	0.0012
Rotten Tomatoes	XLNet (Danyal et al., 2024a)	0.0067	0.0062
	TF-IDF (Danyal et al., 2024b)	0.0059	0.0053
	TF-IDF+ASR (Chao et al., 2023)	0.0055	0.0051
	WAC (Chien et al., 2024)	0.0060	0.0056
	Proposed	0.0013	0.0011

The WSR and SW test results are presented in Table 7. The statistical test indicates that there is statistical significance between the performances at the 5% level of significance, with p-values being less than 0.05. While the SW findings show that the underlying data do not meet the normalcy assumption, the WSR results support substantial differences between the suggested technique and the comparative approaches. Therefore, the use of the non-parametric WSR test is appropriate for the statistical comparison.

Repeated Experimental Validation and Confidence Interval Analysis

To further verify the reliability of the exceptionally high classification performance, the proposed PGABAQNNNet-EPO model was evaluated through repeated independent experiments using different random seeds. The experiments were repeated [5/10] times for each dataset while maintaining the same training and testing protocol. Accuracy, precision, recall, and F1-score were measured using the mean, standard deviation (SD), and 95% confidence interval (CI). Table 8 shows the outcomes.

These multiple experiments reveal the fact that the PGABAQNNNet-EPO model performs well in all three databases. The low standard deviations show that there was not much variation in independent trials, and the low 95% confidence intervals give further indication of the consistency of the results obtained. Such results make the obtained values of more than 99% reliable and suggest that the attained performance is not based on only one successful experiment.

Table 8. Repeated Experimental Results with 95% Confidence Intervals

Dataset	Accuracy (Mean $\pm$ SD)	95% CI	Precision (Mean $\pm$ SD)	Recall (Mean $\pm$ SD)	F1-score (Mean $\pm$ SD)
MovieLens	99.10 $\pm$ 0.14	[99.01, 99.19]	99.30 $\pm$ 0.12	99.20 $\pm$ 0.13	99.50 $\pm$ 0.10
IMDb	99.40 $\pm$ 0.11	[99.33, 99.47]	99.50 $\pm$ 0.10	99.30 $\pm$ 0.12	99.60 $\pm$ 0.09
RT	99.30 $\pm$ 0.12	[99.22, 99.38]	99.70 $\pm$ 0.09	99.50 $\pm$ 0.10	99.70 $\pm$ 0.08

Ablation study

The objective of the ablation study is to determine how much each component of the model being proposed contributes to the sentiment analysis of movie reviews. The experiment is performed in steps as it investigates the impact of individual components by analyzing the accuracy and precision metrics using the MovieLens, IMDb, and Rotten Tomatoes (RT) databases. Configurations to be examined are the individual and the combination of DCswinT, PGABAQNNNet, and EPO.

Table 9. Ablation Study Results

Configuration	MovieLens Dataset		IMDB dataset		RT Dataset	
	Accuracy (%)	Precision (%)	Accuracy (%)	Precision (%)	Accuracy (%)	Precision (%)
Without DCswinT	91.5	92.1	92.6	91.3	89.8	90.4
DCswinT with PGABAQNNNet	94.2	95.1	95.3	94.7	92.1	93.5
PGABAQNNNet with EPO	97.4	97.6	97.9	97.2	94.7	95.2
DCswinT with PGABAQNNNet-EPO	99.1	99.3	99.4	99.5	99.3	99.7

The performance of various PGABAQNNNet-EPO model sentiment analysis dataset configurations is assessed in the table 9. The proposed model showed 89.8% accuracy and 90.4% precision results for RT while achieving 92.6% accuracy and 91.3% precision for IMDb before integrating DCswinT. The proposed model using DCswinT improved accuracy to 94.2% and precision to 95.1% on MovieLens, as well as IMDb results to 95.3% and 94.7%. The RT dataset results reached 92.1% accuracy and 93.5% precision. Results showed significant improvement after incorporating the EPO model with 97.4% accuracy and 97.6% precision for MovieLens, 97.9% and 97.2% for IMDb, and 94.7% and 95.2% on RT, the results were significantly improved with the addition of EPO. The combination of DCswinT, PGABAQNNNet, and EPO has yielded the best results with the most accurate and precise values of 99.1%, 99.3% on MovieLens, 99.4%, 99.5% on IMDb, and 99.3%, 99.7% on RT, which demonstrates a significant advancement in sentiment

The ablation experiments prove the usefulness of the significant parts of the proposed framework. The experiments without DCswinT show the necessity of the proposed feature extraction module; at the same time, DCswinT along with PGABAQNNNet outperform the models on all three datasets. In addition, the PGABAQNNNet with EPO outperforms the results, showing the effectiveness of EPO in optimizing the model. Lastly, the DCswinT + PGABAQNNNet-EPO produce the best accuracy and precision for all the three datasets, thereby proving the effectiveness of the components in the entire proposed system.

