International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(21s) 2026 pp. 2180-2197
https://doi.org/10.70917/ijcisim-2026-5961
Article

A Comparative Study of Metaheuristic Feature Selection and Optimized Deep Learning Models for Customer Churn Prediction in Banking

Dr. S.Vimala[1], B.Thenmozhi[2]

[1]Professor, Dept. of Computer Science, Mother Teresa Women's University, Kodaikanal, Tamilnadu, India., vimalaharini@gmail.com
[2]Ph.D Research Scholar, Mother Teresa Women's University, Kodaikanal, Tamilnadu, India., phdcs19p614bt@gmail.com

Abstract: Customer Churn Prediction plays a vital role in retaining customers in the highly competitive bank sector. The approaches used to preprocess data determine how effective deep learning and machine learning models are, even if they produce high-quality results, perform feature selection and optimize the model. This paper centers on the analysis of customer churn prediction through deep learning techniques using a dataset obtained from the Kaggle website. Data preprocessing with three steps like missing value imputation methods, data encoding, and normalization methods has been introduced for dataset. After preprocessing, data balancing is performed by Improved Synthetic Minority Oversampling Technique (ISMOTE) and Improved K-Nearest Neighbor (IKNN) is utilized to solve the data imbalance by giving minority groups the majority class on a regular basis. Feature Selection (FS) methods like Brownian Movement Dragonfly Optimization Algorithm (BMDOA), and Levy Flight Based Whale Optimization Algorithm (LFWOA) is applied to identify the most significant features for prediction. Convolution and Attention with Bi-directional Gated Recurrent Unit (CAT-BiGRU), Modified Weight and Activation Bidirectional Gated Recurrent Unit (MWABGRU), and Hyperparameter Tuned and Soft Activation Deep Multi-Layer Perceptron (HTSADMLP) have all been implemented to predict churn. The existing methods, Classification Time, and evaluation metrics like Area Under Curve (AUC), Kappa, Brier Score, and Expected Maximum Profit of the Customer (EMPC) are used to analyze the performance of the proposed method.

Keywords: Customer churn, banking sector, classification, data preprocessing, balancing, feature selection, deep learning techniques, metaheuristic methods

1. INTRODUCTION

The banking sector is highly competitive with several options for customers. It is therefore a challenge for financial institutions to keep their customers [1] that has become critical. One of the most important problems with this challenge is customer churn, which is measured by the churn rate which shows the percentage of customers who terminate their banking relationship during a stated period of time [2]. Customer churn is a multifaceted issue that affects a bank's productivity, profitability and viability. A loss of existing customers negatively impacts financial results as a result of the loss of interest income, service fees and other revenue sources. Moreover, the process of gaining new customers is usually more expensive than keeping the current customers. High customer turnover can hinder a bank's ability to increase its number of customers and grow market share. Hence, banks need to take steps to handle customer turnover by focusing on customer retention [3].

Customer churn can be prevented by ensuring quality service delivery, better products and services, clear and direct pricing, effective communication, and constant improvement in services. The maintenance of a good customer base necessitates the determination and addressing of reasons for customer churn [4]. Many researchers have been conducted on the key elements that can play a significant role in retaining the customers and causing them to leave. It

Copyright: © 2026 by the authors

is important to offer easy-to-navigate digital platforms for online and mobile banking and other digital services. Additionally, it is crucial to make use of technologies for making transactions easy and secure [5,6].

Customer Churn Prediction (CCP) refers to the determination of customers who have the potential to stop using the products or services [7]. Machine learning uses sophisticated techniques to analyse and predict customer attrition through evaluation and modelling of customer behaviours. Companies can use the information obtained from customer churn prediction to develop ways of retaining the customers and improving customer satisfaction.

The advancement of technology within the last few decades has facilitated the gathering and storing of data about the clients by banks and several other service providers, who further divide them into two types of churners and non-churners. Machine learning (ML) and deep learning (DL) models have been used for customer churn prediction through electronic banking [8, 9].On the other hand, for structured data, high predictive power has been demonstrated by machine learning techniques as boosting algorithms, Random Forest (RF), Decision Tree (DT), and Support Vector Machine (SVM) [10,11]. Furthermore, deep learning models such as Transformer-based models, Convolutional Neural Networks (CNN), Artificial Neural Networks (ANN), and Long Short-Term Memory networks (LSTM) offer substantial benefits for handling sequential and unstructured data, such as textual feedback and customer interactions.

These methods are greatly influenced by various data-related issues such as class imbalance, irrelevant features, and missing values in the data set. The problem of class imbalance, in which non-churn customers vastly outnumber churn customers, can result in inaccurate conclusions made by the model; a model that is more likely to predict the majority class than the minority class and ends up mispredicting minority instances. Also, if the features are not relevant to the problem and/or redundant, they make the data set high-dimensional and will require more computation cost to achieve, and will lead to overfitting. Uncertainty and lack of confidence in the learned patterns further compromises model performance when there are missing values. Hence, for improving the reliability and accuracy of the models, efficient processing, feature selection and data balancing methods are required.

Feature selection is known as a Non-Deterministic Polynomial-time hard (NP-hard) problem, which involves choosing the optimal subset of characteristics for enhancing the quality of decision making. In the last few years, metaheuristic algorithms have proven to be effective in feature selection. All of the above algorithms are based on full exploration of the search space and can provide a good degree of accuracy for multidimensional problems [12]. The process of selecting features might also help in removing noise from the dataset used by the model, this will enhance the model's performance.

First, the quality and consistency of the data are improved by applying some pre-processing techniques, such as encoding, normalization, or mean/mode imputation. Then, techniques are used to tackle class imbalance and boost the representation of minority classes such as ISMOTE and the proposed IKNN. Thirdly, the selected features are most important and are used to reduce the dimension to make the model efficient using feature selection methods such as BMDOA and LFWOA. Lastly, various classification models are evaluated for customer churn prediction, including HTSADMLP, MWABGRU and the proposed CAT-BiGRU, and the optimized CAT-BiGRU model shows high accuracy and robustness.

2. LITERATURE REVIEW

Văduva et al., [2] the first step involved selecting features that have a direct impact on the customer churn phenomenon. This was done using the Synthetic Minority Over-sampling Technique with Tomek Links (SMOTETomek). SMOTETomek combines Tomek Links to eliminate noisy or borderline occurrences with SMOTE to simulate the minority class. The RF and Light Gradient Boosting Machine (LGBM) classifiers are two advanced machine learning models that were employed in this work. Certain metrics, including precision, sensitivity, accuracy, F1 score, and Brier score, which aid in calibrating predicted probabilities, can be used to assess the performance of these models. The expected probabilities are calibrated using probability calibration techniques to make them more interpretable and reliable. In addition to this, another explainable artificial intelligence (XAI) method called Shapley Additive Explanations (SHAP) is used to make the decision making process more transparent and credible.

A highly accurate prediction model for identifying potential churners in the banking industry was designed by Mansoor et al. [13] through the use of different machine learning approaches. Different balancing methods were used to handle class imbalance in the dataset, which include SMOTE and Adaptive Synthetic Sampling (ADASYN). Highly correlated features were determined through FCBF and RFE. Then, the identified features underwent various data mining techniques, like DT, RF, and ANN. The model that performs the best, according to the models' evaluation, is the one built on the balanced dataset from ADASYN through RFE-selected features.

Nisha and Pramod [14] developed a technique for increasing the accuracy of predictions in high dimensional datasets by means of feature selection using a Genetic Algorithm (GA). Several feature selection-sensitive machine learning algorithms are found throughout this process. The model is evaluated using a primary dataset and contrasted with alternative models. Moreover, the technique helps in identifying those features that have the most influence in the prediction of customer churn for GA-based classification models. This helps banks in changing their marketing policies to attract more customers.

