A Deep Auto Imputation Integrated Bayes Optimized Transfer Learning Model with Hybrid Skill-Levy Search Algorithm (DAI-BOTS) for Call Drop Prediction in Mobile Networks

G. V. Ashok [1,*] and P. Vasanthi Kumari [2]

[1] Department of Computer Science and Engineering, Dayananda Sagar University, Bangalore, Karnataka, India
[2] Department of Computer Applications, Dayananda Sagar University, Bangalore, Karnataka, India
Email: gvashok1877@gmail.com (G.V.A.); vasanthi-bca@dsu.edu.in (P.V.K.)
*Corresponding author

***Abstract*—The most intriguing research areas in mobile communication networks recently is call drop prediction. Several artificial intelligence algorithms are created for this goal in earlier research projects; however, they struggle with the problems of large system complexity, poor efficiency, and lack of accuracy. Therefore, the goal of this research work is to create a cutting-edge framework, named as, DAI-BOTS for predicting mobile call drops using clever deep learning algorithms. Here, the special Deep Auto-Encoder based Data Imputation (DAE-DI) technique is used to generate the imputed data with normalized characteristics after data gathering. Then, a hybrid Skill Search based Levy Flight Optimization (S2LFO) method is created to select the most demanding features from the imputed data to lessen the classifier's training and testing complexity. In addition, for precise call dropout prediction, the Bayes Optimized Transfer Learning Network (BOT-LN) classification algorithm is used. In this work, there are four distinct and emerging datasets such as Call Detail Record (CDR), Synthetic Minority Over-sampling Technique (SMOTE), Cell to Cell and IBC Telco used to assess the call drop prediction results of the proposed DAI-BOTS mechanism. In addition, this study also uses a few other publicly available open-source datasets for system validation, including SMOTE, Cell to Cell, and IBM Telco. Furthermore, the results of the DAI-BOTS system are compared based on a number of evaluation factors, taking into consideration the most recent state-of-the-art model techniques.**

***Keywords*—mobile networks, call drop prediction, data imputation, feature optimization, machine learning, classification**

I. INTRODUCTION

In present times, the telecommunications industry has grown to be one of the most important in industrialized nations [1, 2]. As a result of technological advancements and a rise in operators, the intensity of competition has increased. In order to survive in this cutthroat market, businesses must employ a variety of strategies, with the three primary goals being to increase customer retention, promote existing customers, and acquire new ones. The emergence of 5G technology and changing preferences among consumers provides a plethora of opportunities for telecommunications (telecom) enterprises [3]. The telecom market is currently seeing intense competition and a high rate of customer turnover as a consequence of the tremendous economic opportunities that have recently arisen [4]. Telecom operators must create effective marketing strategies based on comprehensive customer data in order to cut abandonment of clients and boost business income. While acquiring new clients receives a lot of attention, this strategy is currently outdated in the fiercely competitive telecom sector [5, 6]. It is well known that getting new customers costs far more than maintaining existing ones. Churn control in the telecom sector becomes important in this situation. The cost of getting a new customer is substantially more than the cost of keeping an existing one, stated by the telecom service providers [7, 8]. In order to reduce customer churn, telecom carriers are directing their marketing efforts into focused client retention initiatives. One of the primary goals of churn management is to create accurate churn forecasting systems that can pinpoint customers who are likely to leave [9]. The main concept is to use a variety of information sources to build a profile of a client, including call routines contract details, invoices, payment, calls from customer service, and personality traits, and then forecast the likelihood that he or she would leave based on features.

Most people's life will be centered on the services offered by mobile wireless networks in the coming years, hence the demand for higher-quality services is only going to rise. The user would pay additional money if they had had to phone back to wrap up the conversation besides to the annoyance that a disrupted call would have caused. Call drops are generally avoidable by service providers if they take particular stages, eventually including distributing traffic across multiple frequencies, removing

Manuscript received December 25, 2023; revised January 9, 2024; accepted February 1, 2024; published August 15, 2024.

doi: 10.12720/jait.15.8.941-955

interference, and expanding the area of coverage [10−12]. Therefore, the only ways to increase the level of service in mobile networks within the already available spectrum are by fortifying the structure of networks and use innovative techniques for preventing call dropouts. The growing popularity of cellular networks today is expected to result in an enormous spike in network traffic. The length of phone calls is growing, which is a concerning trend. Call dropouts are frequently seen in networks—calls that are "dropped" without any of the parties intentionally disconnecting the phone calls [13, 14]. As a result, the only way to improve mobile network reliability of service within the boundaries of the spectrum at hand is to invest in network infrastructure upgrades and call drop reduction technologies.

Several existing models have been employed to address the challenge of call drop prediction in mobile networks. Common approaches include traditional machine learning techniques such as Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Decision Tree (DT), Logistic Regression (LR), Random Forest (RF), Ensemble Learning (EL), Extreme Gradient Boost (XGB), AdaBoost, and more. These models often rely on various feature engineering and optimization techniques to enhance prediction performance.

In the previous research works [15, 16], several types of learning techniques are developed for determining when calls will drop on mobile networks. Typically, machine learning has been effectively used in numerous fields, including computer vision, healthcare diagnosis, and other domains, and is a key enabler for artificial intelligence. Computers may now learn without explicit programming because of the science of machine learning [17]. In simple terms, there are three types of machine learning techniques: reinforcement-based, unsupervised, and supervised. With the help of specified sample inputs and their desired outcomes, which make up the labeled data set, the learning agent's goal in supervised learning is to acquire a general rule translating inputs to outputs.

Unsupervised learning involves a machine attempting to draw various patterns using its input without the use of labeled data. The application of deep learning [18, 19] has progressed recently, gaining more potent representational skills due to the development of quick, enormously parallel graphic processors and the substantial expansion of data. Still, it has drawbacks [20] like inefficient call drop detection, lower accuracy, not capable of handling large dimensional data, and high time complexity. The proposed study will therefore make use of an efficient and powerful ML for call drop prediction. The major goal of this study is to create a new Artificial Intelligence (AI) [21, 22] model combined with data imputation and optimization model to more accurately forecast call dropouts in mobile networks.

The main objectives of this paper are given below:

- The purpose of this research work is to develop a novel and sophisticated framework for mobile call drop prediction with the use of intelligent deep learning algorithms.
- After data acquisition, the unique Deep Auto-Encoder based Data Imputation (DAE-DI) technique is applied to generate the imputed data with normalized attributes.
- Then, the hybrid Skill Search based Levy Flight Optimization (S2LFO) technique is developed to choose the most required and pertinent features from the imputed data for reducing the training and testing complexity of classifier.
- The Bayes Optimized Transfer Learning Network (BOT-LN) classification algorithm is implemented for an accurate call dropout prediction.
- A real-time dataset called CDR and, some other public open-source datasets such as Synthetic Minority Over-sampling Technique (SMOTE), Cell to Cell and IBM Telco are used in this study for system validation.

The following sections make up the remaining portions of this work: A thorough literature analysis on the various machine learning and deep learning methods utilized in mobile networks for call drop prediction could be accessed in Section II. The proposed DAI-BOTS model is concisely thoroughly explained in Section III. By using the public open-source call record data, Section IV verifies and assesses the performance and comparative outcomes of the proposed model. Section V summarizes the findings, results, and future scope of the whole endeavor.

II. Literature Review

The present section examines various artificial intelligence techniques that support reliable communication in wireless and mobile networks by lowering call dropout rates. The obstacles and constraints of the earlier investigations are also examined for better understanding.

Khan *et al.* [23] utilized a standard Artificial Neural Network (ANN) algorithm for mobile call prediction by using the information of usage patterns, demographic data, and billing details. This study's major goal is to develop a Multilayer Perceptron Neural Network (MLP-NN) model that is capable of predicting telecom dropout by analyzing information gained from the telecommunications industry in accordance with variables. The requirement to anticipate and manage loss of customers has increased as the cellular business has grown. Proactive detection of loyal clients who wish to do business elsewhere can help in locating them and exposing them to preventative measures in an effort to keep them. Consequently, building trustworthy and accurate churn models is necessary for practitioners as well as telecom firm owners. To find a promising strategy for maintaining consistent customer baselines, telecom churn prediction has turned as an emerging technological challenge that provides an early warning system for consumers who are shifting over.

Ahmad *et al.* [24] applied a common machine learning technique on telecom data to forecast churn. The greatest benefit of this analysis is the advancement of a churn forecasting model that aids telecom companies in identifying customers who are most likely to switch. The model developed in this work uses machine learning techniques on a platform that gathers a lot of data and creates a novel method for figuring out features. In this instance, many data mining strategies for churn prediction are discussed, including preprocessing, feature transformation, selection, and classification. Businesses are attempting to create ways to anticipate potential customer migration because it directly affects their profitability, particularly in the telecom industry. To reduce client shifts, it is critical to understand the causes of it. This study aims to design and implement a framework for predicting churn to help telecom carriers pinpoint clients who are among the most inclined to experience churn. Pustokhina *et al.* [25] applied an Optimal Weighted Extreme Machine Learning (OWELM) technique for the identification and classification of customer churn prediction. Here, the SMOTE analysis is also carried out to handle an imbalanced dataset for accurately determining the class labels. Standardization and class labeling are first applied to the consumer data, and the imbalanced dataset is then dealt with the SMOTE approach. The WELM model is subsequently utilized to assign class labels to the applied data.