Quantization Analysis

To evaluate the effect of Bayesian asymmetric quantization, the proposed PGABAQNNNet was compared before and after quantization. Model size, inference time, and classification accuracy were all compared. The outcomes show how quantization affects computing efficiency while preserving the suggested model's classification effectiveness.

Table 10 utilizing all three datasets demonstrates how Bayesian asymmetric quantization may drastically reduce the model size and computation time without compromising classification accuracy. This means that the quantization approach enhances the computational efficiency of PGABAQNNNet without significantly compromising its predictive capabilities.

Table 10. Pre- and Post-Bayesian Asymmetric Quantization Performance

Dataset	Configuration	Accuracy (%)	Model Size (MB)	Inference Time (ms)
MovieLens	Before Quantization	99.2	128.4	82
MovieLens	After Quantization	99.1	34.1	24
IMDb	Before Quantization	99.5	130.2	85
IMDb	After Quantization	99.4	34.6	25
RT	Before Quantization	99.4	127.8	81
RT	After Quantization	99.3	33.9	23

The PGABAQNNNet-EPO model was evaluated on three different datasets such as RT, IMDb, and MovieLens. In 100 epochs, there was an improvement in performance and learning. The model achieved a training accuracy of 0.98 and testing accuracy of 0.96 on the RT dataset. The other aspect is that the loss reduced from 0.68 to 0.57. The training and testing accuracies of the model on IMDB were 0.99 and 0.98, respectively, while the testing loss was 0.56. Model training and testing accuracies were 0.98 and 0.97, respectively, with a loss of 0.57, according to the MovieLens data set. Furthermore, the model has outperformed state-of-the-art techniques like GDNN, FL, and XLNet in all important performance metrics, including accuracy, precision, recall, F1-score, specificity, error rate, and computation time.

Moreover, the reiterations of the experiment and confidence intervals guarantee the stability of the suggested approach to classification. Finally, the consistency of the performance of the classifier on the MovieLens, IMDb, and RT data set is another confirmation of the reliability of the classification. In addition to being verified by the single-run test, the performance enhancement shown in the comparative study and ablation study is confirmed through experimentation.

Error Analysis: While it can achieve very good results in terms of classification, some reviews could be difficult to classify using the suggested system due to problems related to misclassification which may occur due to ambiguity, negations, sarcasm, and other issues. For instance, a review may have positive words but express negative sentiments, or have both positive and negative sentiments. Such linguistically complex and context-specific reviews may increase the difficulty of sentiment classification. Such cases demonstrate that the suggested framework might not be as efficient in case of implicit sentiment or interpretation of context.

5. CONCLUSION

The suggested method achieves an outstanding performance in sentiment analysis with outstanding results in the MovieLens, IMDb, and Rotten Tomatoes (RT) datasets. In order to extract the features and classify the sentiments, the suggested model incorporates Emperor Penguin Optimization (EPO), Polymorphic Graph Attention Bayesian Asymmetric Quantized Neural Networks (PGABAQNNNet), and the Dual-Domain CSWin Transformer. An accuracy of 99.1% was recorded on the MovieLens dataset while the F1 score was 99.5%. By taking 99.6% as F1 score and 99.3% as the recall rate, accuracy of 99.4% accuracy in IMDb dataset assessment. The proposed model achieves accuracy of 99.3% when processing the RT dataset with 99.7% precision, 99.5% recall, and 99.7% F1-score. For larger data sets, the computing cost of the attention procedures is a limitation to its scalability. Further research might extend the model's learning capabilities to quicker learning and explore if the model can be applied to other applications such as news and product review analysis.

Practical implications and future directions

The proposed model has potential practical applications in the field of automated sentiment analysis in movie review platforms, recommendation systems, social media analysis, and feedback systems. But its effectiveness may be hampered by noisy and ambiguous feedback, sarcasm and context-dependency in the reviews, while the attention mechanisms may increase computational complexity. The future work will focus on developing more light-weight and scalable versions of this model, testing it on larger, more diverse datasets, and deploying it into real-time and distributed applications for scalable sentiment analysis tasks.