Rahman and Kumar [15] developed a way to predict the customer churn in the bank with the help of ML algorithms. The technique helps in exploring the probability of churn through customer behavior analysis. KNN, SVM, DT, and RF classification techniques have been employed in this study. Furthermore, to evaluate the system's performance, feature selection approaches have also been used to select the most pertinent features. The experimentation was carried out on the churn data set available on Kaggle. Comparison is done to identify a suitable model having high precision and predictability. The RF model with oversampling is superior to other models based on accuracy.

Ehsani and Hosseini [16] performed feature reduction to remove the irrelevant features from the data and represent it in a reduced dimensional space. The technique RFECV is employed for selecting the best features that can be practically employed. Thereafter, feature importance is applied for scoring each input feature. C5.0, Light Gradient Boosting Machine (LightGBM), AdaBoost, CatBoost, eXtreme Gradient Boosting (XGBoost), and RF is considered for performing classification tasks. This technique involves feature reduction, feature selection, and feature importance. Also, feature optimization on tree-based classifiers was conducted by the author to enhance performance. This technique is novel because it combines feature optimization techniques and tree-based classifiers to properly describe and capture customer churn experience in internet banking services marketing.

Domingos et al., [17] made an experimental study on how various hyperparameters affect the behavior of DNN for banking industry churn prediction. In this experiment, DNN was utilized using the sigmoid function in the output layers and the rectifier function in the hidden layers. This gives some insights into how various batch sizes affect the performance of DNN in predicting churn in the banking sector. The performance of DNN was improved by using a small batch size than the test set data size, and Root Mean Square Propagation (RMSProp) has shown better accuracy in comparison with other algorithms like Adaptive Gradient Algorithm (AdaGrad), Adadelta, AdaMax, Adaptive Moment Estimation (Adam), and Stochastic Gradient Descent (SGD).

Banu et al., [18] suggested an AI-based CCP approach for Telecommunication Business Markets (AICCP-TBM). The AICCP-TBM strategy was created to manage the number of churners and non-churners in the communications sector. Feature selection is done using an AICCP-TBM method designated as Chaotic Salp Swarm Optimization-based Feature Selection (CSSO-FS).The Fuzzy Rule-based Classifier (FRC) is used to distinguish between client churners and non-churners. The membership functions of the FRC model are selected using the Quantum Behaved Particle Swarm Optimization (QPSO) method. The QPSO algorithm, whose evolution equation is easier and which needs less number of parameters as compared to classical PSO, facilitates a more simplified process of convergence and control. The experimental results have been examined from multiple perspectives, and the AICCP-TBM model has been verified using a benchmark CCP dataset. The proposed AICCP-TBM model performed better than the existing state-of-the-art CPP models from the perspective of presentation, according to the imitation results.

Jajam et al., [19] have proposed the Recurrent Neural Network (RNN) model and Stacked Bidirectional Long Short-Term Memory (SBLSTM) for the Arithmetic Optimization Algorithm (AOA) in CCP. The objective of this

proposed model is to accurately forecast the probability of client attrition in the insurance sector. To transform the unprocessed input into useful data, the AOA model first performs pre-processing. Additionally, whether or not customers are churning is categorized using the SBLSTM-RNN model. An optimum hyperparameter tuning technique based on the Improved Gravitational Search Optimization Algorithm (IGSA) is used to improve the performance of CCP utilizing the SBLSTM-RNN model. The health insurance data was used to check the performance of our approach, and there are four different experiments that have been performed using this algorithm in classification.

AlShourbaji et al. [20] proposed the Enhanced Gradient Boosting Model (EGBM), which may improve GBM learning by using Support Vector Machine with Radial Basis Function (SVM-RBF) as a base learner with an exponential loss function. The new base learner considerably increases the classification ability of GBM at the very beginning and provides better results for CP-EGBM during each iteration of boosting by using advanced decision tree learners. The consumption operator from the Artificial Ecosystem Optimization (AEO) algorithm has been added to an enhanced version of the Particle Swarm Optimization (PSO) algorithm to prevent PSO from prematurely converging to local optima. The CP-EGBM's hyper-parameters have been optimized using the proposed strategy. The efficacy of the proposed model has been assessed using a variety of indicators on seven accessible CP datasets. The CP-EGBM has been shown to perform better than the GBM and SVM models.

3. PROPOSED METHODOLOGY

This study presents a comprehensive comparative framework integrating data preprocessing, data balancing, feature selection, and optimized deep learning models. Initially, the raw banking dataset is preprocessed using mean and mode imputation, categorical encoding using one-hot and label encoding, and decimal scaling, Z-score normalization, and Min-Max normalizing are examples of multi-stage normalizing methods, to ensure data consistency and stability. ISMOTE and the IKNN balancing method are conducted to overcome the class imbalance problem. For feature selection, Brownian Movement Dragonfly Optimization Algorithm (BMDOA) and the proposed Lévy Flight-Based Whale Optimization Algorithm (LFWOA) metaheuristic approaches are introduced for dataset. Furthermore, the study compares three deep learning models: (i) HTSADMLP, (ii) MWABGRU, and (iii) the proposed CAT-BiGRU optimized using Adaptive Firefly Optimization (AFO). Performance metrics are used in extensive experiments. The general procedure of the proposed model is depicted in Figure 1.