Ullah *et al.* [26] implemented a new churn prediction mechanism with the use of RF classification technique for improving customer relationship management in telecom sector. There are numerous data analysis and machine learning algorithms available that may be used to evaluate the customer data as a result of new advances in the area of big data. These methods examine the data to determine the causes of customer retention. These strategies can be used by CRM to boost their profit. The recommended work uses the information gain and ranking based feature selection approaches to choose the key attributes. Additionally, the authors classified churn and non-churn on two sizable datasets from the telecom industry adopting an array of machine learning approaches. Furthermore, the researchers claimed that when compared to other machine learning algorithms, the RF method provided better precision. Bahra *et al.* [27] deployed a vector auto-regression model for an effective handover management in mobile networks. Initially the authors used a mobility model constructed using statistical approaches along with deep learning techniques to gather user mobility patterns. To determine a user's potential trajectory, they also used an auto-regression model and a Gated Recurrent Unit (GRU). Here, the amount of pointless handover signaling messages has been decreased down, and the handover process is streamlined utilizing the predicted outcomes. Furthermore, studies have been carried out using the movement data collected from actual users. According to the model's findings, the suggested VAR-GRU mobility paradigm exhibits a smaller prediction error when compared to other approaches. The authors also looked into the expenses associated with handling and transferring handovers under both predictive and non-predictive circumstances.

Fujo *et al.* [28] implemented a deep back-propagated ANN classification technique for customer churn prediction in telecom industry. In the suggested framework, the missing data handling, over sampling, and transformation operations are performed while preprocessing the given data. Then, the feature engineering and exploratory data analysis processes are also carried out for improving the accuracy of churn prediction. Here, the data transformation is performed with the operations of label encoding, one hot encoding, and normalization. In addition, the correlation matrix formation and variance thresholding methodologies are applied in this study for feature selection. However, the suggested technique limits the main problems of inefficient churn prediction and high time complexity. Erunkulu *et al.* [7] utilized a standard ANN technique for identifying and predicting mobile call drops in the GSM networks. In this study, five different Quality of Service (QoS) parameters are considered for call drop prediction, which includes framework error rate, bit error rate, time, received signal quality and strength. These parameters are considered as the input parameters for classification, where the transformation operations are applied on the input. During classification, the weight values are adjusted according to the error rate of actual output and targeted result. However, the accuracy of the suggested technique is low, which affects the performance efficiency of the entire system.

Wu *et al.* [20] constructed an integrated customer analysis framework for an effective churn management in telecom industry. In this study, there are 6 different classifications such as LR, DT, NB, RF, AB and SMOTE used for churn prediction. After that, the factor analysis is separately performed with the use of Bayesian and LR approaches, and the customer segmentation is performed with the use of K-means technique. According to the predicted results of these techniques, the customer behavior analysis also validated in this study. Yet, the major challenges of this paper are high time cost, and complexity in classification. Hussain *et al.* [29] applied a data-driven deep learning approach for maximizing the QoS in 5G mobile networks. Here, the CDR data has been utilized to improve the security level of mobile communication. Moreover, the standard Adam optimization model is used in this study for predicting the normal and anomaly calls with faster response time. Rizwan *et al.* [30] implemented an AI enabled CDR model for enhancing the service management in mobile communication networks. In this study, the familiar CDR data is used to predict the specific type of cell performance. This framework comprises the modules of data preprocessing, clustering, pattern identification, root cause diagnosis, visualization, and classification. However, the computational complexity of the suggested framework is high, which affects the overall performance of the communication system.

Kolli *et al.* [31] utilized a hybrid set of features for accurately predicting churns from the mobile communication networks. The authors intend to analyze the behavior of customers with the use of individual profile information and social links. Nevertheless, the suggested model is not effective in churn prediction, since it fails with the major problems of lower performance rate and inefficient data handling. Gursoy *et al.* [32] conducted a case study to examine the churn behavior of customers in the telecom sector. Here also, the CDR data has been used for churn prediction, which comprises the information relevant to the phone usage of subscribers. In addition, the authors suggested a new prototype model for developing an effective churn prediction mechanism. The standard methods are therefore less effective at detecting service disruptions at small cells. With the advent of new technologies, it is projected that the number of design and optimization parameters would significantly rise. Expanding network administration parameters and identities may dramatically increase the likelihood of parameter misconfiguration. The recommended method employs threshold functions to constantly track downlink signal properties in order to determine whether a cell deviates from the acceptable level established by expert systems. A number of approaches have been developed in Ref. [33, 34] for lowering call dropouts in mobile networks, according to the study's findings. Tragically, most of the techniques have the following drawbacks:

- Ineffective results brought on by changeable dependencies.
- Time-consuming dataset training process.
- Lower efficiency.
- Extremely sensitive to noisy data.
- Unsuitable for handling large-dimensional datasets since the nature of the dataset has a substantial effect on how accurate the classifier performs.

In order to prevent call dropouts in a mobile network, the proposed research aims to develop a novel and sophisticated deep learning model.

III. Proposed Methodology

This section provides the complete explanation for the proposed framework, known as, Deep Auto Imputation based Bayes Optimized Transfer Learning model integrated with skill-Levy search algorithm (DAI-BOTS). The unique contribution of this work is the design and development of novel framework for call drop prediction in mobile networks. For this purpose, advanced and sophisticated algorithms are implemented in this study. Call dropouts often have a substantial impact on both voice and online consumers as well as service providers' income [13]. The work that needs to be done and the connections that need to be maintained with different company customers are both impacted by unexpected call drops. They undermine consumer confidence in telecom service providers [14]. There are several scenarios or use cases that can result in a call being lost.

The clear workflow model of the proposed system is shown in Fig. 1, which comprises the following modules:

- ✓ Input telecom data acquisition
- ✓ DAE-DI based data cleaning
- ✓ S2LFO based feature selection
- ✓ BOT-LN classification
- ✓ Performance testing and validation

In the proposed work, three different telecom datasets such as, real time CDR, and open source SMOTE, Call-to-Call and IBM-Telco have been used for system design and development. After telecom data collection, the novel DAE-DI mechanism is applied to generate the imputed data for improving the overall process of call drop prediction. It is a deep learning architecture model that transforms the raw data into a balanced form for gaining high accuracy. Consequently, the S2LFO technique is applied to choose the most required and useful features from the imputed data for minimizing the training as well testing complexity of classification. Moreover, the BOT-LN classification algorithm is utilized to accurately predict the call drops with high accuracy and performance results. By using the combination of these methodologies, the overall call drop prediction system is greatly improved in the proposed system.

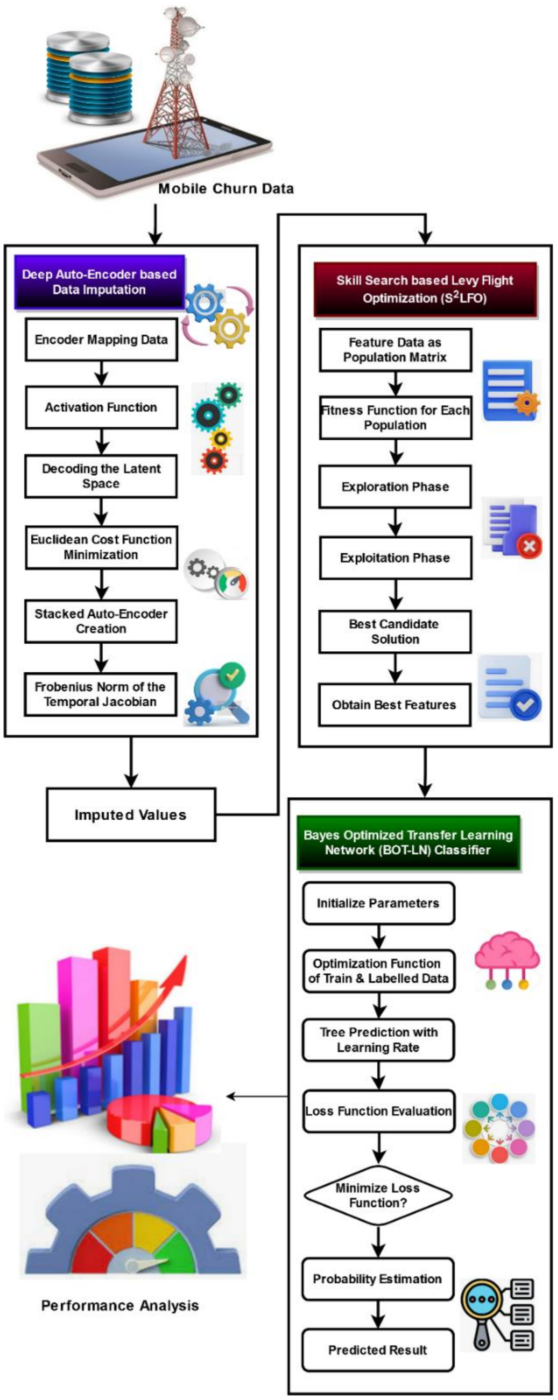


Fig. 1. Flow of the proposed DAI-BOTS model.

A. Data Imputation

Typically, data preparation is an important stage of information discovery activities, which is involved in this framework. It necessitates a number of stages, including the transformation of data as well as information reduction. The efficacy and reliability of learning algorithms will be hampered if raw data is transformed into poor-quality data [35]. As a result, by following proper procedures for data preprocessing and selecting appropriate learning algorithms, the acquired data can be accurately examined. Telecom datasets include a number of problems that need to be solved, including values that are missing, non-numerical features, conflicting feature scales, etc. As a result [36], it is crucial to preprocess the data before applying a learning model. In this study, four different types of datasets are adopted for system performance assessment and evaluation, which includes CDR, Cell2Cell, and IBM Telco, in which Cell to Cell has 58 attributes and 71,047 occurrences.

The data is accessible through Kaggle and the "Centre for Customer Relationship Management Duke University's" website. IBM Telco is the second set of data used in the study for performance assessment and validation. Since the dataset includes a variety of customer-related information, a consistent link has been established between the churn rate and the customer's behavior. Many academics have used the IBM Telco data set to predict customer turnover, which is available in Kaggle and has 21 attributes and 7044 rows. We have created a system that makes use of cutting-edge tools and technologies to obtain real-time data from a specific company or service provider, which is typically difficult to perform. The points of entry where the feature file can be initialized are constructed using the coded methods and functions. The following fields are considered when creating CDR data, however if more are required, the framework can help with their production.