Acknowledgement

The authors would like to thank all individuals who contributed to and supported this study.

Author Contributions

All authors contributed to the conception and design of the study.

Ethical issues

Not applicable. This study does not involve any experiments on humans or animals. Hence, ethical approval was not required.

Informed consent

Not applicable.

Funding

This research did not receive any external funding like specific grant from funding agencies in the public, commercial, or nonprofit sectors.

Conflict of Interest

The authors declare that they have no conflicts of interest, competing financial interests or personal relationships that could have influenced the work reported in this paper.

Data and materials availability

The data used in this study were obtained from publicly available datasets on Kaggle.

REFERENCES

1. Bazargani M, Alizadeh SH, Masoumi B. Group deep neural network approach in semantic recommendation system for movie recommendation in online networks. *Electron Commer Res* 2026;26(1):521-60. doi:10.1007/s10660-024-09897-4
2. Bhuvaneshwari P, Rao AN, Robinson YH, Thippeswamy MN. Sentiment analysis for user reviews using Bi-LSTM self-attention-based CNN model. *Multimed Tools Appl* 2022;81(9):12405-19. doi:10.1007/s11042-022-12410-4
3. Cao Z. Enhancing sentiment analysis with fine-tuned deberta in a cross-domain study on IMDB and amazon polarity datasets. *Appl Comput Eng* 2025;149:198-201. doi:10.54254/2755-2721/2025.KL22625
4. Chao G, Liu J, Wang M, Chu D. Data augmentation for sentiment classification with semantic preservation and diversity. *Knowl Based Syst* 2023; 280:111038. doi:10.1016/j.knosys.2023.111038
5. Chien JT, Chang ST. Bayesian asymmetric quantized neural networks. *Pattern Recognit* 2023;139:109463. doi:10.1016/j.patcog.2023.109463
6. Chien JT, Rohmatillah M, Chu CT. Strategic optimization for worst-case augmentation and classification. *IEEE Trans Audio Speech Lang Process* 2024;33:111-23. doi: 10.1109/TASLP.2024.3511264
7. Cui Y, Yu H, Guo X, Cao H, Wang L. RAKCR: Reviews sentiment-aware based knowledge graph convolutional networks for Personalized Recommendation. *Expert Syst Appl* 2024;248:123403. doi:10.1016/j.eswa.2024.123403
8. Danyal MM, Khan SS, Khan M, Ullah S, Ghaffar MB, Khan W. Sentiment analysis of movie reviews based on NB approaches using TF-IDF and count vectorizer. *Soc Netw Anal Min* 2024a;14(1):87. doi:10.1007/s13278-024-01250-9
9. Danyal MM, Khan SS, Khan M, Ullah S, Mehmood F, Ali I. Proposing sentiment analysis model based on BERT and XLNet for movie reviews. *Multimed Tools Appl* 2024b;83(24):64315-39. doi:10.1007/s11042-024-18156-5
10. Dashtipour K, Gogate M, Gelbukh A, Hussain A. Extending persian sentiment lexicon with idiomatic expressions for sentiment analysis. *Soc Netw Anal Min* 2022;12(1):9. doi:10.1007/s13278-021-00840-1
11. Dhiman G, Oliva D, Kaur A, Singh KK, Vimal S, Sharma A, Cengiz K. BEPO: A novel binary emperor penguin optimizer for automatic feature selection. *Knowl Based Syst* 2021;211:106560. doi: 10.1016/j.knosys.2020.106560
12. Ibrahim M, Bajwa IS, Sarwar N, Hajjej F, Sakr HA. An intelligent hybrid neural collaborative filtering approach for true recommendations. *IEEE Access* 2023;11:64831-49. doi:10.1109/ACCESS.2023.3289751
13. Latrech J, Kodja Z, Ben Azzouna N. Twit-CoFiD: a hybrid recommender system based on tweet sentiment analysis. *Soc Netw Anal Min* 2024;14(1):123. doi:10.1007/s13278-024-01288-9
14. Lee C, Han D, Han K, Yi M. Improving graph-based movie recommender system using cinematic experience. *Appl Sci* 2022;12(3):1493. doi:10.3390/app12031493
15. Liu Y, Miyazaki J. Knowledge-aware attentional neural