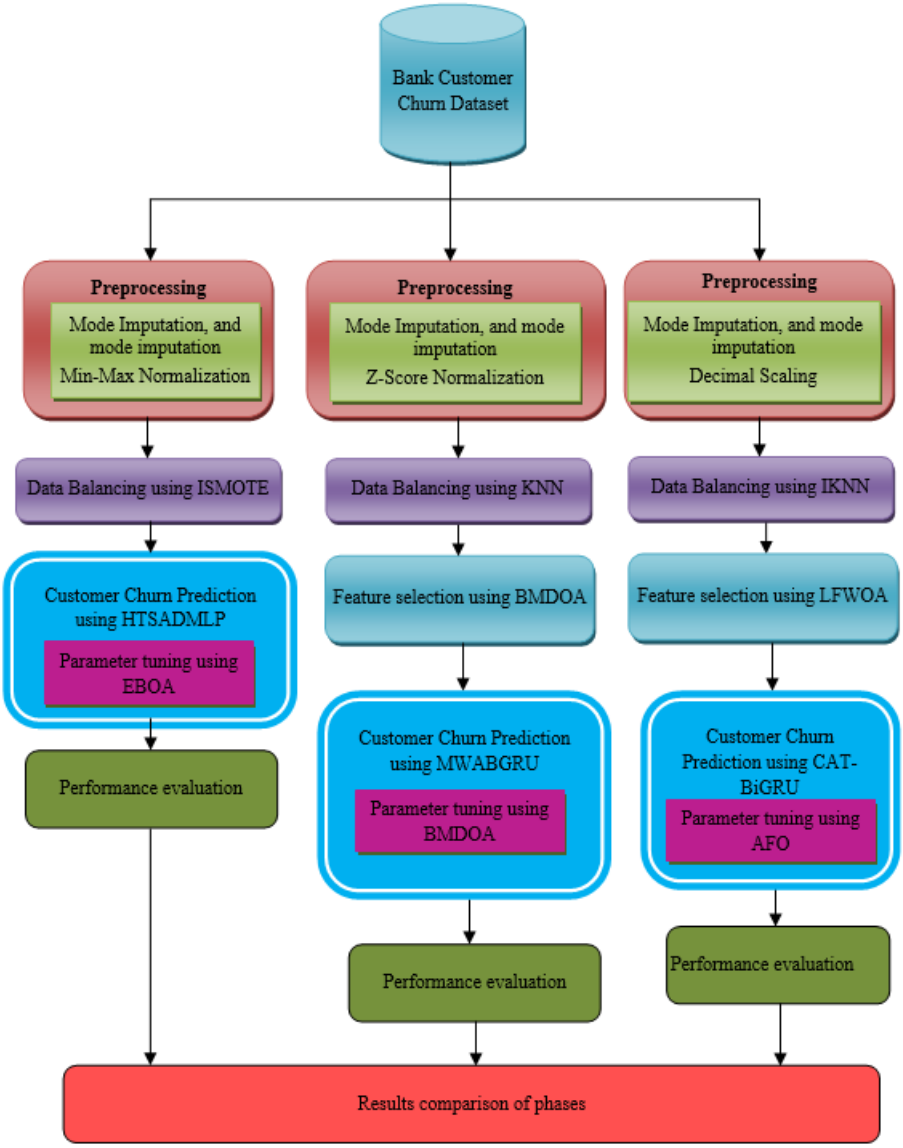


FIGURE 1. OVERALL RESEARCH WORKFLOW

3.1 Data Preprocessing methods

The initial source of the banking client data was the publicly available Kaggle dataset. The process of Customer Churn Prediction using the proposed method includes of data preprocessing with three phases like missing value imputation, data encoding, and normalization,

Mean imputation and mode imputation: The process that handles missing data in the data set is the use of imputation. Mean imputation, which involves calculating the mean of the non-missing observations in the column and substituting the computed mean for the missing values, is used when there are missing values for numerical columns like balance, age, and credit score. Mode imputation is used for categorical columns where the missing values are filled in using the most commonly occurring category.

One-Hot Encoding: It involves encoding the categorical data like gender and geographical factors to numbers to improve compatibility with machine learning algorithms. For geographical factors, one-hot encoding was used while for gender factors, labeling encoding was used with 0 representing female and 1 representing male.

Data normalization: To confirm that each characteristic provides an equal contribution to the process of learning, data normalization is a preprocessing technique that reduces numerical values into a corresponding range or distribution. This work follows three normalization methods,

Mix Max Normalization: The dataset features are normalized using Min-Max Normalization (MMN), which normalizes the characteristics' maximum value of 1 and their minimum value of 0, shifting all values to the same scale between 0 and 1. Therefore, learning will be equally affected by both small and high feature values. The following is the equation (1),

$$MMN = \left(\frac{SI - SI_{min}}{SI_{min} - SI_{max}}\right) \tag{1}$$

where, $SI_{min}$and $SI_{max}$are the dataset's feature's maximum and minimum values.

The major limitation of Min-Max Normalization is highly sensitive to outliers, as extreme values affect scaling, and it doesn't preserve the original data distribution. It has been solved by introducing a Z-Score Normalization (ZSN).

Z-Score Normalization (ZSN): The values of a feature are scaled using this method so that the standard deviation is one, while the mean is zero. This is achieved by dividing the result by the standard deviation after reducing the feature mean of each dataset value. The major limitation of this method is that the transformed values are not bounded (can be negative or very large), and data follows a normal distribution. Interpretation of scaled values is less sensitive. It has been solved by using decimal scaling. The equation (2) is given as follows,

$$ZSN = \left(\frac{SI - \mu}{\sigma}\right) \tag{2}$$

where $\mu, \sigma$is the dataset's feature's mean and standard deviation.

Decimal scaling: Decimal scaling, numerical values are scaled by dividing them by a factor of ten, ensuring that all values fall within a consistent range. This process helps in preventing certain features from dominating others due to their large numerical values.

Z-Score Normalization overcomes the limitations of Min-Max Normalization by providing robustness to outliers and preserving statistical properties, while Decimal Scaling further improves simplicity and ensures bounded feature values without requiring distribution assumptions. To enhance the model's performance, several actions are performed on the collected dataset.

3.2 Data balancing methods

Following the preprocessing stage, data balancing becomes important, since unbalance in the data may lead to inaccurate results during the classification process, whereby the minor classes tend to be labeled as major classes. The problem of class imbalance arises if there are considerable differences between the classes, especially if the minor classes dominate churn classes. The techniques used to address this problem are ISMOTE and Improved K-Nearest Neighbors (IKNN).

3.2.1 ISMOTE

To address the issue of class imbalance in this study, ISMOTE might be employed. The SMOTE method is an oversampling approach that creates artificial data points of by producing new data points for the minority class between the minority data point and its randomly selected nearest neighbor. This method retains all information because there is no reduction of data during imputation.Despite its simplicity, SMOTE suffers from a marginalization problem that can lead to increased class overlap or overgeneralization. During the generation of synthetic samples, the algorithm may create new data points along the boundary of the minority class distribution. If a positive (minority) sample lies near the edge of the minority class region, the synthetic samples generated from it and its neighbouring

samples may also fall near this boundary. Consequently, these samples become increasingly marginalized, which may increase the likelihood of overlap with the majority class and reduce classification accuracy. After computing the center points, to produce synthetic data, a minority class sample is selected. Subsequently, one sample is randomly chosen from its k (k = 5) nearest minority class neighbours. k = 5 indicates that for each minority class sample, SMOTE searches for its five nearest neighbors in the feature space. From these neighbors, one or more are randomly selected, and along the line segment that connects the original sample and selected neighbor, synthetic samples are produced. This process helps increase the minority class's representation inside the study. ISMOTE may increase overlap due to boundary sample generation, and degrade accuracy in CCP. It has also the issue of overgeneralization problem in ISMOTE [21].

3.2.2 IKNN

IKNN generates synthetic samples using structured interpolation based on nearest neighbor distances. Unlike ISMOTE, which randomly selects neighbors, IKNN ensures that new samples are created in relevant regions of the feature space, improving data quality. ISMOTE often generates samples near class boundaries, increasing overlap between minority and majority classes. Uses equation (3) to initially determine the center positions of positive and negative samples [21],

$$CE_c = \frac{P_s + N_s}{2} \tag{3}$$

Synthetic samples are created using a minority class sample which is selected after finding out the center points of that particular sample. A neighbour among the minorities is chosen, and then synthetic samples are generated$DT_m$by interpolating between samples using equation (4),

$$DT_m = SA + Rand(0,1) \times (CE_c - SA) \tag{4}$$

where$Rand(0,1)$indicates a data sample in minority-class sampling and is a random number between 0 and 1.

IKNN solves this by carefully selecting neighbours, thereby reducing misclassification.IKNN outperforms ISMOTE by generating high-quality synthetic samples with better boundary preservation, reduced noise influence, and improved classification accuracy,

IKNN: IKNN approach is used for data balancing [21]. The steps are discussed below

Step 1: Identify the Minority and Majority Classes

- Given a dataset D with N samples as $D = \{(X_1, Y_1), \dots, (X_N, Y_N)\}$, Xis denoted as the feature vectors and Y represents class labels.
- Identify the minority class C_{min}(e.g., churned customers) and the majority class C_{max}(e.g., retained customers) as $| C_{min} | \ll | C_{max} |$, where $n|C_{min}|$is much smaller than $|C_{max}|$.

Step 2: Compute K-Nearest Neighbours for Minority Class Instances

- Determine the K nearest neighbors of each instance X_i in the minority class C_{min} using the Euclidean distance and equation (5),

$$d(X_i, X_j) = \sqrt{\sum_{n=1}^{m} \left(X_{i,n} - X_{j,n}\right)^2} \tag{5}$$

where m is the number of features, choose a suitable value of K based on dataset distribution (commonly K = 5).

Step 3: Generate Synthetic Data Points Using Interpolation

- For each minority instanceX_i, randomly select one of its K-nearest neighbors, X_{nn}.
- Generate a synthetic data point X_{syn} using the interpolation equation (6),

$$X_{syn} = X_i + \lambda(X_{nn} - X_i) \tag{6}$$

where $X_{syn}$is the synthetic sample, $\lambda$is a random number between 0 and 1, ensuring that the synthetic sample is along the vector connecting $X_i$and X_{nn}.