In previous studies, the different types of imputation mechanisms are used for data normalization and cleaning. For instance, attribute-based clustering, distance-based clustering, and some others are used for transforming the raw input dataset into a balanced form. However, the conventional approaches have the major limitations of reduced speed of processing, presence of outliers, and redundant information. Therefore, the proposed work aims to implement a novel as well as intelligent mechanism, named as, DAE-DI for dataset imputation. This technique comprises both encoder and decoder units, where the input is mapped into the latent space with the use of encoder. In this stage, the input telecom data $\mathcal{J}^d$ is taken as the input for processing, and the imputed data $\acute{f}_s''$ is delivered as the resultant data. Here, the data mapping is performed at first as represented in the following equation:

$$\mathrm{H}^{s} = \delta\left(\mathrm{E_r}\mathcal{J}^{\mathrm{d}}\right) \tag{1}$$

where, E_r indicates encoder, D_r represents the decoder, and $\delta(.)$ represents the activation function tanh. Consequently, the decoder D_r is projected into the latent space H^s back to the input space as illustrated in below:

$$\mathcal{J}^d = D_r H^s = D_r \times \delta(E_r \mathcal{J}^d) \tag{2}$$

Moreover, the Euclidean cost function is minimized at the time of training as defined in the following equation:

$$\underset{D_r,\,E_r}{\operatorname{argmin}} \|\mathcal{J}^d - D_r \times \delta(E_r \mathcal{J}^d)\|_F^2 \tag{3}$$

where, ῂ is the regularization coefficient, and the regularizer K can include simple Tikhonov penalties on the encoder and decoder as well as intricate assumptions like sparsity and position deficits. Typically, several variants of auto-encoders are used for imputation, here the stacked auto-encoder has been used by nesting one with other as represented in the following model:

$$\underset{D_r,\,E_r}{\operatorname{argmin}} \left\|\mathcal{J}^{\mathrm{d}} - D_r \times \delta(E_r \mathcal{J}^d)\right\|_F^2 + ῂ K(E_r, D_r) \tag{4}$$

Furthermore, Froebenius norm of the temporal Jacobian is estimated with the expectation matrix value as represented in the following form:

$$\acute{f}_s'' = \sum_{j=1}^{L} \mathbb{E}\left(\sum_{t=1}^{B} \left|\sigma\left(\mathcal{J}^{\mathrm{d}}\right)\right|^2\right) \tag{5}$$

where, L indicates the network layer, $\sigma(\mathcal{J}^d)$ denotes the variance of formed by Jacobian matrix, B is the dimension of feature data, and $\mathbb{E}(.)$ indicates the expectation matrix values. Finally, the featured data is generated with the imputed values as defined in the following model:

$$\acute{f}_s'' = D_r \times \rho[E_r(K)] \tag{6}$$

where, $\rho(.)$ is the sigmoid activation function applied at the encoder layer, in the neural network. By using this data imputation algorithm, the raw telecom dataset is transformed into the featured data, which is further used for call drop prediction operations. The following Algorithm 1 gives the step-wise functioning of the proposed DAE-DI model.

Algorithm 1: Deep Auto Encoder based Data Imputation (DAE-DI)

Input: Input Data $\mathcal{J}^{\mathrm{d}}$;
Output: Imputed data $\acute{f}_s''$;
Procedure:
Step 1: An auto-encoder comprise both encoder $\mathrm{E_r}$ and decoder $\mathrm{D_r}$ units, where the encoder maps the imputed data $\mathcal{J}^{\mathrm{d}}$ into the latent space ($\mathrm{H^s}$) by using Eq. (1);
Step 2: Then, the decoder $\mathrm{D_r}$ projects the latent space $\mathrm{H^s}$ back to the input as shown in Eq. (2);
Step 3: The Euclidean cost function is minimized at the time of training by using Eq. (3);
Step 4: The stacked auto-encoder has been created by nesting one with another as represented in Eq. (4);
Step 5: The Froebenius norm of the temporal Jacobian matrix is formulated as shown in Eq. (5);
Step 6: Finally, the imputed values for all the featured data $\acute{f}_s''$ is represented in Eq. (6).

B. Hybrid Skill Search Based Levy Flight Optimization (S2LFO) Algorithm

Since the effectiveness of the call drop prediction system heavily depends on the dataset's features, feature selection is typically one of the most crucial operations. Therefore, it is more important than ever to take the best features from the available data. For call drop prediction in the mobile network application systems, numerous feature selection mechanisms based on optimization techniques have been proposed in previous studies. However, the majority of approaches have drawbacks related to time requirements, local optimums, and sophisticated mathematical modeling, as well as increased time consumption. Therefore, the proposed study aims to apply a new hybrid algorithm, named as, S2LFO for feature selection. In this method, the most recently developed Skill Search Optimization (SSO) as well as Levy Flight Optimization (LFO) techniques are integrated together for dataset dimensionality reduction.

One of the novel contribution of this work is the hybridization of these methodologies for feature selection. The SSO is motivated by people's efforts to acquire better abilities. The folks or members seek to learn new things in order to develop their talents. The method starts by randomly initializing the members. Along with the worst and best members, the fitness values are adjusted while the process of iteration. Exploration and exploitation are the two steps that SSO performs for updating the population of members. The skill-learning process is carried out by professionals in the very first stage, whilst the subsequent stage rely on events and personal efforts. The primary goal of the exploration phase is to move the algorithm members with the assistance of other members in order to find the overall solution in the area of searching. When the exploring power is raised, the first ideal area is correctly recognized. In contrast, the exploitation phase concentrates on local search, which aids in the convergence of better solutions. The Hybrid S2LFO algorithm selects Levy Flight Distances through a random process during optimization. This exploration phase is vital for the algorithm to effectively navigate the search space and discover optimal solutions. The specific values for Levy Flight Distances are not predetermined but are instead randomly assigned during the optimization process.

By working with an expert member who is in good condition based on fitness value, the population member aims to improve his skill during exploration. The expert member is chosen at random from among the participants who have higher fitness scores than the person being considered. Through learning the ability to do this, it directs the population member to various spots within the area of search. Each team member works independently and practices during the exploitation phase to improve the skill they learned during the discovery phase. To encourage exploitation, it is designed to look like a local search, prompting the member to make an effort to improve his fitness worth. Levy flight is a statistical motion description that goes beyond the extremely common Brownian motion, which was found more than a century ago.

Levy statistics are increasingly being used to characterize a wide variety of both supernatural and artificial systems. Each "generation" employs a certain set of particles. The method will produce a completely new generation starting from one well-known site at random Levy flight distances. The most promising member of the new generation will then be chosen after evaluation. Up until the stopping criteria are met, the process is repeated. Levy Flight Optimization (LFO) has been implemented in a fairly straightforward manner. Both techniques have the unique characteristics of high convergence speed, global optimum solution with low iterations, increased searching efficiency and performance rate. Therefore, the proposed integrates these two distinct optimization techniques for developing a hybridized model to solve the feature selection problem. In this technique the imputed data obtained from the previous stage $\dot{\mathrm{f}}_{\mathrm{d}}$ is taken as the input, and the set of selected features $\ddot{\mathrm{f}}_{\mathrm{s}}$ are delivered as the resultant output. At first, the population matrix is formulated for the given feature data as represented in the following equation:

$$\mathcal{X} = \begin{bmatrix} \mathcal{X}_1 \\ \cdot \\ \cdot \\ \cdot \\ \mathcal{X}_{\mathrm{j}} \\ \cdot \\ \cdot \\ \cdot \\ \mathcal{X}_{\mathcal{M}} \end{bmatrix} = \begin{pmatrix} x_{1,1} \cdots x_{1,\mathrm{a}} \cdots x_{1,\mathrm{g}} \\ x_{\mathrm{j},1} \cdots x_{\mathrm{j,a}} \cdots x_{\mathrm{j,g}} \\ x_{m,1} \cdots x_{m,\mathrm{a}} \cdots x_{m,\mathrm{g}} \end{pmatrix} \tag{7}$$

The fitness function of each candidate solution is calculated for each row in the population matrix by using the following model:

$$\mathcal{F} = \begin{bmatrix} \mathcal{F}_1 \\ \cdot \\ \cdot \\ \cdot \\ \mathcal{F}_{\mathrm{j}} \\ \cdot \\ \cdot \\ \cdot \\ \mathcal{F}_{\mathcal{M}} \end{bmatrix} = \begin{bmatrix} \mathrm{fitness}(\mathcal{X}_1) \\ \cdot \\ \cdot \\ \cdot \\ \mathrm{fitness}(\mathcal{X}_{\mathrm{j}}) \\ \cdot \\ \cdot \\ \cdot \\ \mathrm{fitness}(\mathcal{X}_{\mathcal{M}}) \end{bmatrix} \tag{8}$$

When the new location's fitness value grows better, it is approved; this can be modeled as foll $x_{j,a}^{\varphi_1} = x_{j,a} + rand \times (E_{j,a} - R \times x_{j,a}),\ E_j = X_t, if F_t < F_j$ow.

$$x_{\mathrm{j,a}}^{\varphi_1} = x_{\mathrm{j,a}} + \mathrm{rand} \times (\mathcal{E}_{\mathrm{j,a}} - \mathcal{R} \times x_{\mathrm{j,a}}), \mathcal{E}_{\mathrm{j}} = \mathcal{X}_{\mathrm{t}} \quad \text{if } \mathcal{F}_{\mathrm{t}} < \mathcal{F}_{\mathrm{j}} \tag{9}$$

$$x_{\mathrm{j,a}} = \begin{cases} x_{j,a}^{\varphi_1} & \text{if } \mathcal{F}_{\mathrm{j}}^{\varphi_1} < \mathcal{F}_{\mathrm{j}} \\ x_{j,a} & \text{if } \mathcal{F}_{\mathrm{j}}^{\varphi_1} \geq \mathcal{F}_{\mathrm{j}} \end{cases} \tag{10}$$