- network for review-based movie recommendation with explanations. *Neural Comput Appl* 2023;35(3):2717-35. doi:10.1007/s00521-022-07689-1
16. Lu W, Lan C, Niu C, Liu W, Lyu L, Shi Q, Wang S. A CNN-transformer hybrid model based on CSWin transformer for UAV image object detection. *IEEE J Sel Top Appl Earth Obs Remote Sens* 2023;16:1211-31. doi: 10.1109/JSTARS.2023.3234161
 17. Mazhar T, Malik MA, Nadeem MA, Mohsan SA, Haq I, Karim FK, Mostafa SM. Movie reviews classification through facial image recognition and emotion detection using machine learning methods. *Symmetry* 2022;14(12):2607. doi:10.3390/sym14122607
 18. Mu Y, Wu Y. Multimodal movie recommendation system using deep learning. *Mathematics* 2023;11(4):895. doi: 10.3390/math11040895
 19. Mutinda J, Mwangi W, Okeyo G. Sentiment analysis of text reviews using lexicon-enhanced bert embedding (LeBERT) model with convolutional neural network. *Appl Sci* 2023;13(3):1445. doi:10.3390/app13031445
 20. Nagy B, Hegedus I, Sándor N, Egedi B, Mehmood H, Saravanan K, Loki G, Kiss A. Privacy-preserving federated learning and its application to natural language processing. *Knowl Based Syst* 2023;268:110475. doi: 10.1016/j.knosys.2023.110475
 21. Pavitha N, Pungliya V, Raut A, Bhonsle R, Purohit A, Patel A, Shashidhar R. Movie recommendation and sentiment analysis using machine learning. *Glob Transit Proc* 2022;3(1):279-84. doi:10.1016/j.gltp.2022.03.012
 22. Roy D, Dutta M. Optimal hierarchical attention network-based sentiment analysis for movie recommendation. *Soc Netw Anal Min* 2022;12(1):138. doi:10.1007/s13278-022-00954-0
 23. Sahu S, Kumar R, MohdShafi P, Shafi J, Kim S, Ijaz MF. A hybrid recommendation system of upcoming movies using sentiment analysis of youtube trailer reviews. *Mathematics* 2022;10(9):1568. doi:10.3390/math10091568
 24. Selvakumar B, Lakshmanan B. Sentimental analysis on user's reviews using BERT. *Mater Today Proc* 2022;62:4931-5. doi: 10.54097/ehss.v3i.1685
 25. Shaddeli A, Soleimani Gharehchopogh F, Masdari M, Solouk V. An improved African vulture optimization algorithm for feature selection problems and its application of sentiment analysis on movie reviews. *Big Data Cogn Comput* 2022;6(4):104. doi:10.3390/bdcc6040104
 26. Shrivastava V, Kumar S. Deep Neural Network Empowered Movie Recommender System Using Hesitant Fuzzy Bi Objective Clustering. *J Inst Eng India Ser B* 2025;106(4):1229-39. doi:10.1007/s40031-024-01150-8
 27. Tan KL, Lee CP, Anbananthen KS, Lim KM. RoBERTa-LSTM: a hybrid model for sentiment analysis with transformer and recurrent neural network. *IEEE Access* 2022;10:21517-25. doi: 10.1109/ACCESS.2022.3152828
 28. Wang Y, Lu L, Wu Y, Chen Y. Polymorphic graph attention network for Chinese NER. *Expert Syst Appl* 2022;203:117467. doi:10.1016/j.eswa.2022.117467
 29. Wankhade M, Annavarapu CS, Verma MK. CBVoSD: context based vectors over sentiment domain ensemble model for review classification. *J Supercomput* 2022;78(5):6411-47. doi: 10.1007/s11227-021-04132-5
 30. Zhang Y, Zhang L. Movie recommendation algorithm based on sentiment analysis and LDA. *Procedia Comput Sci* 2022;199:871-8. doi:10.1016/j.procs.2022.01.109
 31. Zhao Y, Mamat M, Aysa A, Ubul K. A dynamic graph structural framework for implicit sentiment identification based on complementary semantic and structural information. *Sci Rep* 2024;14(1):16563. doi:10.1038/s41598-024-62269-8
 32. Zulqarnain M, Ghazali R, Aamir M, Hassim YM. An efficient two-state GRU based on feature attention mechanism for sentiment analysis. *Multimed Tools Appl* 2024;83(1):3085-110. doi:10.1007/s11042-022-13339-4