Step 4: Ensure Class Distribution Balance

Repeat the sample generation until the dataset reaches a balanced state by equation (7),

$$\left|C_{min}^{new}\right| \approx |C_{max}| \tag{7}$$

This ensures that the model has equal exposure to both classes.

3.3 Feature selection methods

The balanced dataset is exposed to Feature Selection (FS) to determine which features are most important for classification. Identifying and selecting the most pertinent and instructive characteristics from a dataset while eliminating redundant, unnecessary or noisy attributes is known as feature selection. This stage prevents overfitting, eliminates unnecessary characteristics, and enhances the model's overall accuracy and efficiency.

3.3.1 Brownian Movement Dragonfly Optimization Algorithm (BMDOA)

The BMDOA algorithm has been proposed for reducing system size, computational complexity, overfitting, and simultaneously for feature selection from the dataset. Although it is easy-to-use and highly effective in searching, the main drawbacks of this algorithm include slow convergence rate and the presence of local optima problem. In order to overcome the above drawbacks, Brownian Movement (BM) technique has been used for updating the positions of dragonflies (DF), which will help in overcoming the above problems. As a result, BM has been added to the conventional DOA algorithm, and BMDOA has been developed [22],

Set the dragonfly population's initial values based on the dataset as $Y_d = \{Y_1, Y_2, Y_3 \ldots\ldots Y_n\}$. The individual's fitness is then assessed as accurately as feasible to determine the fitness value. Equations (8-9) provide a statistical description of it,

$$fitness = Max(Accuracy) \tag{8}$$

$$Accuarcy = \frac{P_{ve}^{+} + P_{ve}^{-}}{TV_L} \tag{9}$$

whereP_{ve}^{+} denotes the true positive (TP), P_{ve}^{-} indicates the true negative (TN), and TV_Lrefers to the balanced dataset's total sample count. The following step function may then be used to determine the dragonflies' new position$\Delta Y_{\tau+1}$which equation (10) models,

$$\Delta Y_{\tau+1} = (sSE_u + aAL_u + cCH_u + fFD_u + eEY_u) + \varpi\ \Delta Y_\tau \tag{10}$$

In this instance, the division, alignment, unity, food attraction, and diversion from the opponent of theu^{-th}each individual was shown as$SE_u, AL_u, CH_u, FD_u, EY_u$.Food, cohesiveness, separation, alignment, and adversary factors are indicated by$s, a, c, f,\ and\ e$. The weight of inertia is represented as ϖ. It shows the iteration counterτ.Equation (11) is used to model,

$$Y_{\tau+1} = Y_\tau + \Delta Y_{\tau+1} \ * \ BM_w \tag{11}$$

$$BM_w = \sqrt{\frac{\alpha}{\chi}} * R_N * PN_m \tag{12}$$

Where,BM_wshows the BM used to enhance the DOA's randomization step. Therefore, an agent's motion time period in seconds is represented byα. In relation to time, the number of rapid moves for the same agent is represented byχ. It shows the random number byR_N. PN_mrepresents the DF's dispersed periodic motion across a time. Until the highest possible number of iterations is reached, the process will be continued. If the DF has neighbours, the position is also updated.

BMDOA enhances the conventional DOA by integrating Brownian motion to improve random exploration and reduce premature convergence. It effectively balances exploration and exploitation through swarm behaviours such as separation, alignment, and cohesion. However, its dependence on small-step random movements can limit its ability to escape complex local optima. Enhances randomness but with small step sizes, slower convergence due to iterative local adjustments, still may get trapped in complex search spaces are the major issues of BMDOA. It has been solved by introducing a LFWOA.

3.3.2 Levy Flight Based Whale Optimization Algorithm (LFWOA)

After balancing the dataset, feature selection is conducted using the LFWOA. This method combines the advantages of Levy flight and the Whale Optimization Algorithm (WOA) to enhance global search ability and convergence speed. LFWOA efficiently selects the most important features while eliminating those that are unnecessary and redundant, leading to improved model performance. WOA is a new generation stochastic optimization algorithm inspired by nature. This algorithm uses a number of search agents to find the optimal solution of an optimization problem. This algorithm is inspired by the foraging behavior of humpback whales and their unique bubble-net hunting technique. There are mainly three stages in the WOA [23].

- **Encircling Prey:** Whales estimate the best solution and adjust their position relative to it.
- **Bubble-Net Attacking Mechanism:** This phase involves either a shrinking encircling method or a spiral position update to refine the search.
- **Searching for Prey:** The whales search the solution space for an optimal solution (the best solution) by exploring different areas of the solution space.

By integrating the Levy flight with Whale Optimization Algorithm, LFWOA greatly improves the global search capability. Strong exploitation is achieved by the bubble-net attacking mechanism, and efficient exploration of the search space by the Levy flight. The hybrid approach enables LFWOA to converge faster and more efficiently determine optimal feature subsets. For more complex and high dimensional optimization problems, LFWOA shows superior global search, convergence speed and quality of feature selection in comparison to BMDOA.

3.4 Customer Churn Prediction methods

Customer Churn Prediction in identifying clients who are likely to discontinue using a service is the objective of customer relationship management. In this work, CCP is performed after data preprocessing, data balancing, and feature selection to ensure high-quality input for the predictive model.

3.4.1 Hyper parameter Tuned and Soft Activation Deep Multi-Layer Perceptron (HTSADMLP)

DMLP is an ANN that maps input vectors to output vectors using multiple feed forwards (FF). Among many other layers, Input Layers (IL), Hidden Layers (HL), and Output Layers (OL) compose this connected graph. The HL and OL include a large number of neurons with activation functions (AF).The selection of the hyperparameter, which is random in DMLP, has a significant impact on the network's rate of convergence on the error rate. To maximize the prediction rate and minimize computing efficiency, the proposed approach selects the hyperparameter (weights and bias) using an Enhanced Butterfly Optimization Algorithm (EBOA).

In addition, since sigmoid and tanh activation functions work effectively in small to moderately large networks, DMLP frequently uses both. However, during backpropagation, the gradient becomes closer to zero as the network increases its depth. This makes them vulnerable to the Gradient Vanishing Problem. Therefore, the weights are updated less frequently. Therefore, in the proposed system, Soft plus activation function with AF is used to overcome the problem of vanishing gradients in the network. This method ensures that the output of a machine is kept positive and is a smooth function as compared to the Rectified Linear Unit (ReLU) Activation Function. In the traditional DMLP method, the process of hyperparameter tuning and activation function modification is known as HTSADMLP. The following are the procedure for HTSADMLP.

Step 1: The neurons in this layer initially receive input (a balanced dataset) and forward it to the network's subsequent levels. Next, use equation (13) to get the input's weighted sum,

$$K_m'' = \sum_{m=1}^{p} \varpi_{m,n} DT_m + B_n \tag{13}$$

where, p −refers to the network's neuron count,DT_m indicates the balanced input dataset, the bias value in the n^{-th} hidden node is represented by B_n, whereas $\varpi_{m,n}$ indicates the connection weight of the m^{-th} node in the IL and the n^{-th} node in the HL. In this case, the DMLP randomly selects the weights and biases, which has an impact on the results of the proposed technique and computing procedure. Therefore, EBOA selects the hyperparameter optimally, as step 2 explains in detail.

Step 2: Hyper parameter Tuning

Using EBOA, the DMLP hyperparameters are typically selected to enhance the performance of the model in a number of ways.