Then the probability chosen from Levy distribution is computed according to the following equation:

$$\mathcal{R} = \begin{cases} \frac{(\vartheta_{jh})^{\gamma} \times (\rho_{jz})^{\lambda}}{\Sigma(\vartheta_{jh})^{\gamma} \times (\rho_{jz})^{\lambda}} & h, z \in \text{allowed} \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

To encourage exploitation, it is designed to look like a local search, encouraging the member to try to improve his fitness worth. This might be done as shown in below:

$$x_{j,a}^{\varphi_2} = \begin{cases} x_{j,a} + \frac{1-2\times \text{rand}\times x_{j,a}}{I_t} & \text{if rand} < 0.5 \\ x_{j,a} + \frac{L_k^B + \text{rand}\times(U_k^B - L_k^B)}{I_t} & \text{else} \end{cases} \quad (12)$$

$$x_{j,a} = \begin{cases} x_{j,a}^{\varphi_2} & \text{if } \mathcal{F}_{\mathrm{j}}^{\varphi_2} < \mathcal{F}_{\mathrm{j}} \\ x_{j,a} & \text{if } \mathcal{F}_{\mathrm{j}}^{\varphi_2} \geq \mathcal{F}_{\mathrm{j}} \end{cases} \quad (13)$$

The following can be used to represent the fitness function:

$$fitness(\mathcal{X}_1) = \int_o^{I_t} \left[\sum_{j=0}^{\mathcal{M}} \left|\Delta\mathcal{F}_j + \Delta x_{j,a}\right| t\right] dt \quad (14)$$

Finally, the best featured data is obtained as represented in the following equation:

$$\dot{f}_s'' = \prod_j \text{fitness}(\mathcal{X}_j), \text{if fitness}(\mathcal{X}_{\mathrm{j}}) < \text{fitness}(\mathcal{X}_{j-1})$$

$$\dot{f}_s = \prod_j \text{fitness}(X_j), \text{if fitness}(X_j) < \text{fitness}(X_{j-1}) \quad (15)$$

By using this hybrid optimization algorithm, the best fitness value is estimated and is used to choose the most relevant features from the provided dataset. The list of symbols and its appropriate descriptions are given in Table I.

TABLE I. LIST OF SYMBOLS AND DESCRIPTIONS

Symbols	Descriptions
$\mathcal{X}_j$	j^{th} candidate
$x_{j,g}$	g^{th} variable value proposed via the ith member of the population
$\mathcal{M}$	Number of members
m	Number of considered variable
$\mathcal{F}_j$	Fitness value of the j^{th}candidate solution
$x_{j,a}^{\varphi_1}$	Updated position of member j
φ_1	a in the first phase
$\mathcal{F}_j^{\varphi_1}$	Fitness value of $x_{j,a}^{\varphi_1}$
$\mathcal{E}_j$	Selected expert member
$\mathcal{E}_{j,a}$	a^{th} dimension of the expert member
rand	Random number in range [0, 1]
$\mathcal{R}$	Probability chosen from Levy distribution
ϑ	Pheromone
ρ	Attractiveness
γ and λ	Impact factors for ϑ and ρ.
$x_{\mathrm{j,a}}^{\varphi_2}$	Updated position of member j
$\mathcal{F}_{\mathrm{j}}^{\varphi_2}$	Fitness value of $x_{j,a}^{\varphi_2}$
I_t	Number of iterations
U_k^B and L_k^B	Upper and lower limits of the j^{th} variable
$\dot{f}_s''$	Featured data

The following Algorithm 2 provides the step-wise operations involved in the proposed S2LFO model.

The DAI-BOTS framework highlights the importance of feature engineering to improve model performance. For call drop prediction in mobile networks, feature engineering involves extracting temporal features related to network traffic, statistical measures, and call handover dynamics. The framework integrates interpretability techniques to illuminate the model's decision-making process. Feature importance analysis and SHapley Additive exPlanations (SHAP) values are used to attribute the model's prediction to individual features, making the decision-making process transparent and explainable. The combination of advanced feature engineering and interpretable methods in the DAI-BOTS framework enhances the model's predictive capabilities and fosters trust among stakeholders involved in mobile network management.

Algorithm 2: Hybrid Skill Search based Levy Flight Optimization (S2LFO)

Input : Imputed Data $\dot{f}_d$;
Output : Selected Features $\dot{f}_s''$;
Procedure:
Step 1: Generate the population matrix from the given featured data as represented in Eq. (7);
Step 2: The fitness function $\mathcal{F}$ of each candidate solution is calculated for each row in the population matrix by using Eq. (8);
Step 3: //Exploration Phase:
When the fitness value of the new location $x_{j,a}^{\varphi_1}$ improves, it is accepted. This can be modeled by using Eqs. (9) and (10);
Step 4: Estimate the probability $\mathcal{R}$ value chosen from the Levy distribution as represented in Eq. (11);
Step 5: //Exploitation Phase:
To encourage exploitation, it is designed to look like a local search $x_{j,a}^{\varphi_2}$, encouraging the member to try to improve his fitness worth. This might be done as shown in Eq. (12);
Step 6: Compute the fitness function $\text{fitness}(\mathcal{X}_1)$ using Eq. (13);
Step 7: Obtain the best featured data $\dot{f}_s''$ using Eq. (14).

C. Bayes Optimized Transfer Learning Network (BOT-LN) Classifier

The selected features are passed to the BOT-LN classifier at this stage, which uses appropriate training and testing procedures to provide an accurate call drop prediction. Numerous machine learning and deep learning techniques have been used in earlier research to anticipate call drops. However, the majority of approaches are constrained by the key issues of increasing overfitting, decreased accuracy, prolonged training, and high system complexity. As a result, the proposed study aims to leverage the BOT-LN unique learning mechanism to anticipate call dropouts from the provided data. It is a methodical approach to looking at and analyzing datasets to find patterns and guarantee that reliable results are generated in accordance with the stated objectives. BOT-LN Classifier uses probability distributions to represent beliefs about model parameters given observed data. The optimal decision function is derived through Bayesian optimization, updating posterior distributions to improve decision-making. Bayesian information processor integrates information from posterior densities to guide decision-making. The specifics of the family of posterior densities, state, and optimal decision function may vary depending on the details of the Bayesian optimization approach adopted.

During training, it is intended that a regression-based machine learning model will recognize and highlight correlations between the data input values and the desired output reply. In this technique, the training set with the chosen features is considered into account for classification, and the predicted for identifying call dropouts is delivered as the output. After getting the features, the learning rate, tree and posterior probability value are initialized for classification. Then, the expected loss is minimized as represented in the following equation:

$$L_E(t) = \min_{\beta} \int \left(\varphi - \beta(t)\right)^2 \delta(\varphi|t)\, d\varphi \tag{16}$$

where, $\delta(\varphi|t)$ is the family of posterior probability density, φ is the state, $\beta(t)$ is the optimal decision function, $\vartheta_{(\varphi|t)}$ represents the posterior mean, and these parameters are calculated as shown in below:

$$\beta(t) = \vartheta_{(\varphi|t)} \tag{17}$$

$$\delta(\varphi|t) = \frac{F(\varphi|t)\, g_d(\varphi)}{\tau(t)} \tag{18}$$

where, $F(\varphi|t)$ indicates the conditional density function, $g_d(\varphi)$ represents the prior density, and $\tau(t)$ denotes the Bayesian information processor that is calculated in below:

$$\tau(t) = \int F(\varphi|t)\, g_d(\varphi) \tag{19}$$

Moreover, the final risk decision solution is estimated as shown in the following model:

$$R_C = \int L_E(t) \times \tau(t)\, dt \tag{20}$$

During the training process, a regularization term is computed by the hyper parameter $\mathbb{I}_{\vartheta}$ and the process is mathematically described as represented in the following equations:

$$f\left(f''_{s_i}\right) = \left|\bar{\mathcal{Y}} - P_{t-1}\left(f''_s\right)\right|_i^g \tag{21}$$

$$T_a\left(f''_s\right) = \underset{g}{\operatorname{argmin}} \sum_{t=1}^{A^T}\left[\bar{\mathcal{Y}} - f\left(f''_{s_t}\right)\right]^2 + \mathbb{I}_{\vartheta} \times \left|f''_{s_t}\right|_t^2 \tag{22}$$

$$\mathcal{G}\left(f''_s\right) = \underset{g}{\operatorname{argmin}} \sum \left(\bar{\mathcal{Y}} - \mathbb{I}_{\vartheta} \times T_a\left(f''_s\right)\right)^2 \tag{23}$$

$$\mathcal{L}\left(\mathcal{Y}, f\left(f''_{s_t}\right)\right) = \sum_{t=1}^{A^T}\left[\bar{\mathcal{Y}} - f\left(f''_{s_t}\right)\right]^2 \tag{24}$$

$$P\left(f''_s\right) = P_{t-1}\left(f''_s\right) + \beta \times \sum_{t=1}^{A^T} T_a\left(f''_s\right)_t \tag{25}$$

By using these functions, the prediction result is determined as $P\left(f''_s\right)$ that is used to accurately identify the call dropouts with high performance rate. The following Algorithm 3 describes the classification process of proposed BOT-LN model.

Algorithm 3: Bayes Optimized Transfer Learning Network (BOT-LN) Classifier

Input : Training set: $\{(\mathcal{Y}_1, f''_{s_1}),\ (\mathcal{Y}_2, f''_{s_2}), \ldots,\ (\mathcal{Y}_n, f''_{s_n})\}$; Learning rate value, $\mathbb{I}_{\vartheta}$ and Tree number A^T, obtained through Bayes optimal decision;

Output : Predicted result $P\left(f''_s\right)$;

Step 1: Compute the family of the posterior densities $\delta(\varphi|t)$ of the state φ;

Step 2: Minimize the expected loss $L_E(t)$ using Eq. (15);

Step 3: Compute the optimal decision function $\beta(t)$ and bayes information processor $\tau(t)$ as shown in Eqs. (16)–(18);

Step 4: Obtain the final risk decision solution R_C using Eq. (19);

Step 5: Loss function, $\mathcal{L}\left(\mathcal{Y}, f\left(f''_{s_t}\right)\right)$

Step 6://Training Process:

For t = 1 to A^T, **do** :

$f\left(f''_{s_i}\right) = \left|\bar{\mathcal{Y}} - P_{t-1}\left(f''_s\right)\right|_i^g$

Train $T_a\left(f''_s\right)$ using $\left(f''_s,\ \bar{\mathcal{Y}}\right)_i^g$

A regularization term controlled by another hyper parameter $\mathbb{I}_{\vartheta}$;

Comput $T_a\left(f''_s\right)$e using Eq. (21);

Compute $\mathcal{G}\left(f''_s\right)$ using Eq. (22);

Compute loss function $\mathcal{L}\left(\mathcal{Y}, f\left(f''_{s_t}\right)\right)$ based on Eq. (23);

if $\min_{t = 1 \text{ to } A^T} \mathcal{L}\left(\mathcal{Y}, f\left(f''_{s_i}\right)\right)$

$P\left(f''_s\right) = P_{t-1}\left(f''_s\right) + \beta \times \sum_{t=1}^{A^T} T_a\left(f''_s\right)_t$

end if

end for;

Step 7: **Return** predicted result $P\left(f''_s\right)$.

The accuracy of call drop prediction in mobile networks can be improved by the feature importance and relevance analysis utilized in the proposed DAI-BOTS model. This analysis involves evaluating the significance of each feature in contributing to the predictive performance of the model. The Hybrid S2LFO algorithm is used for this feature selection process, which combines SSO and LFO to reduce dataset dimensionality. SSO focuses on learning from the best-performing members of the population, while LFO introduces randomness for effective exploration.

The analysis includes generating a population matrix and calculating the fitness function for each candidate solution, evaluating the importance of each feature. There are exploration and exploitation phases involved in this process. During exploration, SSO explores the search space to find the global solution, while LFO focuses on local search to refine the solution. Levy Flight Distances are chosen randomly during the optimization process, allowing for effective exploration in the search space. Probability values selected from the Levy distribution guide the exploration process, influencing the selection of features. Expert member learning enhances skills during exploration, directing members to various spots in the search space.

Fitness improvement occurs as members independently practice during the exploitation phase, attempting to improve their fitness values. The fitness function is calculated during the exploitation phase, emphasizing local search. The algorithm aims to obtain the best-featured data based on fitness values, selecting the most relevant features. The holistic approach of combining SSO and LFO ensures a comprehensive evaluation of feature relevance. The BOT-LN classifier is used to assess the impact on call drop prediction. In this stage, selected features undergo further evaluation. Bayesian optimization is employed to determine the learning rate, tree number, and posterior probability values, which impact the classification process.

IV. Result and Discussion

This section validates the performance and results of the proposed DAI-BOTS method by using the 4 distinct datasets such as SMOTE, CDR, Cell to Cell and IBM Telco [20, 28, 37]. Also, several performance indicators are used for validating the outcomes of the proposed call drop prediction model. The hardware and software environment consist of Python version 3.12, with libraries TensorFlow and scikit-learn. The hardware setup comprises an Intel Core i5 processor running at 2.7 GHz and 8GB RAM, and the experiments are conducted using Jupyter Notebook. An example IBM dataset called IBM-Telco is used to forecast customer churn and create client retention strategies. Teradata Center for Customer Relationship Management at Duke University provides Cell to Cell dataset.

The DAI-BOTS model is configured with a three-layered DAE-DI. The encoder has 128 nodes in the first layer and 64 in the second, mirrored in the decoder. Tanh activation functions are employed in both the encoder and decoder layers, with a learning rate of 0.001. A stacked auto-encoder with two layers, each containing 64 nodes, is utilized, and the regularization coefficient is set to 0.001. The Hybrid S2LFO is fine-tuned with a population size of 50 and 100 iterations. SSO includes exploration and exploitation phases, with a learning rate of 0.01. LFO involves Levy flight distances that are randomly chosen. Feature selection is performed using probability values determined during optimization. The BOT-LN is configured with a learning rate of 0.01, 100 trees, and a Bayes optimal decision mechanism. The training setup utilizes a regression-based machine learning model, and regularization terms are hyperparameter-controlled.

Data preprocessing involves missing value imputation using DAE-DI on the raw telecom dataset, standard scaling for data normalization, encoding for categorical features, and addressing outliers and conflicting feature scales. Feature selection is achieved through the S2LFO algorithm, considering probability values, exploration and exploitation phases, and fitness function details. The training process includes batch sizes of 32 and 100 epochs and the Adam optimizer.

In order to assess the effectiveness of models, accuracy, precision, recall, F1-Score, and Area Under the Curve (AUC) are measured. The percentage of samples that the algorithm appropriately predicts to all samples is known as accuracy. Precision is the percentage of samples that are actually positive compared to those that are expected to be positive. The proportion of correctly anticipated positive examples in the actual group is known as recall, while the proportion of correctly predicted negative instances within the real sample is known as specificity. The following equations are used to compute these parameters:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \tag{26}$$

$$\text{Sensitivity} = \frac{TP}{TP+FN} \times 100\% \tag{27}$$

$$\text{Specificity} = \frac{TN}{TN+FP} \times 100\% \tag{28}$$

$$\text{Precision} = \frac{TP}{TP+FP} \times 100\% \tag{29}$$

$$\text{F1}-\text{Score} = \frac{2\times\text{Precision}\times\text{Sensitivity}}{\text{Precision}+\text{Sensitivity}} \times 100\% \tag{30}$$

where, True Positive (TP) denotes that the sample is indeed positive and meets the prediction that it will be positive. False Negative (FN) signifies that the sample has turned out positive even though it was expected to be negative. False Positive (FP) denotes a sample that was expected to be positive but is really negative. True Negative (TN) implies that the sample is both genuinely and predictably negative. By calculating the False Positive Rate (FPR) and True Positive Rate (TPR) of the classifier and using these as both the vertical and horizontal axes, respectively, the Receiver Operating Characteristics (ROC) curve of the predictions might be generated. The region under the ROC curve is known as AUC, and a higher AUC number is highly preferable for accurate classification.

$$\text{FPR} = \frac{FP}{FP+TN} \times 100\% \tag{31}$$

$$\text{TPR} = \frac{TP}{TP+FN} \times 100\% \tag{32}$$

Addressing data imbalance is a crucial aspect of enhancing the robustness and fairness of the DAI-BOTS framework. Imbalanced datasets, where one class outnumbers the other, can lead to biased model training and suboptimal predictive performance. To mitigate these challenges, advanced preprocessing techniques are employed, such as resampling or generating synthetic samples. During the data preparation phase, advanced preprocessing techniques create a more balanced distribution of classes to ensure that the model is exposed to an equitable representation of both positive and negative instances. Resampling methods are employed to address data imbalance, either by oversampling the minority class, under-sampling the majority class, or a combination of both. Oversampling duplicates instances from the minority class, while under-sampling removes instances from the majority class. The goal is to achieve a more balanced class distribution, preventing the model from being biased toward the majority class.

In addition, synthetic sample generation techniques like SMOTE are applied to augment the minority class instances. SMOTE creates synthetic instances that resemble the minority class, thereby providing a more diverse and balanced dataset. The integration of these advanced preprocessing techniques aims to improve the model’s ability to generalize across both classes, leading to better predictive performance on imbalanced datasets. This ensures that the model’s predictions are not skewed by the dominance of one class, enhancing its overall effectiveness in call drop prediction in mobile networks.

The proposed S2LFO technique’s fitness curve with regard to varying iterations is depicted in Fig. 2. Since a good optimization technique can arrive at the best optimal solution with few iterations, the optimization efficiency and searching performance are evaluated according to its fitness curve. The results demonstrate that the suggested

S2LFO technique achieves the most optimal result in the search space with a minimal number of iterations. The DAI-BOTS framework uses several time-series analysis techniques to improve its call drop prediction capabilities. It employs time-series data preprocessing, temporal feature engineering, RNNs, seasonal decomposition, and time-series cross-validation. The framework preprocesses raw datasets, extracts relevant temporal features, integrates RNNs, applies seasonal decomposition techniques and uses time-series cross-validation methods. The imputed data is dynamically updated to capture temporal dynamics and ensure the model remains relevant.

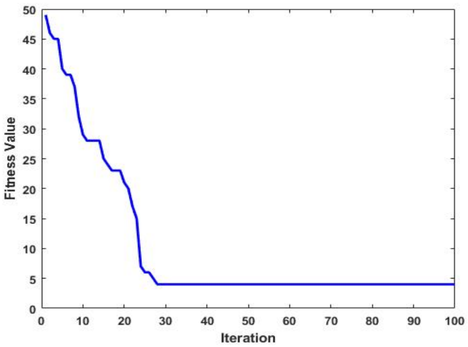


Fig. 2. Fitness curve.

Moreover, the confusion matrix for Cell to Cell dataset is depicted in Fig. 3, and its appropriate ROC curve with and without S2LFO technique is shown in Fig. 4. The effectiveness and accuracy of the classifier's prediction rate are examined using the ROC and confusion matrix parameters, respectively. The results demonstrate that by precisely detecting the call drops from the provided datasets, the suggested DAI-BOTS model offers the best prediction results. Furthermore, it performs significantly better when combined with the S2LFO algorithm. Since it significantly affects the reduction of feature dimensionality during training and testing of the data.

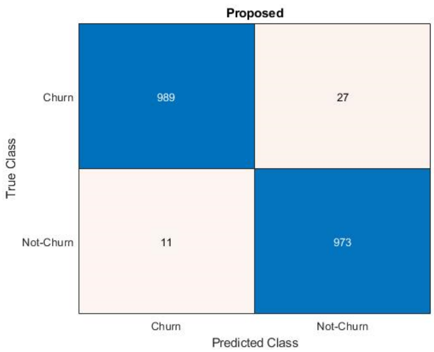


Fig. 3. Confusion matrix for cell to cell dataset.