Step 2.1: Initialize the butterfly population at random, with each butterfly serving as a representation of the model's hyperparameters. Let's use equation (14) to get the ith position,

$$Y_i = Y_{i1}, Y_{i2}, \ldots . . Y_{in} \tag{14}$$

where$Y_{ij}$is denoted as the j^{th}hyperparameter of the *ith* butterfly; the representation n stands for the number of hyperparameters. The range is shown below, and the hyperparameters are set at random as$Y_{ij} \in (Y_{low}, Y_{high})$, $Y_{low}\&Y_{high}$is represented as the hyperparameter *j* lower and upper limits space.

Step 2.2: To assess the fitness of butterflies and measure model performance, utilizing the existing set of hyperparameters, train the DMLP model. Equation (15) selects the results with the highest accuracy as the fittest solutions for the current iteration,

$$F(Y_i) = Accuracy\ (Y_i) \tag{15}$$

Step 2.3: Cauchy mutation, which introduces randomness through the Cauchy distribution, may be used to modify the hyperparameters. Equation (16) determines the mutation of each solution,

$$Y_{ij}(new) = Y_{ij} + \partial . \frac{K_{ij}}{1 + \propto^2} \tag{16}$$

where $K_{ij} \sim Cauchy\ (0, \partial)$is represented by an arbitrary number with a scaling parameter from the Cauchy distribution∂, $\propto$is indicated by a random integer that usually ranges between 0 and 1, $\partial$is represented as the magnitude of the mutation step to allow for larger steps in SS. The Cauchy mutation makes it possible for butterflies to efficiently explore new search space areas by enabling more random jumps in SS.

Step 2.4: The locations of the greatest and random butterflies in the world, which is represented by equation (17), are combined to update the butterfly positions,

$$Y_i(new) = Y_i + \mu(Yb - Y_i) + \rho(Y_i - Yr) \tag{17}$$

where, $Yb$is denoted as the butterfly's position with the highest fitness value, $Yr$is represented by the population's random butterfly position and $\mu \& \rho$ is used to represent the coefficients that regulate the impact of both random and global butterflies. By doing thus, exploitation and exploration are balanced.

Step 2.5: To reassess the butterflies' fitness, utilizing the updated hyperparameters, retrain the DMLP model. By comparing the new fitness, update the ultimate worldwide best butterfly. $Y_i(new)$ with the best fitness ($Gbest$) by equation (18),

$$G_{best} = (G_{best}, F(Y_i(new)) \tag{18}$$

If the butterfly's new position provides the best fitness according to equation (19), update the global best solution,

$$Y_{best} = Y_i(new) \tag{19}$$

Step 3: Until the maximum number of iterations is achieved, repeat the previous procedures. The optimal set of hyperparameters for the model is contained in the algorithm's output.

Limited ability to capture temporal dependencies may require careful tuning of multiple hyper parameters. However, in the dataset, BGRU records both short-term and long-term dependence, and improves prediction accuracy. MWABGRU leveraging bidirectional learning and optimized weight selection, making it more suitable for customer churn prediction tasks.

3.4.2 Modified Weight and Activation Bidirectional Gated Recurrent Unit (MWABGRU)

MWABGRU is introduced for CC classification from the optimal feature set. A BGRU processes input sequences in both forward and backward directions, allowing both short-term and long-term dependencies to be captured by the model. A GRU consists of three key components: the update gate, reset gate, and candidate hidden state. These components use learnable weight parameters to generate the output. However, the randomly initialized weight matrices increase computational complexity and require additional training time for parameter optimization. The weight parameters are optimally determined using the BMDOA algorithm, which reduces computational complexity for the given input features. This optimization also improves classification accuracy while decreasing processing time. The resulting model, which combines optimally optimized weights with modified activation functions, is termed the MWABGRU. Initially, the forward GRU receives the optimal feature set generated by the BMDOA algorithm as its input for feature learning. The forward hidden state and the forward hidden layer are

combinedh_{t-1} at time step t with the current input x_t to perform a number of linear transformations and activation functions. The update gate z_t and the reset gate r_t are the two gates that constitute each GRU unit.The update gate regulates the amount of information transferred from the previous hidden state to the current state, allowing relevant information to be retained. In contrast, the reset gate regulates how much data from the prior hidden state is eliminated, preventing irrelevant information from influencing the current computation. The network can achieve this access to the gating mechanism preserve meaningful patterns while filtering out redundant information during training. The operations of the forward GRU can be mathematically represented by Equations (20)–(23),

$$U_t = \eta^*(W_U\ [H_{t-1} \rightarrow,\ F_t]) \tag{20}$$

$$R_t = \eta^*(W_R\ [H_{t-1} \rightarrow,\ F_t]) \tag{21}$$

$$h_t = \kappa^*\left(W_{F_t}\ [R_t\ , H_{t-1} \rightarrow,\ F_t]\right) \tag{22}$$

$$H_t \rightarrow= (1 - U_t)H_{t-1} \rightarrow + U_t\ h_t \tag{23}$$

where W is denoted as the forward GRU weight matrix, which BMDOA generally selects, $h_t$is denoted as the candidate HL, $H_t \rightarrow$is denoted as the output of the forward hidden state, η^* is denoted as the hard swish activation function by equation (24), and $\kappa^*$is denoted as ReLU by equation (25). Equation (24) introduces the nonlinearity characteristic to a deep learning model, and ReLU is an activation function that resolves the gradient problem,

$$\eta^* = 2F\left(0, \left(1,\ (\phi F * 0.2 + 0.5)\right)\right) \tag{24}$$

$$\kappa^* = (0,\ F) \tag{25}$$

where $F$is represented as a set of input characteristics and $\phi$the trainable parameter is indicated. The backward GRU then uses equation (26) to accept the input (optimal features set) in reverse,

$$H_t = [H \leftarrow_t,\ H \rightarrow_t] \tag{26}$$

At each t, it produces a hidden state through a sequence of computations, similar to the forward GRU$H \leftarrow_t$. This condition is then combined with the final result, it is classified as either a non-churn or churn client$H \rightarrow_t$.Longer training time and less efficient for intricate feature interactions than hybrid models.

3.4.3 Convolution and Attention with Bi-directional Gated Recurrent Unit (CAT-BiGRU) with AFO

To classify customers as either churn or non-churn, CAT-BiGRU is introduced. Adaptive FireFly Optimization (AFO) selects the GRU's weight values optimally, and Hard Swish and ReLU are used in place of CNN's conventional activation function to improve classification accuracy and reduce computational time. CAT-BiGRU model is presented that combines Convolutional Neural Network (CNN), Attention Mechanisms, and Bidirectional Gated Recurrent Unit (BiGRU) for the purpose of enhancing the predictive accuracy. BiGRU model is an advanced version of GRU, which preserves sequential information from customer interactions through both forward and backward processing of data. Both the past and future behavioral trends are understood in this bi-directional approach, which will result in better classification performance. CNN based Attention Mechanism (CAT): By concentrating on the most crucial elements causing churn, the attention mechanism improves feature extraction. CNN layers capture the spatial relationship and relationships between customers, and the attention mechanism gives higher weight to the attributes that are more relevant to the customer, making the learning process more focused. Use Hard Swish and ReLU instead of traditional activation function in CNN layers to further enhance the efficiency of the model and prevent vanishing/exploding gradients. To optimize the network parameters, the Adaptive Firefly Optimization (AFO)

method for metaheuristics has been introduced, which is inspired by the bioluminescent communication between fireflies. This guarantees optimal parameter tuning in order to achieve the best model convergence and avoid overfitting. The performances of churn classification are significantly improved due to the proposed CAT-BiGRU with AFO model which uses BiGRU, CNN-based Attention Mechanism, Hard Swish and ReLU activation functions, and Adaptive Firefly Optimization. This method allows the model to adapt to each customer's interactions in a bank, efficiently capture critical predictive attributes and is optimized for banking applications, ultimately helping banks identify customers with a high risk and implement effective retention strategies. CAT-BiGRU with AFO surpasses MWABGRU in incorporating the convolutional feature extraction, attention mechanism and adaptive optimization, leading to high improvement in classification accuracy, generalization and the ability to handle CCP.