With the help of the SMOTE dataset, Fig. 5 confirm and contrast some of the most popular machine learning techniques with the suggested model. As a result, as shown in Fig. 6, additional characteristics like precision, recall, and F1-Score are also validated and compared to the suggested DAI-BOTS model. The results show that the suggested DAI-BOTS technique functions admirably for the provided dataset and delivers excellent results by precisely forecasting the call dropouts.

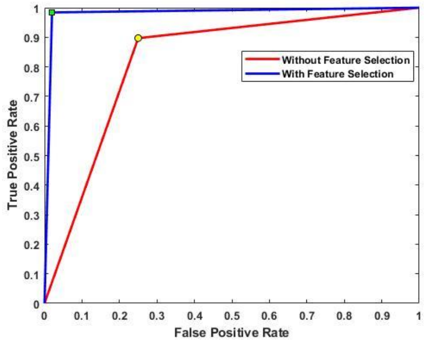


Fig. 4. ROC curve with and without S2LFO technique.

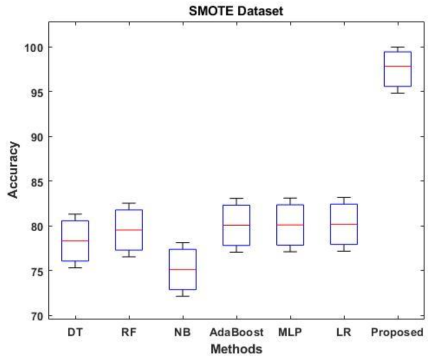


Fig. 5. Performance comparison with classic machine learning models using SMOTE dataset.

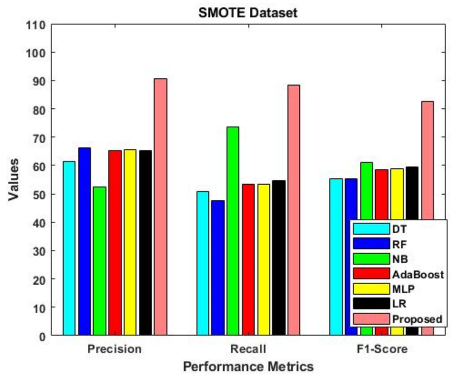
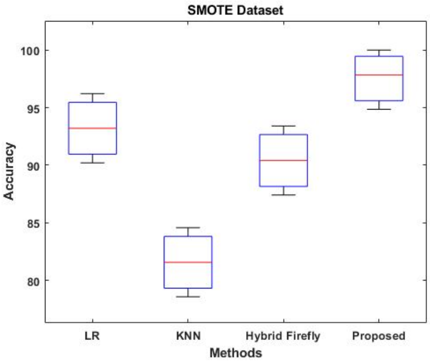
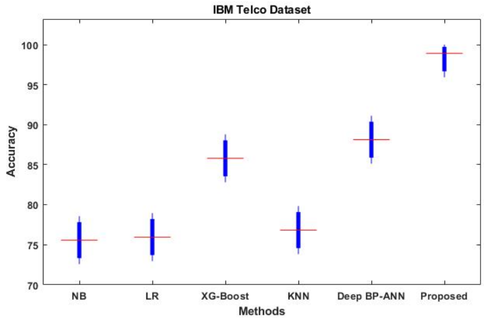


Fig. 6. Performance comparison using SMOTE dataset.

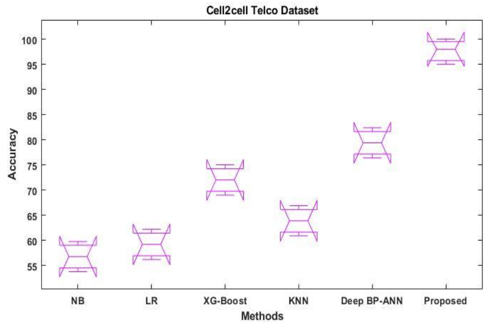
Additionally, as shown in Fig. 7, the accuracy of the SMOTE, IBM Telco and Cell to Cell datasets is estimated and compared for the current and suggested DAI-BOTS models. These results also show that the proposed model's accuracy has been significantly enhanced by the addition of DAE-DI-based imputation and S2LFO-based feature selection methods. Due to the fact that S2LFO is used to reduce dimensionality and increase dataset quality at the time of imputation. The suggested call drop prediction system improves the overall training and testing procedures of the BOT-LN classifier by utilizing these factors. As a result, the DAI-BOTS model performs much more accurately than conventional categorization strategies.



(a) Accuracy analysis using SMOTE dataset.



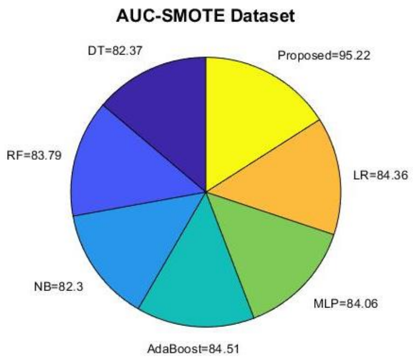
(b) Accuracy analysis using IBM Telco dataset.



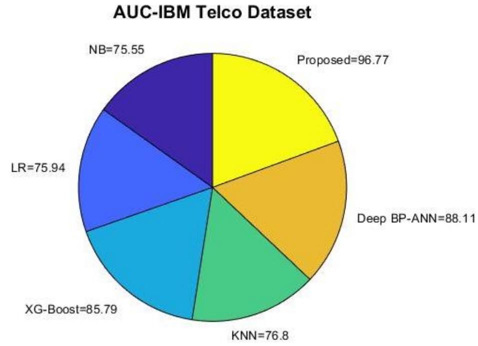
(c) Accuracy analysis using Cell to Cell dataset.

Fig. 7. Accuracy analysis using SMOTE, IBM Telco and Cell to Cell dataset.

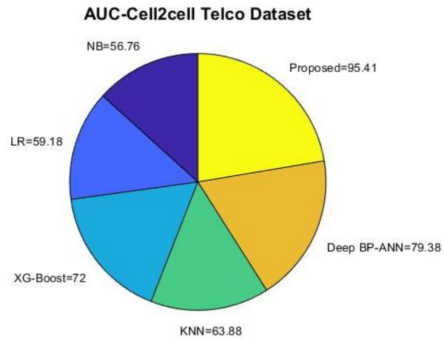
In addition, the AUC is also estimated for the SMOTE and IBM-Telco datasets for both conventional and proposed approaches as shown in Fig. 8 respectively. According to the predictions, it is noted that the AUC values provided by the DAI-BOTS model is superior to the other classification algorithms for all the datasets.



(a)



(b)

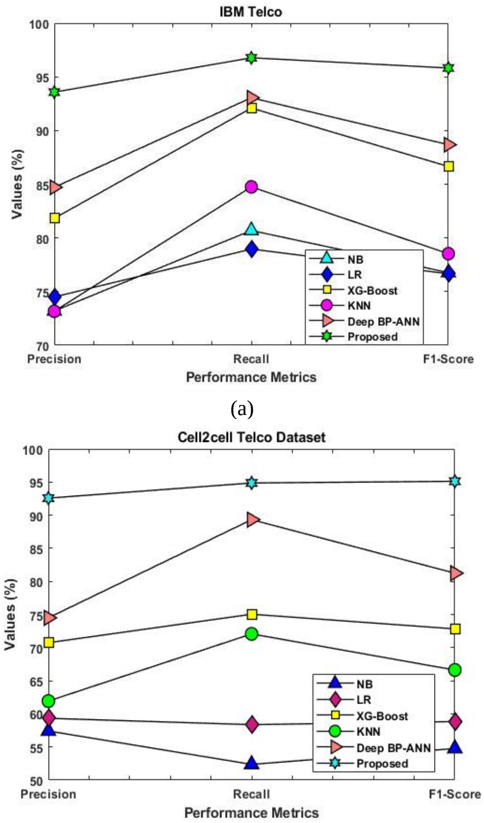


(c)

Fig. 8. AUC analysis. (a) AUC analysis using SMOTE dataset; (b) AUC analysis using IBM-Telco dataset; (c) AUC analysis using Cell to Cell dataset.

By using the IBM Telco and Cell to Cell datasets, respectively, Fig. 9 confirm and compare the precision, recall, and F1-Score values for the traditional and new call drop prediction algorithms. According to different numbers of epochs, the proposed DAI-BOTS technique additionally computes the overall training and testing loss, as illustrated in Fig. 10. The next step is to validate and compare the overall call drop prediction accuracy of the recently developed deep learning and the suggested BOT-LN classifiers, as shown in Fig. 11. Overall results lead to the conclusion that the proposed DAI-BOTS outperforms other current classification models and yields superior outcomes.

The proposed DAI-BOTS framework is benchmarked against existing methods to ensure its effectiveness. This is done by evaluating it using a diverse set of metrics, covering different aspects of model performance. The purpose of benchmarking is to validate the proposed approach in call drop prediction in mobile networks. Various methods commonly used in call drop prediction will be compared with the proposed DAI-BOTS framework, including SVM, KNN, DT, LR, NB, RF, EL, XGB, AdaBoost, COLKR, and other relevant methods identified during the literature review. The evaluation metrics employed for benchmarking ensure a comprehensive understanding of the proposed approach's strengths and weaknesses. These metrics include accuracy, precision, recall, F1-Score, ROC-AUC, and Confusion Matrix. The benchmarking process uses standardized datasets representative of real-world telecom scenarios, ensuring the generalizability of results. The proposed DAI-BOTS framework is trained and tested on these datasets, and the aforementioned metrics will be calculated and compared against the baseline methods.



(b)

Fig. 9. Performance comparison. (a) Performance comparison using IBM Telco dataset; (b) Performance comparison using Cell to Cell dataset.

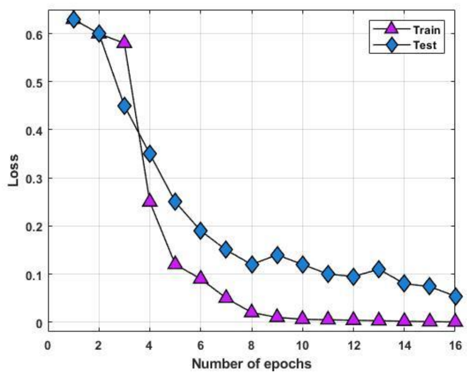


Fig. 10. Overall training and testing loss.

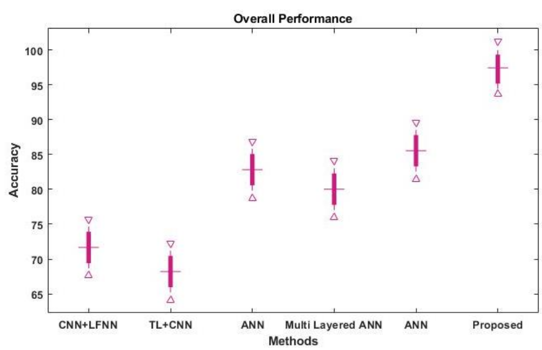


Fig. 11. Overall prediction accuracy.

The benchmarking process contributes to the robustness of the comparison, ensuring that the proposed approach's performance is rigorously evaluated across various dimensions. Sensitivity analyses, cross-validation, and statistical tests will be conducted to analyse the benchmarking results. This provides insights into the strengths and limitations of the DAI-BOTS framework compared to existing methods in call drop prediction in mobile networks. The comprehensive benchmarking process will form a critical component of validating the proposed DAI-BOTS framework's efficacy.

Table II validates the error rate of the conventional and proposed classification approaches used for call drop prediction based on CDR dataset. Typically, the classifier's overall prediction performance may suffer from the higher error rate. Hence, it should be reduced through efficient training and testing to achieve superior outcomes. The calculated results demonstrate that the DAI-BOTS model has a significantly reduced error rate of 0.065 when compared to the other models. Due to efficient data imputation and appropriate feature optimization, the classifier's training and testing procedures run smoothly. As a result, the suggested DAI-BOTS generates superior results when compared to existing classifiers. As indicated in Table III, additional metrics, including sensitivity, specificity, and F1-Score, are also evaluated as well as compared for the preexisting and hypothesized models. These findings also demonstrate that DAI-BOTS performs significantly better than the alternative options. The prediction rate has increased as a result of the greater sensitivity, specificity, and F1-Score values.

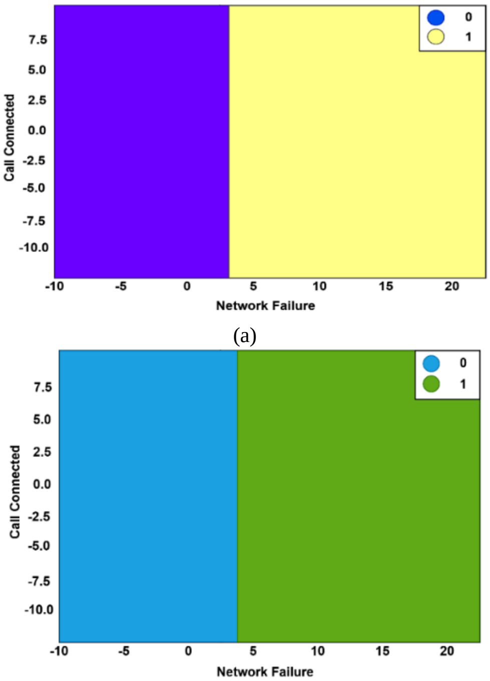
TABLE II. ERROR RATE ANALYSIS USING CDR DATASET

Techniques	Error rate
Support Vector Machine (SVM)	0.36
K-Nearest Neighbor (KNN)	0.33
Decision Tree (DT)	0.324
Logistic Regression (LR)	0.255
Naïve Bayes (NB)	0.319
Random Forest (RF)	0.238
Ensemble Learning (EL)	0.24
Extreme Gradient Boost (XGB)	0.39
AdaBoost	0.42
COLKR	0.098
Proposed	0.065

TABLE III. OVERALL PERFORMANCE ANALYSIS

Techniques	Sensitivity (%)	Specificity (%)	F1-Score (%)
Support Vector Machine (SVM)	88.1	88.3	88
K-Nearest Neighbor (KNN)	88.9	88.5	88.2
Decision Tree (DT)	90.2	89.8	89.3
Logistic Regression (LR)	94.6	94.1	94.22
Naïve Bayes (NB)	90.7	91.1	90.3
Random Forest (RF)	92.9	92.5	91.9
Ensemble Learning (EL)	92.1	92.1	91.5
Extreme Gradient Boost (XGB)	88.96	90.1	89.76
AdaBoost	87.7	88.2	87.9
COLKR	99.1	99.3	99.2
Proposed	99.5	99.4	99.5

Fig. 12(a) and (b) demonstrates the training and testing results of the proposed DAI-BOTS classification model with regard to of network failure and call connected rate respectively. This analysis demonstrates the performance of the classifier by validating the training and testing outcomes. By utilizing the optimized feature set obtained with the aid of S2LFO, the training and testing efficiency outcomes of DAI-BOTS in the proposed call drop prediction system have been significantly improved. DAI-BOTS outperforms existing models with improved accuracy in call drop prediction. Advanced techniques like deep auto-imputation and hybrid skill-levy search contribute to this enhancement. The model also reduces training and testing complexity by employing efficient algorithms like S2LFO for feature selection, enhancing dataset quality. The integration of deep auto-imputation and transfer learning techniques further enhances the efficiency of call drop prediction, providing more reliable and timely predictions.



(b)

Fig. 12. Training and testing results of the proposed DAI-BOTS classification model. (a) Training and (b) Testing results.

The evaluation of the DAI-BOTS framework revolves around its computational complexity and scalability. In terms of computational complexity, the DAE complexity depends on factors like the number of layers, nodes, and dataset size. The Stacked Auto-Encoder, despite introducing added complexity through nesting, effectively manages it using Tanh activation functions and regularization. The Hybrid S2LFO algorithm, integrating SSO and LFO, introduces complexity influenced by population size, iterations, and dataset dimensionality, with the random choice of Levy Flight Distances contributing stochastic elements affecting computational complexity. The BOT-LN classifier introduces complexity through Bayesian optimization during learning rate, tree number, and posterior probability determination, with regularization terms and hyperparameter-controlled processes contributing further.

Regarding scalability, DAE-DI's scalability relies on its ability to handle large datasets, which may become computationally intensive as the dataset size increases. S2LFO's scalability is influenced by population size and dataset dimensionality, requiring more computational resources for larger datasets. The scalability of the BOT-LN classifier is influenced by the training set size and the complexity of Bayesian optimization, where handling large datasets and optimizing hyperparameters efficiently may impact scalability. Overall, DAI-BOTS combines deep learning and optimization for effective call drop prediction, with a hybrid approach in S2LFO balancing exploration and exploitation, and BOT-LN's Bayesian optimization enhancing adaptability. Considerations include substantial computational resources for computational complexity during training and vital scalability considerations for large-scale telecom datasets. Recommendations include exploring parallel processing and distributed computing frameworks for enhanced scalability and implementing optimization strategies like mini-batch training to manage computational complexity during training.

The implementation and deployment of the DAI-BOTS framework face various challenges such as computational resources, large-scale data handling, model interpretability, real-time predictions, hyperparameter tuning, model robustness, regulatory and privacy compliance, and integration with existing systems. These challenges are addressed by using techniques such as parallel processing, distributed computing, and cloud-based services, data preprocessing techniques, leveraging big data technologies, model interpretability techniques, optimizing model architectures, and using efficient algorithms. Automated techniques, such as Bayesian optimization and grid search streamline the hyperparameter tuning process, while extensive testing, validation, and continuous monitoring can ensure model robustness. Collaboration with legal experts helps address regulatory and privacy compliance. Finally, a multidisciplinary approach combining expertise in machine learning, telecommunications, and domain-specific knowledge, along with continuous feedback loops and iterative improvements, is crucial for successful deployment of the DAI-BOTS framework in real-world mobile network environments.

Research is conducted to improve the prediction framework of DAI-BOTS by integrating network anomalies and external factors. This integration aims to enhance the model's robustness and real-world applicability, taking into consideration the dynamic nature of mobile networks and the potential impact of various external influences. The integration of network anomalies involves a meticulous examination of irregularities or deviations from the expected behavior within the telecom infrastructure. By assimilating information about network anomalies, the DAI-BOTS model can recognize and adapt to abnormal patterns, thereby improving its resilience in the face of unforeseen disruptions.

Simultaneously, the research explores the inclusion of external factors that may influence call drop occurrences. These external factors could include environmental variables, weather conditions, or events affecting specific geographical regions. By considering these external dynamics, the DAI-BOTS model aims to capture a more comprehensive understanding of the contextual factors contributing to call drop events. This holistic approach recognizes the interconnectedness of various elements impacting mobile networks and seeks to create a prediction framework that is not only adept at handling routine scenarios but also resilient in the face of anomalies and external influences. The anticipated outcome is an enriched DAI-BOTS model that aligns more closely with the complexities of real-world mobile network environments, thereby improving its efficacy in predicting and mitigating call drop incidents.