4. RESULTS AND DISCUSSION

In this section, assessment methods like AUC, Kappa, Brier Score, EMPC and Classification Time are used to evaluate the effectiveness of the proposed technique with respect to the available approaches. Python 3.7, an Intel Core i7 CPU, an NVIDIA Geforce 930M graphics card, and 8 GB of RAM are used in the system's implementation. Predicting whether a client would churn that is, quit the bank is the aim of the Bank Client Churn dataset, which comprises of customer records from a financial organization. Because keeping existing customers is frequently more economical than finding new ones, churn prediction is essential for banks and other financial organizations. Dataset totally consists of 10,000 customers with 10 features with 1 target (churn), and 1 identifier (customer_id). The dataset consists of 10000 records, including 8000 non-churn samples and 2000 churn samples.

Preprocessing often includes,

- One-hot encoding for categorical variables (country, gender)
- Scaling numerical variables (age, balance, estimated_salary)
- Handling class imbalance (SMOTE, under sampling, or class weights).

This data is often used in categorizing churn prediction. There are a number of attributes with the data set, which can give us clues about the customer's behavior and demographics from https://www.kaggle.com/datasets/gauravtopre/bank-customer-churn-dataset. Some key features are,

1. Customer ID: An individual customer number.
2. Credit Score: A customer's creditworthiness is represented by a numerical score.
3. Geography: The country in which the client resides (such as France, Spain, or Germany).
4. Gender: The gender of the client (male or female).
5. Age: Age of the consumer.
6. Tenure: The length of time, measured in years, that the customer has stayed in connection with the bank.
7. Balance: The total amount of funds available in the customer's bank account.
8. Number of Products: The total number of banking services or products that the client uses, including credit cards, loans, and savings accounts.
9. Has Credit Card: A binary characteristic that indicates whether the client has a credit card issued by the bank (Yes/No).
10. Is Active Member: A binary indicator showing whether the customer actively utilizes the bank's services.
11. Estimated Salary: The customer's approximate annual income.
12. Exited (Target Variable): A binary outcome variable representing the customer's churn status, where 1 indicates that the customer has left the bank and 0 indicates that the customer has been retained.

Credit score, age, tenure, balance, product number, active member, and expected pay are selected as important characteristics.

Area Under Curve (AUC): The AUC metric assesses how effectively the model can distinguish between customers who are churning and those who are not. The true positive rate versus the false positive rate across various thresholds is represented graphically by the ROC graph, which is based on equations (27–29),

$$AUC = \int_0^1 True\ Positive\ Rate\ (TPR) \times False\ Positive\ Rate\ (FPR) \quad (27)$$

$$TPR = \frac{TP}{(TP + FN)} \quad (28)$$

$$FPR = \frac{FP}{(FP + TN)} \quad (29)$$

Brier Score: The Brier score is a technique for evaluating the accuracy of probabilistic predictions. The mean squared difference between the predicted probability and the actual binary outcomes is found using equation (30),

$$Brier\ Score = 1/N \sum_{i=1}^{N} (p_i - o_i)^2 \quad (30)$$

where p_i is, for example, the anticipated probability, $o_i$is the result that was observed (1 for churn, 0 for non-churn).

Kappa (Cohen's Kappa): Equation (31) includes the potential of chance agreement when computing Cohen's Kappa, a metric that assesses how effectively two or more predictions agree,

$$\kappa = \frac{p_0 - p_e}{1 - p_e} \quad (31)$$

where $p_0$is the denoted as observed agreement (proportion of correct predictions),$p_e$is the predicted, random agreement.

Expected Maximum Profit of the Customer (EMPC): By concentrating on maximizing earnings from accurately detected churn predictions using equation (32), EMPC assesses the possible profit gain from the customer churn prediction model,

$$EMPC = Profit \times (TP - FP) \quad (32)$$

where false positives are denoted by FP and true positives by TP.

Classification Time: Equation (33) considers the model's computational cost (time) needed to provide predictions for a specific number of occurrences,

$$Classification\ Time = \frac{Total\ Time\ Taken\ for\ Prediction}{Number\ of\ Instances\ Predicted} \quad (33)$$

True Positive (TP): Instances when churn was accurately predicted by the model.

True Negative (TN): Instances where non-churn was accurately predicted by the model.

False Positive (FP): Instances in which the true class is non-churn but the model predicted churn inaccurately.

False Negative (FN): Instances in which the actual class is churn while the model predicted non-churn.

Table 1 show the performance comparison of prediction methods with respect to evaluation metrics like AUC, Brier, Kappa, EMPC, and classification time. Proposed model has highest results of 99.21%, 8.48%, 98.61%, 10.93%, and 37.85 seconds for AUC, Brier, Kappa, EMPC, and classification time.

TABLE 1. RESULTS COMPARISON OF THE PREDICTION METHODS

Metrics(%)	RF	MLP	RNN	GRU	HTSADMLP	MWABGRU	CAT-BiGRU
AUC (%)	89.56	91.25	93.17	95.23	96.69	97.83	99.21
Brier (%)	21.32	19.64	17.41	15.78	12.95	10.56	8.48
Kappa	0.8764	0.9076	0.9289	0.9465	0.9607	0.9719	0.9861
EMPC (%)	4.21	5.35	6.43	7.57	8.65	9.76	10.93
Classification time (seconds)	183.61	139.23	115.84	64.26	55.24	50.42	37.85

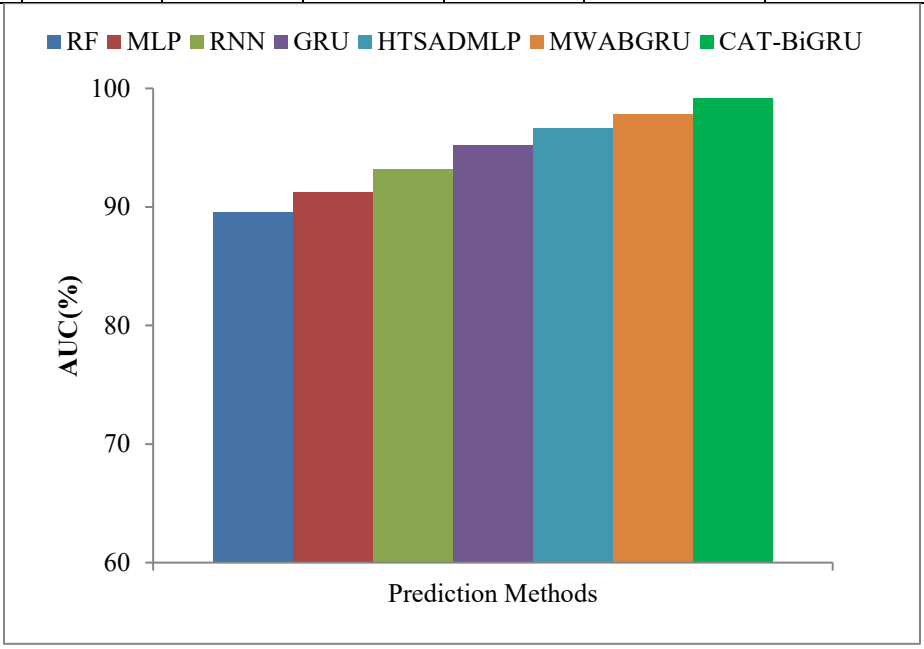


FIGURE 2. AUC COMPARISON OF PREDICTION METHODS

Figure 2, AUC results of the proposed and existing models. The proposed system has highest AUC of 99.21%, RF, MLP, RNN, GRU, HTSADMLP and MWABGRU has lowest AUC of 89.56%, 91.25%, 93.17%, 95.23%, 96.69%, and 97.83%. The proposed method has 9.65%, 7.96%, 6.04%, 3.98%, 2.52%, and 1.38% highest AUC when compared to RF, MLP, RNN, GRU, HTSADMLP and MWABGRU. The inclusion of attention mechanisms and optimized activation functions further enhances feature representation and learning capability.