V. CONCLUSION

Every day, telecommunications sectors produce enormous volumes of data, where the people making decisions underlined that it is more difficult to acquire new customers than to retain current ones. Call dropouts continue to be a problem despite the best efforts of authorities throughout the years. One of the most pressing issues in mobile networks is the call drop prediction, which still occurs. In order to do this, this research introduces a novel framework known as, DAI-BOTS for call drop prediction in mobile networks. For the system design and development in the proposed work, three different telecom datasets including real-time CDR, open source SMOTE, Call-to-Call, and IBM-Telco have been used. Following the acquisition of telecom data, the innovative DAE-DI technique is used to produce imputed data in order to enhance the call drop prediction process overall. It is a deep learning architecture model that balances unprocessed data to achieve high accuracy. To reduce the training and testing complexity of classification, the S2LFO technique is used to choose the most important and practical characteristics from the imputed data. Additionally, the BOT-LN classification algorithm is used to forecast call drops with good accuracy and performance outcomes. The proposed approach significantly improves the call drop prediction system overall by combining these methodologies. The proposed DAI-BOTS model's outcomes are contrasted with some of the most recent state-of-the-art call drop prediction methodologies for performance evaluation and validation. The suggested DAI-BOTS performs well and offers improved results up to 99.5% accuracy with a lower error rate up to 0.065 for all the datasets utilized in this investigation, according to the findings. Future work can be expanded upon by putting in place a new security architecture to safeguard mobile communication networks.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest.

AUTHOR CONTRIBUTIONS

G. V. Ashok involved in design, theoretical formalism, performed the analytic calculations and performed the numerical simulations and evaluation process presented in the paper. P. Vasanthi Kumari helped to derive the mathematical equation and background study of the paper and also provided a factual review and helped to edit the manuscript. All authors had approved the final version.

REFERENCES

[1] S. Saleh and S. Saha, "Customer retention and churn prediction in the telecommunication industry: A case study on a Danish university," *SN Applied Sciences*, vol. 5, 173, 2023.

[2] A. Amin, A. Adnan, and S. Anwar, "An adaptive learning approach for customer churn prediction in the telecommunication industry using evolutionary computation and Naïve Bayes," *Applied Soft Computing*, vol. 137, 110103, 2023.

[3] S. A. Panimalar and A. Krishnakumar, "Customer churn prediction model in cloud environment using DFE-WUNB: ANN deep feature extraction with weight updated tuned Naïve Bayes classification with Block-Jacobi SVD dimensionality reduction," *Engineering Applications of Artificial Intelligence*, vol. 126, 107015, 2023.

[4] A. Şenyürek and S. Alp, "Churn prediction in telecommunication sector with machine learning methods," *International Journal of Data Mining, Modelling and Management*, vol. 15, pp. 184–202, 2023.

[5] J. Xu, J. Liu, T. Yao, and Y. Li, "Prediction and big data impact analysis of telecom churn by backpropagation neural network algorithm from the perspective of business model," *Big Data*, 2023.

[6] G. Pratibha and N. P. Hegde, "Low complexity early employee attrition analysis using boosting and non-boosting ML techniques," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 11, pp. 246–256, 2023.

[7] O. O. Erunkulu, E. N. Onwuka, O. Ugweje, and L. A. Ajao, "Prediction of call drops in GSM network using artificial neural network," *Jurnal Teknologi dan Sistem Komputer*, vol. 7, pp. 38–46, 2019.

[8] F. Tang, B. Mao, Y. Kawamoto, and N. Kato, "Survey on machine learning for intelligent end-to-end communication toward 6G: From network access, routing to traffic control and streaming adaption," *IEEE Communications Surveys and Tutorials*, vol. 23, pp. 1578–1598, 2021.

[9] Y. Beeharry and R. T. Fokone, "Hybrid approach using machine learning algorithms for customers' churn prediction in the telecommunications industry," *Concurrency and Computation: Practice and Experience*, vol. 34, no. 4, e6627, 2022.

[10] K. Anjana and S. Urolagin, "Churn prediction in telecom industry using machine learning algorithms with K-Best and principal component analysis," in *Proc. the First Global Conference on Artificial Intelligence and Applications (GCAIA 2020)*, 2021, pp. 499–507.

[11] M. J. Jafari, M. J. Tarokh, and P. Soleimani, "An interpretable machine learning framework for customer churn prediction: A case study in the telecommunications industry," *Journal of Industrial Engineering and Management Studies*, vol. 10, pp. 141–157, 2023.

[12] L. Geiler, S. Affeldt, and M. Nadif, "A survey on machine learning methods for churn prediction," *International Journal of Data Science and Analytics*, vol. 14, pp. 217–242, 2022.

[13] A. Bugajev, R. Kriauzienė, O. Vasilecas, and V. Chadyšas, "The impact of churn labelling rules on churn prediction in telecommunications," *Informatica*, vol. 33, pp. 247–277, 2022.

[14] H. Nalatissifa and H. F. Pardede, "Customer decision prediction using deep neural network on telco customer churn data," *Electronics and Telecommunications Journal*, vol. 21, pp. 122–127, 2021. (in Indonesian)

[15] S. A. Panimalar and A. Krishnakumar, "A review of churn prediction models using different machine learning and deep learning approaches in cloud environment," *Journal of Current Science and Technology*, vol. 13, pp. 136–161, 2023.

[16] S. Karthiga and A. Abirami, "Hybrid deep learning model for efficient prediction of telecom data using EMF radiation," *Journal of Intelligent and Fuzzy Systems*, pp. 1–16, 2023.
[17] Y. Liu, Z. Yu, C. Chen, Y. Han, and B. Yu, "Prediction of protein crotonylation sites through Light GBM classifier based on SMOTE and elastic net," *Analytical biochemistry*, vol. 609, 113903, 2020.
[18] N. Tamuka and K. Sibanda, "Real time customer churn scoring model for the telecommunications industry," in *Proc*. *2020 2nd International Multidisciplinary Information Technology and Engineering Conference (IMITEC)*, 2020, pp. 1–9.
[19] P. Lalwani, M. K. Mishra, J. S. Chadha, and P. Sethi, "Customer churn prediction system: a machine learning approach," *Computing*, pp. 1–24, 2022.
[20] S. Wu, W.-C. Yau, T.-S. Ong, and S.-C. Chong, "Integrated churn prediction and customer segmentation framework for telco business," *IEEE Access*, vol. 9, pp. 62118–62136, 2021.
[21] G. Ashok and V. Kumari, "An investigation of various machine learning techniques for mobile call data analysis for reducing call drop," *Materials Today: Proceedings*, vol. 51, pp. 2476–2478, 2022. https://doi.org/10.1016/j.matpr.2021.11.622
[22] G. Ashok and P. V. Kumari, "To Analyse and optimise the issues in call drops and enhance the real time (CDR) processing," *Easy Chair*, pp. 2516–2314, 2020.
[23] Y. Khan, S. Shafiq, A. Naeem, S. Ahmed, N. Safwan, and S. Hussain, "Customers churn prediction using Artificial Neural Networks (ANN) in telecom industry," *International Journal of Advanced Computer Science and Applications*, vol. 10, 2019.
[24] A. K. Ahmad, A. Jafar, and K. Aljoumaa, "Customer churn prediction in telecom using machine learning in big data platform," *Journal of Big Data*, vol. 6, pp. 1–24, 2019.
[25] I. V. Pustokhina, D. A. Pustokhin, P. T. Nguyen, M. Elhoseny, and K. Shankar, "Multi-objective rain optimization algorithm with WELM model for customer churn prediction in telecommunication sector," *Complex and Intelligent Systems*, pp. 1–13, 2021.
[26] I. Ullah, B. Raza, A. K. Malik, M. Imran, S. U. Islam, and S. W. Kim, "A churn prediction model using random forest: Analysis of machine learning techniques for churn prediction and factor identification in telecom sector," *IEEE Access*, vol. 7, pp. 60134–60149, 2019.
[27] N. Bahra and S. Pierre, "A hybrid user mobility prediction approach for handover management in mobile networks," *Telecom*, pp. 199–212, 2021.
[28] S. W. Fujo, S. Subramanian, and M. A. Khder, "Customer churn prediction in telecommunication industry using deep learning," *Information Sciences Letters*, vol. 11, 24, 2022.
[29] B. Hussain, Q. Du, S. Zhang, A. Imran, and M. A. Imran, "Mobile edge computing-based data-driven deep learning framework for anomaly detection," *IEEE Access*, vol. 7, pp. 137656–137667, 2019.
[30] A. Rizwan, M. Jaber, F. Filali, A. Imran, and A. Abu-Dayya, "A zero-touch network service management approach using AI-enabled CDR analysis," *IEEE Access*, vol. 9, pp. 157699–157714, 2021.
[31] N. Kolli and N. Balakrishnan, "Hybrid features for churn prediction in mobile telecom networks with data constraints," in *Proc. 2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 2020, pp. 734–741.
[32] U. F. Gursoy, M. Kurulay, and M. S. Aktas, "Customer churn behavior predicting using social network analysis techniques: A case study," in *Proc*. *2021 International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, 2021, pp. 1–6.
[33] M. Duraipandian, "Long term evolution-self organizing network for minimization of sudden call termination in mobile radio access networks," *Journal of Trends in Computer Science and Smart Technology (TCSST)*, vol. 2, pp. 89–97, 2020.
[34] R. Su, D. Zhang, R. Venkatesan, Z. Gong, C. Li, F. Ding *et al.*, "Resource allocation for network slicing in 5G telecommunication networks: A survey of principles and models," *IEEE Network*, vol. 33, pp. 172–179, 2019.
[35] G. Kaur, R. K. Goyal, and R. Mehta, "An efficient handover mechanism for 5G networks using hybridization of LSTM and SVM," *Multimedia Tools and Applications*, vol. 81, pp. 37057–37085, 2022.
[36] L. Xu, X. Zhao, Y. Yu, Y. Luan, L. Zhao, X. Cheng *et al*., "A comprehensive operation and revenue analysis algorithm for LTE/5G wireless system based on telecom operator data," in *Proc*. *2019 IEEE Smart World, Ubiquitous Intelligence and Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, 2019, pp. 1521–1524.
[37] N. N. Vo, S. Liu, X. Li, and G. Xu, "Leveraging unstructured call log data for customer churn prediction," *Knowledge-Based Systems*, vol. 212, 106586, 2021.