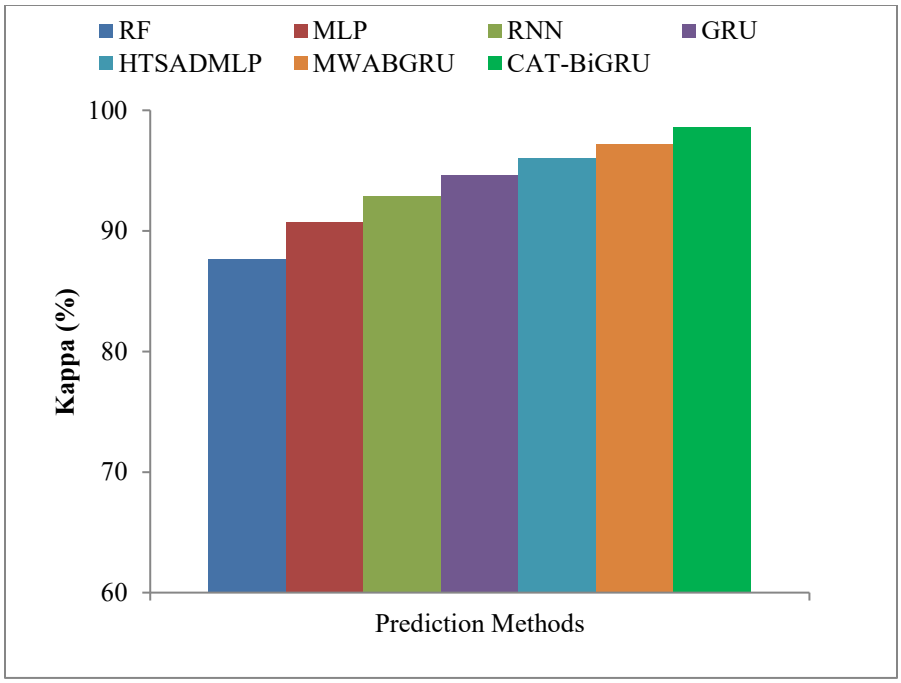


FIGURE 3. KAPPA COMPARISON OF PREDICTION METHODS

Figure 3, proposed system has highest Kappa of 98.61%, RF, MLP, RNN, GRU, HTSADMLP and MWABGRU has lowest Kappa of 87.64%, 90.76%, 92.89%, 94.65%, 96.07%, and 97.19%. The proposed method has 10.97%, 7.85%, 5.72%, 3.96%, 2.54%, and 1.42% highest Kappa when compared to RF, MLP, RNN, GRU, HTSADMLP and MWABGRU.

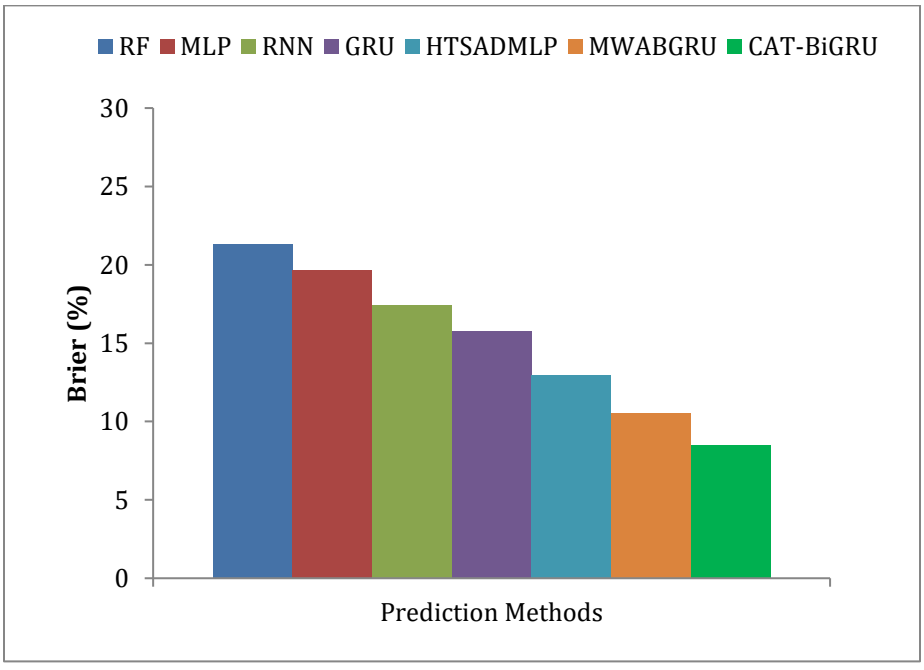


FIGURE4. BRIER COMPARISON OF PREDICTION METHODS

Situations where the model predicted non-churn but the actual class is churn. The proposed system has lowest brier of 8.48%, RF, MLP, RNN, GRU, HTSADMLP and MWABGRU has increased brier of 21.32%, 19.64%,

17.41%, 15.78%, 12.95%, and 10.56%. The proposed system has lowest brier of 12.84%, 11.16%, 8.93%, 7.30%, 4.47%, and 2.08% when compared to RF, MLP, RNN, GRU, HTSADMLP and MWABGRU.

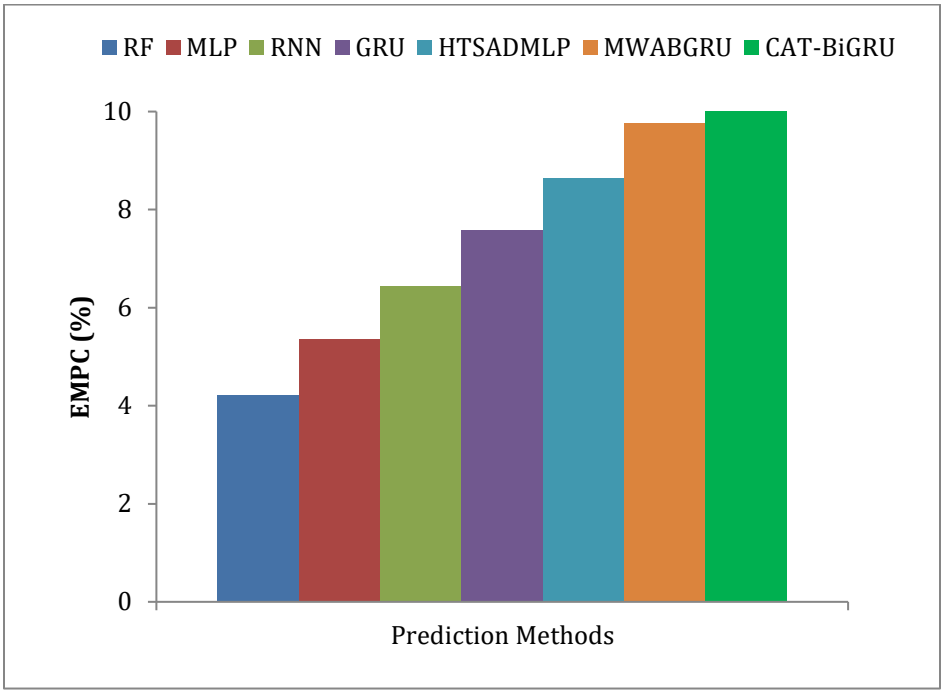


FIGURE 5. EMPC COMPARISON OF PREDICTION METHODS

Figure 5, proposed system has highest EMPC of 10.93%, RF, MLP, RNN, GRU, HTSADMLP and MWABGRU has lowest EMPC of 4.21%, 5.35%, 6.43%, 7.57%, 8.65%, and 9.76%. The proposed system has highest results of 6.72%, 5.58%, 4.50%, 3.36%, 2.28%, and 1.17% when compared to RF, MLP, RNN, GRU, HTSADMLP and MWABGRU.

5. CONCLUSION AND FUTURE WORK

This study presents a comprehensive comparative framework integrating data preprocessing, data balancing, feature selection, and optimized deep learning models. Initially, the raw banking dataset is preprocessed using mean and mode imputation, categorical encoding using one-hot and label encoding, and Z-score normalization, decimal scaling, and Min-Max normalization are examples of multi-stage normalizing methods, to ensure data consistency and stability. ISMOTE and the IKNN balancing method are conducted to overcome the class imbalance problem. For feature selection, BMDOA and the LFWOA approaches are introduced for dataset. BMDOA is introduced by integrating Brownian motion to improve random exploration and reduce premature convergence. It effectively balances exploration and exploitation through swarm behaviors such as separation, alignment, and cohesion. LFWOA combines the procedure of WOA, and Levy flight to significantly enhance global search capability. The bubble-net attacking mechanism ensures strong exploitation, while Levy flight introduces large random jumps, enabling efficient exploration of the search space. Furthermore, the study compares three deep learning models: (i) HTSADMLP, (ii) MWABGRU, and (iii) the proposed CAT-BiGRU optimized using Adaptive Firefly Optimization (AFO).HTSADMLP refers to the activation function adjustments and hyperparameter modification in traditional DMLP.MWABGRU is introduced for CC classification from the optimal feature set. The input is taken forward by GRU and backwards by the other. CAT-BiGRU model is introduced which integrates CNN, Attention Mechanisms, and BiGRU to improve predictive accuracy. AFO, a metaheuristic algorithm inspired by the bioluminescent signalling behavior of fireflies. The optimal values for AUC, Brier, Kappa, EMPC, and classification time are 99.21%, 8.48%, 98.61%, 10.93%, and 37.85 seconds for the proposed model. In addition to allowing banks to provide their services

in accordance with the expectations of new clients, the model may be expanded by utilizing additional evolutionary algorithms.

REFERENCES

1. Broby, D., 'Financial technology and the future of banking', Financial Innov. 7 (1), 1–19 (2021).
2. Văduva, A.G., Oprea, S.V., Niculae, A.M., Bâra, A., & Andreescu, A.I., 'Improving Churn Detection in the Banking Sector: A Machine Learning Approach with Probability Calibration Techniques', Electronics, 13(22), pp.1-33(2024).
3. de Lima Lemos, R. A., Silva, T. C. & Tabak, B. M., 'Propension to customer churn in a financial institution: A machine learning approach', Neural Comput. Appl. 34 (14), 11751–11768 (2022).
4. Keramati, A., Ghaneei, H. & Mirmohammadi, S.M., 'Investigating factors affecting customer churn in electronic banking and developing solutions for retention', Int. J. Electron. Bank. 2 (3), 185–204 (2020).
5. Soni, P. K., & Nelson, L., 'PCP: Profit-driven churn prediction using machine learning techniques in banking sector', International Journal of Performability Engineering, vol. 19, no. 5, pp. 303-311 (2023).
6. Muneer, A., Ali, R.F., Alghamdi, A., Taib, S.M., Almaghthawi, A. & Ghaleb, E.A., 'Predicting customers churning in banking industry: A machine learning approach', Indonesian Journal of Electrical Engineering and Computer Science, 26(1), pp.539-549 (2022).
7. Lalwani, P., Mishra, M.K., Chadha, J.S., & Sethi, P., 'Customer churn prediction system: a machine learning approach', Computing, 1-24(2022).
8. Vo, N.N., Liu, S., Li, X. & Xu, G., 'Leveraging unstructured call log data for customer churn prediction', Knowledge-Based Systems, 212, pp.1-21(2021).
9. Agarwal, V., Taware, S., Yadav, S.A., Gangodkar, D., Rao, A.L.N., & Srivastav, V.K., 'Customer-Churn Prediction Using Machine Learning', In 2022 2nd International Conference on Technological Advancements in Computational Sciences (ICTACS), pp. 893-899 (2022).
10. Imani, M., & Arabnia, H.R., 'Hyperparameter Optimization and Combined Data Sampling Techniques in Machine Learning for Customer Churn Prediction: A Comparative Analysis', Technologies, 11(6), pp.1-26(2023).
11. Imani, M., Beikmohammadi, A.,&Arabnia, H.R., 'Comprehensive Analysis of Random Forest and XGBoost Performance with SMOTE, ADASYN, and GNUS Under Varying Imbalance Levels', *Technologies, vol.* 13, no. 3,pp.1-40 (2025).
12. Lee, J., Yoon, Y., Kim, J., & Kim, Y.H., 'Metaheuristic-Based Feature Selection Methods for Diagnosing Sarcopenia with Machine Learning Algorithms', *Biomimetics*, *9*(3), pp.1-18(2024).
13. Mansoor, U., Sivakumar, V. & Jayabalan, M., 'Customer churn prediction in the banking sector on imbalance dataset', In International Conference on Integrated Intelligence and Communication Systems (ICIICS), 1–5 (2023).
14. Nisha,T.N., & Pramod, D., 'Improving bank customer churn prediction with feature reduction using GA', *Sci Rep* **15**, 40035, pp.1-20 (2025).
15. Rahman, M., & Kumar, V., 'Machine learning based customer churn prediction in banking,' In 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, pp. 1196–1201 (2020).
16. Ehsani, F., & Hosseini, M., 'Customer churn analysis using feature optimization methods and tree-based classifiers', Journal of Services Marketing, Vol. 39 No. 1 pp. 20–35 (2025).
17. Domingos, E., Ojeme, B., & Daramola, O., 'Experimental analysis of hyperparameters for deep learning-based churn prediction in the banking sector', Computation, 9(3), pp.1-19 (2021).
18. Banu, J.F., Neelakandan, S., Geetha, B., Selvalakshmi, V., Umadevi, A., & Martinson, E.O., 'Artificial Intelligence Based Customer Churn Prediction Model for Business Markets', Comput. Intell. Neurosci. 2022, 2022, 1–14 (2022).
19. Jajam, N., Challa, N.P., Prasanna, K.S.L., & Deepthi, C.H.V.S., 'Arithmetic Optimization with Ensemble Deep Learning SBLSTM-RNN-IGSA Model for Customer Churn Prediction', IEEE Access, 11, 93111–93128 (2023).
20. AlShourbaji, I., Helian, N., Sun, Y., Hussien, A.G., Abualigah, L. & Elnaim B., 'An efficient churn prediction model using gradient boosting machine and metaheuristic optimization', Sci Rep 13, 14441, pp.1-18, (2023).
21. Li, Y., Yang, Y., Song, P., Duan L., & Rui R., 'An improved SMOTE algorithm for enhanced imbalanced data classification by expanding sample generation space', Sci Rep 15, 23521, pp.1-21, (2025).
22. Zhao, J., Zhang, D., He, Q., & Li, L., 'A Hybrid-Strategy-Improved Dragonfly Algorithm for the Parameter Identification of an SDM', Sustainability, 15(15), pp.1-35 (2023).
23. Liang, Z., Shu, T., & Ding, Z., 'A Novel Improved Whale Optimization Algorithm for Global Optimization and Engineering Applications', *Mathematics*, *12*(5), pp.1-43 (2024).